И. К. Сафронов

Задачник-практикум по ИНФОРМАТИКЕ

Санкт-Петербург
«БХВ-Петербург»
2002

УДК 681.3.06(075.3)
ББК 32.973.26-018
С21

Сафронов И. К.

С21 Задачник-практикум по информатике. — СПб.: БХВ-Петербург, 2002. — 432 с.: ил.

ISBN 5-94157-186-0

Основу содержания книги составляют авторские задачи, вопросы, примеры и практикумы по информатике. Рассматриваются вопросы представления информации и ее кодирования, устройство и программное обеспечение компьютера, алгоритмизация и программирование на языке Бейсик, обработка текстовой, графической, числовой информации, мультимедийные технологии и компьютерные коммуникации. Каждая из рассматриваемых тем предваряется коротким теоретическим вступлением, поясняющим приведенные задачи и практические задания.

Для учащихся 7—11 классов и преподавателей школ

УДК 681.3.06(075.3)
ББК 32.973.26-018

Группа подготовки издания:

Главный редактор	*Екатерина Кондукова*
Зав. редакцией	*Анна Кузьмина*
Редактор	*Григорий Добин*
Компьютерная верстка	*Натальи Смирновой*
Корректор	*Наталия Першакова*
Дизайн обложки	*Игоря Цырульникова*
Зав. производством	*Николай Тверских*

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 21.08.02.
Формат 70×100 1/16. Печать офсетная. Усл. печ. л. 34,83.
Тираж 5000 экз. Заказ №
"БХВ-Петербург", 198005, Санкт-Петербург, Измайловский пр., 29.

Гигиеническое заключение на продукцию, товар № 77.99.02.953.Д.001537.03.02
от 13.03.2002 г. выдано Департаментом ГСЭН Минздрава России.

Отпечатано с готовых диапозитивов
в Академической типографии "Наука" РАН
199034, Санкт-Петербург, 9 линия, 12.

ISBN 5-94157-186-0

© Сафронов И. К., 2002
© Оформление, издательство "БХВ-Петербург", 2002

Содержание

Предисловие **1**

Часть I. Задачи, вопросы, практические задания **3**

Введение **5**

Глава 1. Информация и информационные процессы **8**

1.1. Понятие об информации и информационных процессах 8
1.2. Кодирование информации 12
1.3. Единицы измерения информации 25
1.4. Системы счисления 30
1.5. Основы логической алгебры 37

Глава 2. Компьютер **47**

2.1. История развития вычислительной техники 47
2.2. Персональные компьютеры. Классификация 56
2.2.1. Основные устройства персональных компьютеров 57
2.3.2. Основные характеристики IBM-совместимых компьютеров 58
2.3.3. Внешние (периферийные) устройства персонального компьютера 61
2.3. DOS и файловая система 71
2.3.1. Основные внутренние команды DOS 77
2.3.2. Основные внешние команды DOS (для версии MS-DOS 6.22) 79
2.4. Оболочка Norton Commander 85
2.5. Windows и программа Проводник 92
2.6. Инсталляция программ 102
2.7. Архиваторы 103
2.8. Антивирусные программы 105

Глава 3. Компьютерное моделирование **111**

3.1. Краткие сведения о моделях 111
3.2. Задачи 113
Задача 1. Прямоугольный треугольник 113
Задача 2. Обои 113
Задача 3. Расстановка мебели 114
Задача 4. Неотвратимость наказания 114

Задача 5. Бородино....117
Задача 6. Казино....118
Задача 7. Большая семья....119
Задача 8. Три учителя....119
Задача 9. Встреча на конгрессе....120
Задача 10. Кто с кем знаком?....120
Задача 11. Не так-то легко....121
Задача 12. Едем на экскурсию....121
Задача 13. Распределение грузов....122
Задача 14. Моделирование физических явлений....124
Задача 15. Волк кролику не товарищ. Экологическая модель....125
Задача 16. Перевод температур....125
Задача 17. Десять городов....125
Задача 18. Страны Африки. Структурная информационная модель....126
Задача 19. Взлет ракеты с поверхности планеты....126
Задача 20. Море, море... Крупнейшие зарубежные пассажирские суда XX века. Структурированная модель....127

Глава 4. Алгоритмизация и программирование....129

4.1. Алгоритмы. Их типы и свойства....129
4.1.1. Виды алгоритмов....130
4.1.2. Свойства алгоритмов....130
4.2. Переменные и работа с ними....142
4.2.1. Переменные....142
4.2.2. Арифметика в программировании....142
4.2.3. Оператор присваивания....146
4.2.4. Стандартные функции Бейсика....149
4.3. Алгоритмическое программирование....152
4.3.1. Линейные алгоритмы....152
4.3.2. Разветвляющийся алгоритм....173
4.3.3. Циклический алгоритм....189
4.3.4. Символы и строки....217
4.3.5. Массивы....220
4.3.6. Подпрограммы....229
4.3.7. Файлы последовательного и прямого доступа....232
4.4. Задания повышенной трудности, интегрированные, азартные....236
4.4.1. Угадайка (математика и программирование)....236
4.4.2. Анаграммы (русский язык и программирование)....236
4.4.3. Стрельба из пушки (физика, математика и программирование)....237
4.4.4. Царь-пушка (математика, физика, экономика, история, русский язык и программирование)....238
4.4.5. Кинотеатр "Кристалл-Палас" (математика, экономика и программирование)....240
4.4.6. Тараканьи бега (математика, дизайн и программирование)....241
4.4.7. Тесты (психология, русский язык и программирование)....243

Глава 5. Информационные технологии 244

5.1. Обработка текстовой информации. Первое знакомство с MS Word 244
5.1.1. Создание, сохранение документов. Открытие уже существующих документов. Переход от одного документа к другому через меню *Окно* 245
5.1.2. Набор и редактирование текста 247
5.1.3. Настройка параметров страницы 255
5.1.4. Шрифтовое форматирование 256
5.1.5. Абзацное выравнивание 259
5.1.6. Оформление списков 265
5.1.7. Настройка абзацев по образцу 271
5.1.8. Копирование и перенос фрагментов текста, одновременная работа с несколькими окнами 271
5.1.9. Автозамена 274
5.1.10. Форматирование текста в колонки 274
5.1.11. Работа с таблицами 277
5.1.12. Колонтитулы 284
5.1.13. Поиск и замена элементов текста 286
5.1.14. Написание математических формул 287
5.1.15. Оформление текста графическими объектами 289
5.1.16. Создание фигурного текста 298
5.1.17. Связывание документов гиперссылками 299
5.1.18. Добавление закладки 301
5.1.19. Создание простейших Web-страниц 302
5.1.20. Создание документов по шаблону 303
5.2. Обработка числовой информации MS Excel 306
5.2.1. Ввод и редактирование данных. Использование простейших формул. Относительная и абсолютная адресация 307
5.2.2. Использование функции *ЕСЛИ* 311
5.2.3. Функция *ЕСЛИ*. Построение графиков и диаграмм 314
5.2.4. Комплексные задания 317
5.2.5. Макросы в Excel 325
5.3. Обработка структурированной информации (базы данных) 335
5.3.1. Основные понятия баз данных 335
5.4. Обработка графической информации 347

Глава 6. Мультимедийные технологии 357

6.1. Программы распознавания текста 357
6.2. Программы-переводчики 358
6.3. Работа с цифровой камерой 360
6.4. Компьютерные презентации 360
6.5. Создание маленького сайта 367
6.6. Поиск в Интернете 368
6.7. Электронная почта 372
6.8. Мультимедийные проекты 377
6.9. Сетевые игры 379

Часть II. Решения избранных задач 381

Глава 7. Ответы и решения 383

Глава 1 383
Вопрос 1 383
Вопрос 2 383
Вопрос 3 384
Вопрос 4 384
Вопрос 5 384
Тест 1 384
Практикум 2 384
Практикум 5 384
Практикум 7 385
Практикум 8 385
Практикум 13 387
Практикум 19 387
Глава 2 390
Тест 2 390
Тест 3 390
Тест 5 390
Глава 4 391
Свойства алгоритмов 391
Арифметика в программировании 397
Оператор присваивания 397
Стандартные функции Бейсика 397
Линейные алгоритмы 398
Графика в Бейсике 399
Разветвляющийся алгоритм 401
Условный переход 401
Циклический алгоритм 403
Символы и строки 409
Массивы 412
Простейшие сортировки 418
Глава 6 420
Тест 2 420
Практикум 7 420
Практикум 8 421
Тест 3 422
Тест 4 422

Заключение 423

Список дополнительной литературы 424

Предисловие

Представляемая читателю книга является плодом многолетней работы автора по преподаванию в школе загадочного предмета "Информатика". Будучи не обременен (тут есть свои и плюсы, и минусы) специальным педагогическим образованием, автор к сегодняшнему дню понял, что, во-первых, информатика нужна и, более того, жизненно необходима, а, во-вторых, что в основном преподают ее замечательные специалисты, с огромным багажом работы как с людьми, так и с компьютерами. Теоретические основы курса каждый из таких специалистов может изложить практически не задумываясь, с одними только ему присущими подходами и ви́дениями. А вот наполненности курса разнообразными, многоуровневыми, интегрированными заданиями, вопросами и практикумами, на взгляд автора, отнюдь не наблюдается.

Все выходящие из печати издания такого свойства мгновенно раскупаются, и хочется надеяться, что этот скромный вклад в дело практического овладения школьниками курсом информатики не будет лишним.

В начале 1990-х годов наблюдалась тенденция к эдакой педагогической вольнице, когда фактически каждый педагог мог учить информатике по своей собственной программе, иногда руководствуясь опытом, иногда интуицией, а иногда и авантюризмом. Сейчас, слава богу, пришло осознание того, что в преподавании любой дисциплины нужна система, некая общая база, отталкиваясь от которой, можно внедрять и профильное обучение. И вот принят государственный стандарт. Плох он или хорош, обсуждать можно долго, но он есть — и надо его исполнять. А наша книга попробует вам в этом помочь.

Задачник охватывает все разделы федерального стандарта по информатике. В первой части шесть глав. Каждую главу предваряет очень краткий теоретический экскурс, позволяющий пояснить основные направления и понятия, договориться об общих взглядах на излагаемый предмет. Затем предлагаются практикумы, интегрированные задания и контрольные вопросы для проверки знаний и умений.

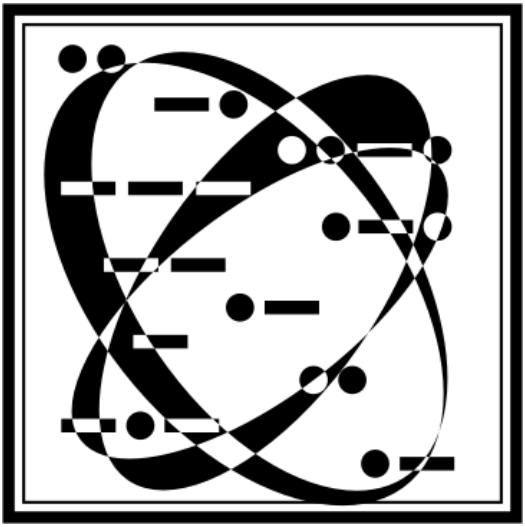
Для тех задач, которые требуют загрузки готовых файлов, преподаватель должен их предварительно создать по приведенным в книге образцам.

В пределах каждой главы нумерация вопросов, задач, практикумов и примеров — сквозная.

Во второй части предлагаются избранные решения некоторых задач. (Тут у меня как у автора, с одной стороны, и преподавателя, с другой, возникают противоречивые чувства — вроде хочется дать все решения, но, вспоминая себя школьником, понимаешь, что от соблазна списать очень трудно удержаться.)

И еще одно — автор будет рад любым отзывам коллег и школьников. Пишите на электронный адрес: **old_matros@mail.ru**.

Удачи!



ЧАСТЬ I

Задачи, вопросы, практические задания

Введение

Прежде чем перейти собственно к задачам, вопросам, примерам и практикумам, хочется поделиться несколькими, на наш взгляд, полезными советами по проведению занятий.

Во-первых, для организации внимания учащихся, повышения их собранности и тренировки работы головного мозга в начале урока всегда предлагается традиционная головоломка "Телефон" (всего пять минут, но умственная гимнастика — это так важно!). Суть заключается в следующем: берется реально существующий семизначный номер телефона и предлагается за пять минут, используя цифры заданного телефона, с помощью знаков арифметических действий — сложения, вычитания, умножения и деления, круглых скобок, группировки подряд идущих цифр, извлечения квадратного корня, извлечения корней другой степени, возведения в степень и т. п., добиться того, чтобы после выполнения действий получилось ровно 100. Главное — чтобы не был нарушен порядок следования цифр в телефоне, и чтобы все цифры были использованы.

Например, телефон 212-85-06.

Возможные варианты решений:

- $(21 : (-2 + 8)) \times 50 \times 3 = 100$;
- $21 \times (2 - 8 + 50 + 6) = 100$;
- $(-2 + 12)^{(8 - 5 \times 0 - 6)} = 100$.

За каждое решение учащийся получает плюсик или пятерку (в зависимости от широты души и настроения учителя), которые в течение четверти, полугодия или года могут как-то сказаться на итоговой оценке. Кроме того, решения оцениваются как в фигурном катании — и за технику, и за артистизм (в этом смысле высшую оценку получит третье решение).

Надо сказать, что автор находит время проводить по этому, с позволения сказать виду спорта, турниры, в которых с удовольствием участвуют многие ученики разных классов. Телефоны берутся обычно из рекламных листовок и плакатов в вагонах метро, по дороге на работу и обратно. Любопытно, что пока не нашлось ни одного реально существующего телефона, который дети не смогли бы "решить". И еще. Когда особо одаренным уже этого мало, количество цифр в телефоне уменьшается на одну.

Ну, и еще один пример.

Телефон 567-20-51.

Возможные варианты решений:

- $(56 - 7) \times 2 + 0 + \sqrt{5-1} = 100$;
- $(-5 + 6)^7 + 20 \times 5 - 1 = 100$;
- $(-5 + 6) \times (7^2 + 0 + 51) = 100$.

Вообще, игровые моменты довольно часто используются как в подаче нового материала, так и в проверке уже усвоенных знаний (подробнее об этом будет рассказано в следующей книге автора "Развлечения в процессе обучения").

Это могут быть вопросы в стиле "Верите ли вы, что...", например:

Верите ли вы, что

- Винчестер — это фамилия основателя американской фирмы по выпуску стрелкового оружия? (Да).
- Юзер и ламер — это одно и то же? (Нет).
- Двоичная система представления информации в компьютере изначально выбрана из соображений секретности, т. к. первые компьютеры были предназначены для Министерства обороны? (Нет).

Это может быть так называемое "эрудит-лото", когда к вопросу предлагается 4 варианта ответов. Вы скажете, что это известная тестовая форма проверки знаний, но отличие "эрудит-лото" в том, что один из вариантов ответов правдоподобен, но смешон, тем самым давая возможность снятия стресса от прохождения теста и поднятия настроения.

Возможный вариант вопроса.

Оставив пост главы корпорации Microsoft, Билл Гейтс получил титул:

а) великий компьютерный гуру;

б) главный открыватель окон;

в) главный разработчик программ;

г) отец программистов.

Правильный ответ — в).

Это может быть разновидность тестовой проверки в форме игры "Кто хочет стать миллионером?", когда вопросы усложняются, их будет, например, пять, каждый соответствует одному баллу оценки, второй и четвертый баллы являются несгораемыми. В результате ставится та оценка, на сколько баллов был оценен последний правильно отвеченный вопрос. В любой момент можно остановиться, получить свою оценку или рисковать, чтобы получить лучшую. Допускается, как и в телевизионном варианте, вводить подсказки. Можно увеличить количество вопросов.

Например:

Вопрос на единицу.

"Голубым гигантом" в компьютерном мире называют:

а) оболочку Norton Commander;

б) ЭВМ ЕС-1041;

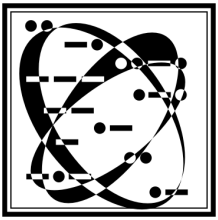
в) фирму IBM;

г) основателя корпорации Microsoft Билла Гейтса.

Правильный ответ — в).

Вообще, ученики, как любые дети, любят играть. Особенно во всякие игры, хоть отдаленно напоминающие телевизионные, будь то "Что? Где? Когда?", "Брэйн-Ринг", "Своя игра" или "Слабое звено". Поэтому вспомним старый добрый лозунг: "Твори, выдумывай, пробуй!".

Глава 1



Информация и информационные процессы

1.1. Понятие об информации и информационных процессах

В словаре иностранных слов понятию "Информатика" дается следующее объяснение: *"Отрасль знаний, изучающая общие свойства и структуру информации, а также закономерности и принципы ее создания, преобразования, накопления, передачи и использования в различных областях человеческой деятельности"*.

Многие школьники ошибочно, на наш взгляд, полагают, что информатика — наука о компьютерах, и заниматься на информатике нужно исключительно программированием. Мы же полностью согласны с определением словаря иностранных слов, данным выше, с одним только маленьким дополнением в самом конце *"при помощи компьютера"*. То есть, в нашем понимании, компьютер в информатике — суть инструмент, помогающий информацию создавать, преобразовывать, накапливать, передавать и использовать.

А программисты у нас, как известно, самые лучшие в мире. Только надо их стране не так много, и всем изучать программирование вовсе не требуется. Желательно знать лишь основы алгоритмизации для создания у себя в голове системного подхода к жизненным проблемам и понять их на основе какого-нибудь несложного языка программирования. А вот пользоваться компьютером для всех вышеперечисленных задач надо всем! Это входит в понятие культуры современного человека.

Лирическое отступление закончилось.

В том же словаре понятие "информация" трактуется так: "от латинского *information* — разъяснение, изложение. 1) сообщение о чем-либо; 2) сведения, являющиеся объектом хранения, переработки и передачи; 3) в математике, кибернетике, информатике — количественная мера устранения неопределенности, мера организации системы.

Последняя фраза очень интересна, и простым языком ее следовало бы пояснить так: чем больше мы имеем информации о каком-либо процессе или объекте, тем меньше неопределенность, непознанность этого процесса или объекта, тем легче моделировать и прогнозировать данный процесс или объект.

Итак, абстрагируясь пока от компьютера, исходя только из здравого смысла, давайте попробуем ответить на следующие вопросы.

Вопрос 1. Перечислите виды информации (по способам получения ее человеком, его органами чувств).

Вопрос 2. А знаете ли вы основные свойства информации?

Вопрос 3. Какие вы знаете носители информации?

Вопрос 4. Каковы основные информационные процессы?

Вопрос 5. Какие информационные "революции" вы бы выделили в истории человечества?

Если не получилось ответить, загляните в конец книги, в *часть II.* "Решение избранных задач". Проанализируйте данные там ответы и попытайтесь пройти следующий тест.

Тест 1. Информатика, информация и информационные процессы

Замечание

В процессе выполнения этого теста и последующего его разбора дебаты с учениками возникают всегда и практически по каждому из вопросов. Это нормально — в споре, как известно, рождается истина. А информация, в отличие от алгоритма, вовсе не обязана быть однозначной. Поэтому в каждом вопросе возможен не один правильный ответ.

1. Понятие информации в информатике означает:

   а) сведения о чем-либо;

   б) уменьшение неопределенности;

   в) знания;

   г) компьютерную программу;

   д) компьютерные сообщения.

2. Что является наиболее общим между папирусом, берестяной грамотой, книгой и дискетой?

   а) материал, из которого они изготовлены;

   б) способ производства;

   в) хранение информации;

г) стоимость;

д) форма.

3. Компьютер — универсальное средство для работы в первую очередь с:

а) числами;

б) информацией;

в) символами;

г) программами;

д) языками программирования.

4. Информатика, в основном, изучает:

а) методы обработки информации;

б) компьютер;

в) графический редактор;

г) законы накопления информации;

д) текстовую информацию.

5. Что из нижеперечисленного не является основным свойством информации:

а) достоверность;

б) однозначность;

в) полнота;

г) ценность, актуальность;

д) ясность, понятность.

6. Работником информационной сферы деятельности является:

а) фотомодель;

б) телохранитель;

в) журналист;

г) актер;

д) спортсмен.

7. Оперативная информация может храниться:

а) в книгах;

б) на видеокассетах;

в) на компакт-дисках;

г) в памяти человека;

д) в газетах и журналах.

8. Какие образы информации человек может хранить на внешних носителях?
   а) вкусовые образы;
   б) изображения и звук;
   в) осязательные образы;
   г) обонятельные образы;
   д) текст, написанный на каком-либо языке.
9. Укажите, в каком виде представлена информация в следующих примерах?
   а) роман "Преступление и наказание";
   б) аромат духов;
   в) карта местности;
   г) радиопередача;
   д) телевизионный рекламный ролик.
10. Укажите информационный носитель в каждом из следующих примеров:
   а) журнал;
   б) дискета с играми;
   в) аудиокассета с песнями;
   г) компакт-диск;
   д) кинофильм.
11. Поставьте в соответствие приведенным примерам (табл. 1.1) тип информационного процесса из следующих возможных типов:
   а) хранение;
   б) обработка;
   в) ввод;
   г) вывод;
   д) передача.

Таблица 1.1. *Примеры типов информационного процесса*

Пример	**Тип информационного процесса**
1. Мама дотронулась до Ксюшиного лба рукой и поняла, что дочка заболела	
2. Дорожный знак предупреждает водителя о ведущихся впереди работах	
3. Фотографии напоминали о пребывании в детском саду	

Таблица 1.1 *(окончание)*

Пример	Тип информационного процесса
4. Учитель сообщил ребятам интересные сведения об обитателях океана	
5. Сигнал маяка предупреждал капитана об опасности — рядом были рифы	
6. Славин папа записал выпускной вечер на видеокассету	
7. По срезу дерева можно определить, засушливым или дождливым был каждый год его жизни	
8. Володя представил на компьютере презентацию своей новой игровой программы	
9. Ирина набрала реферат на компьютере	

12. В следующих примерах укажите источник и приемник информации:
    а) вы читаете учебник;
    б) звенит будильник;
    в) вы смотрите футбольный матч по телевизору;
    г) вы играете в компьютерную игру;
    д) ученый рассматривает в телескоп звезду.

1.2. Кодирование информации

Кодирование и шифрование информации для автора этих строк с детства было чем-то очень увлекательным и интересным. До сих пор жива тетрадь (теперь она уже у сына), в которой содержится с полсотни известных и придуманных автором шифров, с помощью которых он и его дворовые друзья переписывались и разрабатывали свои детские военные операции.

Первый и самый простой пример кодирования информации — перевод на другой язык. Разрушил Бог Вавилонскую башню, и не стало у нас общего языка, видят люди разных народов один и тот же стол, а называют его поразному и друг друга не понимают.

Во время Второй мировой войны американцы для секретных переговоров привлекли индейцев одного очень малочисленного племени, которых оставалось в США не более двухсот человек, те просто говорили в эфире, ничего не скрывая, на своем языке, и никакие лучшие дешифраторы фашистской Германии и Японии ничего не могли понять!

А английский писатель Конан Дойль придумал пляшущих человечков (рис. 1.1).

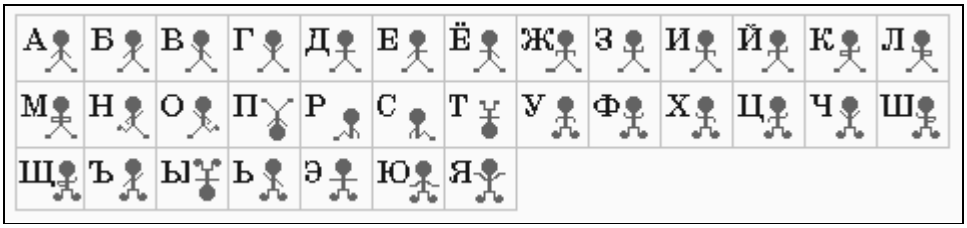


Рис. 1.1. Пляшущие человечки

Практикум 1. Расшифровка загадки пляшущих человечков

Расшифруйте загадку (рис. 1.2).

Рис. 1.2. Загадка

Леонардо да Винчи (рис. 1.3) писал свои записки в зеркальном отражении.

Рис. 1.3. Леонардо да Винчи

А американский художник Морзе (рис. 1.4) придумал свою знаменитую азбуку (рис. 1.5), чей знаменитый сигнал SOS (три точки, три тире, три точки) спас немало человеческих жизней.

Рис. 1.4. Сэмюэл Морзе

А • —	Л • — • •	Ц — • — •
Б — •••	М • — —	Ч — — — •
В • — —	Н — •	Ш — — — —
Г — — •	О — — —	Щ — — • —
Д — • •	П • — — •	Ъ • — — • — •
Е •	Р • — •	Ы — • — —
Ж • • • —	С • • •	Ь — • • —
З — — • •	Т —	Э • • — • •
И • •	У • • —	Ю • • — —
Й • — — —	Ф • • — •	Я • — • —
К — • —	Х • • • •	

Рис. 1.5. Азбука Морзе

Практикум 2. Расшифруйте следующее сообщение на языке азбуки Морзе

– –/• – / •••• /• – • / – – – /•••/ – • – – / • – – – /
– • – / • – •• / – – – / –

Практикум 3. Зашифруйте при помощи азбуки Морзе свои имя и фамилию

Особо остро стояла проблема передачи информации на море, между кораблями и сушей, особенно в период военных действий. Ведь правы были древние, говоря: "Кто владеет информацией, тот владеет миром".

Тогда были изобретены два способа.

Первый — *флажковая азбука*, когда каждый флаг обозначал какую-либо букву, и вывешивая их между мачтами, можно было передавать сообщения. Вот эта азбука (рис. 1.6).

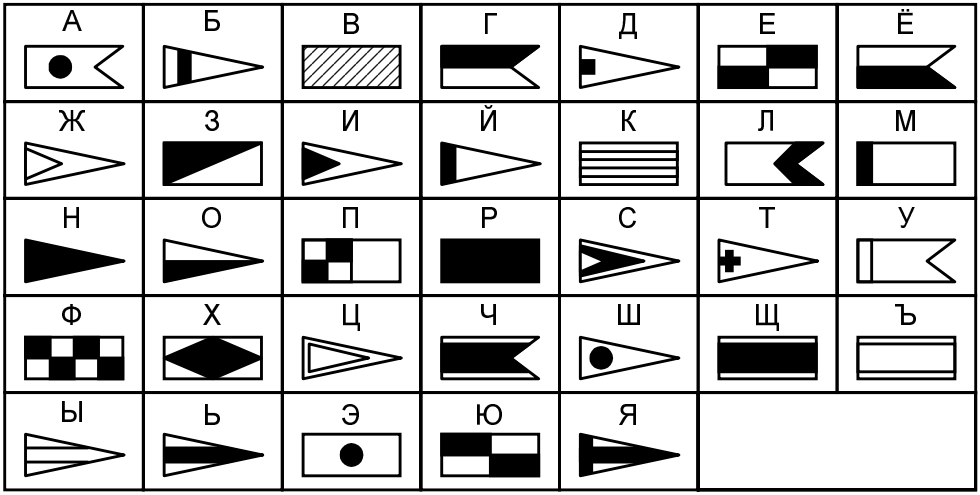


Рис. 1.6. Флажковая азбука

Практикум 4. Что за слово здесь зашифровано?

Расшифруйте слово (рис. 1.7).

Рис. 1.7. Слово

А есть еще и *семафорная азбука*. В ней используются два флажка, а разнообразие комбинаций достигается за счет изменения положения рук сигнальщика.

Вокруг нас масса закодированной информации. Цифры, ноты, иконки на рабочем столе Windows, знаки дорожного движения, химические элементы — ужас какой, сколько современный человек должен держать в голове! Одни *штрих-коды* на товарах чего стоят. На них можно остановиться немножко подробнее. Информация полезная.

Прежде всего развеем несколько мифов, которые сложились вокруг штрих-кодов:

- это — не гарантия подлинности;
- там нет информации об экологической чистоте продукта;
- там нет также указания на цену.

А что же там есть?

В свое время оптовые торговцы и производители столкнулись с серьезной проблемой: товаров много (средний универмаг, например, оперирует с десятью тысячами наименований), и к каждому — длинный сертификат, в котором расписано, где он сделан, на какой фирме, сколько весит, какие габариты и т. д. Поэтому решили всю эту информацию "утрамбовывать" в виде штрихов, которые потом считываются специальным лазерным кассовым аппаратом. Это значительно облегчило складские операции. Впоследствии, в конце 80-х, сложилась мировая система штрих-кодирования, которую обеспечивает ассоциация EAN. Создано два стандарта: американский UPC с 12 цифрами и европейский с 13. В России штрих-коды, начинающиеся с цифр 46, присваивает ассоциация ЮНИСКАН, которая является полноправным членом EAN. Такие ассоциации есть и во всех других странах.

Посмотрите на рис. 1.8.

Рис. 1.8. Штрих-код

Перед вами код 4009993134132, состоящий из 13 цифр (система EAN-13). Первые две цифры (40) — страна, следующие пять (09993) — код предприятия, другие пять (13413) — зашифрованные потребительские свойства (причем первая цифра из этих пяти скрывает имя товара, вторая — особенности его потребительских свойств, третья — массу, четвертая — состав, пятая — цвет) и последняя цифра — контрольная. Для чего она нужна — узнаем позднее.

Те, кто бывал в крупных супермаркетах, видели, как кассир, делая расчет, просто проносит товар, повернув его штрих-кодом вниз, над кассовым аппаратом, и на экране аппарата мгновенно выскакивает цена. Поэтому думают, будто в штрих-коде заложена цена. На самом деле, конечно, в самом коде данных о цене нет. Ведь один и тот же товар в разных магазинах продается по разным ценам. Но когда товар поступает в данный супермаркет, принимается решение, по какой цене его продавать, и в компьютер заносятся данные о его штрих-коде и цене, так что кассовый аппарат распознает, с каким товаром он имеет дело, и высвечивает заложенную стоимость.

Рядовому покупателю с помощью первых двух-трех цифр кода легко определить, в какой стране сделан товар (табл. 1.2). Но бывают недоразумения. Например, видят, что, согласно штрих-коду, товар сделан на Тайване, а буквами написано: Made in Germany. Покупатель начинает возмущаться: "Подделка!" Может, и подделка. А может эта фирма, хоть и производит товар в Германии, но штрих-код по каким-то причинам зарегистрировала на Тайване. Или на Тайване у нее филиал. Или она работает по тайваньской лицензии. Или это совместное германо-китайское предприятие. Или... Причин может быть много. Например, импортные йогурты, продающиеся в России, имеют российские штрих-коды. Западные производители пошли на это, чтобы пользоваться определенными льготами и легче продавать в России свой товар.

Некоторых покупателей смущает, если на штрих-коде имеются только собственно штрихи, а цифр нет. Это также не признак подделки. Для лазерного кассового аппарата цифры вообще не имеют значения, и, если места на товаре мало, их не ставят.

Не нужно пытать продавца и в том случае, если штрих-код узкий, или короткий, или вообще "какой-то не такой". Обычно так бывает на мелких по размеру товарах. ЮНИСКАН разрешает производителям таковых использовать сокращенные варианты кодировки.

Слышим, как читатели восклицают: получается, нам можно на штрих-код вообще не смотреть, там нет полезной для нас информации?

Это не так — есть способ, хотя и трудоемкий, узнать, поддельный ли товар. Помните, мы говорили, что последняя цифра — контрольная?

Итак, если нам крайне важно узнать, с чем мы имеем дело, нужно произвести следующие арифметические действия:

1. Сложить цифры, стоящие на четных позициях: в нашем примере $6 + 1 + 2 + 1 + 0 + 1 = 11$.
2. Сумму умножить на 3: $11 \times 3 = 33$.
3. Сложить цифры, стоящие на нечетных позициях, не считая контрольной цифры: $5 + 0 + 7 + 1 + 1 + 0 = 14$.
4. Сложить то, что получилось в результате второго и третьего действий: $33 + 14 = 47$.
5. От результата отбросить первую цифру. Получится 7.
6. И отнять от 10 то, что получилось в пятом пункте: $10 - 7 = 3$.

Этот результат должен совпадать с контрольной цифрой. Если нет — товар поддельный.

Метод, конечно, сложный. Однако если вы покупаете дорогую вещь или есть сомнения, доброкачественный ли продукт питания перед вами — имеет смысл произвести эти в общем-то элементарные процедуры. Ведь отравление и разочарование от неудачной покупки обойдутся гораздо дороже.

Таблица 1.2. *Штрих-коды некоторых стран*

Штрих-код	Страна	Штрих-код	Страна
00–09	США и Канада	539	Ирландия
30–37	Франция	54	Бельгия и Люксембург
380	Болгария	560	Португалия
383	Словения	569	Исландия
385	Хорватия	57	Дания
400–440	Германия	590	Польша
460–469	Россия, СНГ	599	Венгрия
4605	Латвия	600–601	ЮАР
471	Тайвань	619	Тунис
489	Гонконг	64	Финляндия
45, 49	Япония	690	Китай
50	Великобритания	70	Норвегия
520	Греция	729	Израиль
529	Кипр	73	Швеция
535	Мальта	740–745	Гватемала, Сальвадор, Гондурас, Никарагуа, Коста-Рика, Панама

Таблица 1.2 *(окончание)*

Штрих-код	Страна	Штрих-код	Страна
750	Мексика	84	Испания
759	Венесуэла	850	Куба
76	Швейцария	859	Чехия и Словакия
770	Колумбия	860	Югославия
773	Уругвай	869	Турция
775	Перу	87	Нидерланды
779	Аргентина	880	Южная Корея
780	Чили	885	Таиланд
786	Эквадор	888	Сингапур
789	Бразилия	90–91	Новая Зеландия
80–83	Италия	955	Малайзия

Ну, это было хоть и полезное, но отступление, а теперь вернемся к кодированию и выполним несколько упражнений посложнее, на декодировку.

Практикум 5. День святого Валентина

Рис. 1.9. День святого Валентина

Расшифруйте четверостишие русского поэта Олега Григорьева.

У вас имеется таблица кодов — аналог азбуки Морзе (рис. 1.10) и таблица с шифрованным текстом (рис. 1.11).

А	Б	В	Г	Д	Е	Ж	З
12	2111	122	221	211	1	1112	2211

И	Й	К	Л	М	Н	О	П
11	1222	212	1211	22	21	222	1221

Р	С	Т	У	Ф	Х	Ц	Ч
121	111	2	112	1121	1111	2121	2221

Ш	Щ	Ы	Ю	Ь,Ъ	Э	Я
2222	2212	2122	1122	2112	11211	1212

Рис. 1.10. Таблица кодов

1122111112121212222211221111211211 221111	32144334312244	41423
1122221112121222121112122222221121 21211111	23223233123433411	17252
2221121122221111222222222121222121 2111211121211122212111212211212	3513343222324231243424 4	324356
121211211112112-2222111212121112211111122222222222 121211211112112-2222111212121112211111122222222222	24432311143443433	142217

Рис. 1.11. Шифровка

В первом столбце шифровки — непосредственно зашифрованный текст, во втором столбце — количество символов, соответствующих каждой букве, в третьем — вы сами должны понять, что за цифры (по аналогии со вторым), ну и расшифровать сами слова, в которых перепутан порядок букв (так называемые *анаграммы*).

Практикум 6. Двузначные номера

Каждую букву исходного сообщения заменили ее двузначным порядковым номером в русском алфавите согласно рис. 1.12.

Полученную цифровую последовательность разбили (справа налево) на трехзначные цифровые группы без пересечений и пропусков. Затем каждое из полученных трехзначных чисел умножили на 77 и оставили только три последние цифры произведения. В результате получилась следующая после-

довательность цифр: 3175644049700176775505478503 55. Восстановите исходное сообщение.

А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й	К	Л	М	Н	О	П
01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17
Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	
18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	

Рис. 1.12. Двузначные коды

Практикум 7. Телетайп

Для проверки правильности работы телетайпа, печатающего буквами русского алфавита "АБВГДЕЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ", передан набор из 9 слов, содержащий все 33 буквы алфавита. В результате неисправности телетайпа на приемном конце получены слова: ГЪЙ АЭЕ БПРК ЕЖЩЮ НМЬЧ СЫЛЗ ШДУ ЦХОТ ЯФВИ. Восстановите исходный текст, если известно, что характер неисправности таков, что каждая буква заменяется буквой, отстоящей от нее в указанном алфавите не дальше, чем на две буквы. Например, буква Б может перейти в одну из букв А, Б, В, Г.

Ну, а теперь непосредственно к компьютерному кодированию информации. Всю получаемую информацию компьютер оцифровывает, используя для этого всего две цифры — 1 и 0. Почему так произошло, мы узнаем в *разд. 1.4*. А сейчас надо понять, сколько же символов мы можем закодировать с помощью всего двух цифр.

Если для кодирования применить одноразрядную схему, то получится только два символа (рис. 1.13).

А	0
Б	1

Рис.1.13. Одноразрядная схема

Если двухразрядную, то можно закодировать уже четыре символа (рис. 1.14).

А	00
Б	01
В	10
Г	11

Рис. 1.14. Двухразрядная схема

Если трехразрядную... Стоп!

Вопрос 6. А сколько же символов будет возможно закодировать?

Обычно застигнутые врасплох учащиеся говорят 9. Попросим кого-нибудь на доске выписать возможные комбинации. Оказывается, что их получается только 8 (рис. 1.15), еще некоторое время продолжаются поиски девятой комбинации, потом приходится разрешить ребятам поискать ее дома, а если вдруг повезет найти, то можно готовиться к получению Нобелевской премии.

А	000
Б	001
В	010
Г	011
Д	100
Е	101
Ж	110
З	111

Рис. 1.15. Трехразрядная схема

Более внимательные на этом месте соображают, что можно вывести общую формулу количества возможных кодируемых двумя цифрами символов в зависимости от разрядности:

$$N = 2^k,$$

где k — число разрядов.

Вопрос 7. А сколько же тогда нужно разрядов, чтобы закодировать 32 русские буквы при помощи 1 и 0?

Собственно, на этом и основаны современные таблицы кодировок. Одно время в СССР была принята кодировка КОИ-7, которая могла закодировать в своих семи разрядах... Правильно, 128 символов. Оказалась она неудобна и уступила свое место уже в современной России кодировке КОИ-8. Кто был в Интернете, тот знает.

Но есть также широко и повсеместно используемая международная таблица кодов ASCII (American Standard Code for Information Interchange). Стандартная и расширенная таблицы кодов ASCII приведены на рис. 1.16 и 1.17 соответственно.

Те, кто сталкивался при работе в Интернете или с электронной почтой с нечитаемыми страницами или письмами, понимают, что необходим какой-то стандарт. И он почти найден. Это 16-разрядная кодировка универсального кода Unicode.

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00		☺	☻	♥	♦	♣	♠	•	◘	○	◙	♂	♀	♪	♫	☼
10	►	◄	↕	‼	¶	§	▬	↨	↑	↓	→	←	∟	↔	▲	▼
20		!	"	#	$	%	&	'	(	)	*	+	,	–	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	⌂
80	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
90	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	¢	£	¥	₧	ƒ
a0	á	í	ó	ú	ñ	Ñ	ª	º	¿	⌐	¬	½	¼	¡	«	»
b0	░	▒	▓	│	┤	╡	╢	╖	╕	╣	║	╗	╝	╜	╛	┐
c0	└	┴	┬	├	─	┼	╞	╟	╚	╔	╩	╦	╠	═	╬	╧
d0	╨	╤	╥	╙	╘	╒	╓	╫	╪	┘	┌	█	▄	▌	▐	▀
e0	α	ß	Γ	π	Σ	σ	µ	τ	Φ	Θ	Ω	δ	∞	φ	ε	∩
f0	≡	±	≥	≤	⌠	⌡	÷	≈	°	∙	·	√	ⁿ	²	■	

Рис. 1.16. Таблица кодов ASCII (стандартная)

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00		☺	☻	♥	♦	♣	♠	•	◘	○	◙	♂	♀	♪	♫	☼
10	►	◄	↕	‼	¶	§	▬	↨	↑	↓	→	←	∟	↔	▲	▼
20		!	"	#	$	%	&	'	(	)	*	+	,	–	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	⌂
80	А	Б	В	Г	Д	Е	Ж	Ш	И	Й	К	Л	М	Н	О	П
90	Р	С	Т	У	Ф	Х	Ц	Ч	с	Щ	Ъ	Ы	Ь	Э	Ю	Я
a0	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
b0	░	▒	▓	│	┤	╡	╢	╖	╕	╣	║	╗	╝	╜	╛	┐
c0	└	┴	┬	├	─	┼	╞	╟	╚	╔	╩	╦	╠	═	╬	╧
d0	╨	╤	╥	╙	╘	╒	╓	╫	╪	┘	┌	█	▄	▌	▐	▀
e0	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
f0	Ё	ё	Є	є	Ї	ї	Ў	ў	°	∙	·	√	№	¤	■	

Рис. 1.17. Таблица кодов ASCII (расширенная)

Вопрос 8. А сколько же такой код позволит закодировать символов?

Замечание

Да, кажется, здесь и китайцам с их иероглифами места хватит, да еще и японцам останется.

Стандарты Unicode и ISO 10646. До недавнего времени Unicode был одним из двух проектов международных стандартов кодировки наряду с другим, известным как ISO 10646. К счастью, недавнее соглашение между Консорциумом Unicode и Комитетом ISO 10646 произвело слияние двух проектов: Unicode теперь является собственным подмножеством набора знаков ISO 10646.

Основные коды стандарта Unicode составляют первые 65 536 кодов стандарта ISO 10646 и содержат все знаки, в настоящее время определенные ISO 10646. Остальные коды ISO 10646 остаются незаполненными и зарезервированы для будущего расширения. Чтобы отразить знаки, имеющиеся в ISO 10646, стандарт Unicode включил более чем 3600 новых китайских, японских и корейских знаков и более чем 1000 других знаков.

Слияние стандартов Unicode и ISO 10646 зафиксировало только один развиваемый проект международного стандарта кодировок и остановило сражение между стандартами, что лишь на пользу конечным потребителям.

Какие знаки включает Unicode? Стандарт Unicode определяет коды для знаков, применяемых во всех современных письменных языках. Это и латинский алфавит для английского языка, кириллический алфавит для русского и других славянских языков, греческий, иврит и арабский алфавиты, другие алфавиты, используемые в странах Европы, Африки, Индокитая и Азии.

Unicode включает также алфавиты типа японского kana, корейского hangul и китайского bopomofo. Самая большая часть стандарта Unicode посвящена тысячам обьединенных знаков для китайских, японских и корейских иероглифов.

Unicode включает много наборов символов с кодами знаков пунктуации, математических и технических символов, стрелок и др. Это обеспечивает наличие кодов для диакритов, которые используются как модификации знаков типа тильды и появляются в соединении с другими знаками. Всего Unicode обеспечивает коды для более чем 29 000 знаков мировых алфавитов, наборов иероглифов и символов.

Unicode содержит еще более чем 29 000 неиспользованных кодов для расширения, позволяющих включить в стандарт новые знаки. В будущем это даст возможность использовать исторические знаки типа египетских иероглифов и возможные расширения существующих алфавитов и/или наборов символов. Кроме того, Unicode резервирует более чем 6000 кодов для частного применения, которые разработчики программного обеспечения и аппаратных средств могут использовать для собственных знаков и символов.

Принципы построения Unicode. Для более легкого внедрения Unicode в качестве всемирного знакового стандарта для кодирования текста, в нем реализованы следующие принципы:

- в Unicode используются фиксированные 16-битные коды для знаков и нет зависимости от состояний или способов кодировки специальных знаков;
- Unicode включает наборы знаков многих существующих стандартов: например, Latin-1 как свои первые 256 знаков; кроме того, он включает репертуар знаков других общих, национальных и международных стандартов;
- Unicode использует унификацию Хана для объединения китайских, корейских и японских иероглифов;
- Unicode допускает создание отмеченных знаков: он кодирует каждый знак и диакрит или метку гласного отдельно, а также позволяет объединять знаки, чтобы создать отмеченные.

1.3. Единицы измерения информации

Все в этом мире измеряется, как ни грустно. Да нет, это шутка, конечно, не все! И мы продолжаем верить, что настоящие чувства: любовь, дружбу, преданность — не очень-то можно измерить.

Но мы говорим об информатике. Если вас послать в магазин с поручением купить килограмм яблок — то вы примерно представляете, сколько это будет. Если вам скажут, что сейчас сюда придет человек ростом 2 м 27 см, вы невольно посмотрите вверх.

А вот информация... Как быть с ней? Измеряется и она. А т. к. используются в ней только две цифры — 1 и 0, то и единицы ее измерения из нашей привычной системы счисления немного выпадают.

Но начнем с начала. Самая маленькая единица информации: 1 бит — один нолик или одна единичка. *Бит* (BIT) есть не что иное, как аббревиатура от BInary digiT (двоичная цифра), придуманная в 1946 году выдающимся американским ученым-статистиком Джоном Тьюки. Он был советником пяти президентов Соединенных Штатов. Тьюки избрал бит для обозначения одного двоичного разряда, способного принимать значение 0 или 1.

А вот с привычным *байтом*, оказывается, не все так просто. Если быть точным, то байтом (BYTE — это сокращение от BinarY Term, двоичный элемент) надлежит называть последовательность длиной от 8 до 10 битов! А для обычного восьмибитного байта есть специальный термин октет (Octet). Есть еще и экзотическая коротышка — четырехбитный ниббл (Nibble).

Тем не менее, мы будем называть байтом восьмибитовую последовательность символов, и, как нам уже ясно из *разд. 1.2*, именно одним байтом кодируется

один символ таблицы ASCII-кодов. Вот это важно: **1 символ = 1 байт**. Значения единиц информации в байтах приведены в табл. 1.3.

Таблица 1.3. *Значения единиц измерения информации*

Название единицы измерения	Численная величина в байтах	Точное количество байтов
Килобайт	2^{10}	1024 байт
Мегабайт	2^{20}	1024 килобайт 1 048 576 байт
Гигабайт	2^{30}	1024 мегабайт 1 073 741 824 байт
Терабайт	2^{40}	1024 гигабайт 1 099 511 627 776 байт
Петабайт	2^{50}	1024 терабайт 1 125 899 906 842 624 байт
Экзабайт	2^{60}	1024 петабайт 1 152 921 504 606 846 976 байт
Зеттабайт	2^{70}	1024 экзабайт 1 180 591 620 717 411 303 424 байт
Йоттабайт	2^{80}	1024 зеттабайт 1 208 92 81 614 629 174 706 176 байт

Этимология приставок не лишена интереса. Все началось в 1789 году, после Великой французской революции, с переходом на метрическую систему. Тогда появились первые приставки "кило" (kilo, k, 10^3) и "мега" (mega, m, 10^6), и их вполне хватало. В XX веке, после Второй мировой войны, этот ряд приставок был продолжен, и появились две следующие: "гига" (giga, G) и "тера" (tera, T) — соответственно для 10^9 и 10^{12}, т. е. миллиард и триллион. В 1975 году Генеральная конференция мер и весов обогатила мир еще двумя приставками: "пета" (peta, P, 10^{15}) и "экза" (exa, E, 10^{18}), т. е. квадриллион и квинтиллион. На этом процесс порождения приставок не остановился. В последнем издании Британской энциклопедии признаны права гражданства за "зетта" (zetta, Z, 10^{21}) и "йотта" (yotta, Y, 10^{24}), т. е. секстиллион и септиллион.

Названия "зетта" и "йотта" образованы от последней и предпоследней букв латинского алфавита.

Чтобы физически представить себе, а сколько это — 1 килобайт, представим себе заполненную текстом страницу формата А4. Набрана она стандартным шрифтом размера 12 без всяких прикрас. В среднем на такой странице помещается 60 строк по 80 символов в каждой. Соответственно, символов на странице $80 \times 60 = 4800$, а значит столько же и байтов, поделим приблизительно на 1000, чтобы получить килобайты — 4,8, т. е. приблизительно 5 килобайтов. Итак, страница обычного текста занимает примерно 5 Кбайт.

С графической информацией сложнее. Рассмотрим точечное изображение. Если точка только черная или только белая — она занимает в памяти 1 бит. Но это будет что-то типа негатива — контрастная черно-белая картинка. Два бита на точку позволят нам получить уже 4 оттенка, три бита — 8 и т. д. Поэтому, чем более изображение качественное и чем из большего числа цветов оно состоит, тем больше места оно занимает.

Вопрос 9. Компьютер имеет оперативную память 256 Мбайт. Сколько битов это составляет?

Вопрос 10. У племени чичевоков в алфавите 24 буквы и 8 цифр и больше ничего — ни знаков препинания, ни арифметических. Сколько разрядов минимум им необходимо для представления всех символов?

Если получается 5, то подумайте, что вы не учли?

Вопрос 11. В детской игре "Угадай число" первый участник загадал число в интервале от 1 до 64. Какое минимальное количество вопросов "Загаданное число больше числа ...?" должен задать второй участник, чтобы отгадать число?

Практикум 8. Кодирование и подсчет количества информации

Во всех вариантах надо проделать следующее.

- В *задании 1* необходимо, пользуясь таблицей кодов КОИ-8 (рис. 1.18), расшифровать три четырехзначных буквосочетания.

Код	Символ	Код	Символ	Код	Символ	Код	Символ
00100000	пробел	00110000	0	01000000	@	01010000	P
00100001	!	00110001	1	01000001	A	01010001	Q
00100010	“	00110010	2	01000010	B	01010010	R
00100011	#	00110011	3	01000011	C	01010011	S
00100100	$	00110100	4	01000100	D	01010100	T
00100101	%	00110101	5	01000101	E	01010101	U
00100110	&	00110110	6	01000110	F	01010110	V
00100111	‘	00110111	7	01000111	G	01010111	W
00101000	(	00111000	8	01001000	H	01011000	X
00101001	)	00111001	9	01001001	I	01011001	Y
00101010	*	00111010	:	01001010	J	01011010	Z
00101011	+	00111011	;	01001011	K	01011011	[
00101100	,	00111100	<	01001100	L	01011100	/
00101001	–	00111001	=	01001001	M	01011001	]
00101110	.	00111110	>	01001110	N	01011110	^
00101111	/	00111111	?	01001111	O	...	

Рис. 1.18. Фрагмент таблицы кодов КОИ-8

- В *задании 2* необходимо вычислить, какое количество символов можно закодировать единицами и нулями при известном количестве разрядов k.
- В *задании 3* необходимо вычислить информационный объем текстового документа в Мбайт (с точностью до сотых) и узнать, сколько дискет 3,5 дюйма объемом 1,44 Мбайт понадобится для переноса этого документа из дома в издательство (наличием электронной почты и архиваторов пренебрегаем).
- В *задании 4* необходимо вычислить информационный объем в Кбайт смешанного документа (с точностью до сотых).

Вариант 1

1.

   а) 0100001101001111010011110100 1100

   б) 0100111001000001010011010100 0101

   в) 0101101001000101010100100100 1111

2. $k = 3$.
3. Учебник "Основы информатики" — 224 страницы, 53 символа в строке, 45 строк на странице.
4. Черно-белая фотография разбита на точки, каждая из которых кодируется четырьмя битами информации. Каков информационный объем фотографии 9×12 см, если один квадратный сантиметр содержит 576 точек?

Вариант 2

1.

   а) 0100001101010001010100100100 0100

   б) 0101010001010010010001010100 0101

   в) 0100001101000001010010110100 0101

2. $k = 4$.
3. Пушкин А. С. Том 1 — 734 страницы, 26 символов в строке, 42 строки на странице.
4. Черно-белая фотография разбита на точки, каждая из которых кодируется двумя битами информации. Каков информационный объем фотографии 6×10 см, если один квадратный сантиметр содержит 576 точек?

Вариант 3

1.

   а) 0100010001010010010101010100 0111

   б) 0100011001001111010101010101 0010

   в) 0100111101010000010001010100 1110

2. $k = 5$.
3. Справочник "Желтые страницы" — 1392 страницы, 56 символов в строке, 60 строк на странице.
4. Черно-белая фотография разбита на точки, каждая из которых кодируется тремя битами информации. Каков информационный объем фотографии 6×8 см, если один квадратный сантиметр содержит 900 точек?

Вариант 4

1.

   а) 010001100100100101010110010001 01

   б) 01010000010101010101001001000101

   в) 01010110010010010100010101010111

2. $k = 6$.
3. Англо-русский словарь — 604 страницы, 89 символов в строке, 66 строк на странице.
4. Черно-белая фотография разбита на точки, каждая из которых кодируется пятью битами информации. Каков информационный объем фотографии 10×15 см, если один квадратный сантиметр содержит 1600 точек?

Вариант 5

1.

   а) 01001110010010010100111001000101

   б) 01000110010010010100110001000101

   в) 01001101010000010101001001001011

2. $k = 7$.
3. Книга рекордов Петербурга — 235 страниц, 60 символов в строке, 48 строк на странице.
4. Черно-белая фотография разбита на точки, каждая из которых кодируется тремя битами информации. Каков информационный объем фотографии 15×18 см, если один квадратный сантиметр содержит 3025 точек?

Вариант 6

1.

   а) 01000010010011000101010101000101

   б) 01000111010010010101001001001100

   в) 01001101010011110100111101001110

2. $k = 8$.
3. С. Я. Лурье "История Греции" — 674 страницы, 60 символов в строке, 48 строк на странице.
4. Черно-белая фотография разбита на точки, каждая из которых кодируется шестью битами информации. Каков информационный объем фотографии 5×8 см, если один квадратный сантиметр содержит 1764 точки?

Вариант 7

1.

   а) 01010111010010000100010101001110

   б) 01010010010011110100001101001011

   в) 01000010010101010100110001001100

2. $k = 9$.
3. "Библейская энциклопедия" — 902 страницы, 80 символов в строке, 55 строк на странице.
4. Черно-белая фотография разбита на точки, каждая из которых кодируется двумя битами информации. Каков информационный объем (в килобайтах) фотографии 9×14 см, если один квадратный сантиметр содержит 576 точек?

1.4. Системы счисления

Вопрос 12. Какие два класса систем счисления вы знаете?

Вопрос 13. Что такое базис системы счисления?

Вопрос 14. Что такое основание системы счисления?

Вопрос 15. Как определяется "вес" цифры в позиционной системе счисления?

Вопрос 16. Почему для компьютерной техники за основу была выбрана двоичная, а не десятичная система счисления?

Вопрос 17. Почему мы сами пользуемся десятичной системой счисления? Откуда пришли цифры нашей системы счисления? Откуда пришел ноль?

Вопрос 18. Каковы достоинства и недостатки двоичной системы счисления?

Практикум 9. Древние системы записи чисел

Рассмотрите внимательно таблицы с различными древними системами записи чисел (рис. 1.19—1.21). Запишите число 988 — год крещения Руси — в египетской, вавилонской и древнерусской записях.

	ЕГИПЕТ	ВАВИЛОН	СИРИЯ	АТТИКА
0				
1	𓏺	𒁹	𐡕	I
2	𓏺𓏺	𒁹𒁹	𐡕𐡕	II
3	𓏺𓏺𓏺	𒁹𒁹𒁹	𐡕𐡕𐡕	III
4	𓏺𓏺𓏺𓏺	𒁹𒁹𒁹𒁹	𐡕𐡕𐡕𐡕	IIII
5	𓏺𓏺𓏺 𓏺𓏺	𒁹𒁹𒁹 𒁹𒁹	𐡕𐡕𐡕𐡕𐡕	Γ
6	𓏺𓏺𓏺 𓏺𓏺𓏺	𒁹𒁹𒁹 𒁹𒁹𒁹	𐡕𐡕𐡕𐡕𐡕𐡕	ΓI
7	𓏺𓏺𓏺𓏺 𓏺𓏺𓏺	𒁹𒁹𒁹 𒁹𒁹𒁹	𐡕𐡕𐡕𐡕𐡕𐡕𐡕	ΓII
8	𓏺𓏺𓏺𓏺 𓏺𓏺𓏺𓏺	𒁹𒁹𒁹 𒁹𒁹𒁹 𒁹𒁹	𐡕𐡕𐡕𐡕𐡕𐡕𐡕𐡕	ΓIII
9	𓏺𓏺𓏺 𓏺𓏺𓏺 𓏺𓏺𓏺	𒁹𒁹𒁹 𒁹𒁹𒁹 𒁹𒁹𒁹	𐡕𐡕𐡕𐡕𐡕𐡕𐡕𐡕𐡕	ΓIIII
10	𓎆	𒌋	𐡘	Δ
15	𓎆𓏺𓏺𓏺 𓏺𓏺	𒌋𒁹𒁹𒁹 𒁹𒁹	𐡘𐡕	ΔΓ
20	𓎆𓎆	𒌋𒌋	𐡘𐡘	ΔΔ
30	𓎆𓎆𓎆	𒌋𒌋𒌋	𐡘𐡘𐡘	ΔΔΔ
40	𓎆𓎆𓎆𓎆	𒌋𒌋𒌋𒌋	𐡘𐡘𐡘𐡘	ΔΔΔΔ
50	𓎆𓎆𓎆 𓎆𓎆	𒌋𒌋𒌋 𒌋𒌋	𐡘𐡘𐡘𐡘𐡘	Γ̄
60	𓎆𓎆𓎆 𓎆𓎆𓎆	𒁹𒁹𒁹 𒁹𒁹𒁹	𐡘𐡘𐡘𐡘𐡘𐡘	Γ̄Δ
70	𓎆𓎆𓎆𓎆 𓎆𓎆𓎆	𒁹𒁹𒁹𒁹 𒌋	𐡘𐡘𐡘𐡘𐡘𐡘𐡘	Γ̄ΔΔ
80	𓎆𓎆𓎆𓎆 𓎆𓎆𓎆𓎆	𒁹𒁹𒁹𒁹 𒌋𒌋	𐡘𐡘𐡘𐡘𐡘𐡘𐡘𐡘	Γ̄ΔΔΔ
90	𓎆𓎆𓎆 𓎆𓎆𓎆 𓎆𓎆𓎆	𒁹𒁹𒁹𒁹 𒌋𒌋𒌋	𐡘𐡘𐡘𐡘𐡘𐡘𐡘𐡘𐡘	Γ̄ΔΔΔΔ
100	𓍢	𒁹𒁹	𐡘𐡕	H
500	𓍢𓍢𓍢 𓍢𓍢	𒁹𒁹𒁹 𒁹𒁹		Γ̄H
1000	𓆼	𒌋𒁹𒁹		X

Рис. 1.19. Примеры цифровой записи разных народов

I	1	VIII	8	LXXV	75	D	500
II	2	IX	9	XCII	92	DCXCV	695
III	3	X	10	IC	99	DCCIL	749
IV	4	XVIII	18	C	100	M	1000
V	5	XXXI	31	CCCII	302	MCMIX	1909
VI	6	XLVI	46	CDXLI	441	MCMLXXXIV	1984
VII	7	L	50	ID	499	MIM	1999

Рис. 1.20. Римские цифры

				СОТНИ		ТЫСЯЧИ		ДЕСЯТКИ В СОЕДИНЕНИИ С ЕДИНИЦАМИ			
а҃	1	і҃	10	р҃	100	҂а҃	1000	а҃і	11	к҃а	21
в҃	2	к҃	20	с҃	200	҂в҃	2000	в҃і	12	к҃в	22
г҃	3	л҃	30	т҃	300	҂г҃	3000	г҃і	13	к҃г	23
д҃	4	м҃	40	у҃	400	҂є҃	5000	д҃і	14	л҃в	32
є҃	5	н҃	50	ф҃	500	҂з҃	7000	є҃і	15	м҃г	43
ѕ҃	6	ѯ҃	60	х҃	600	в҃	20000	ѕ҃і	16	н҃є	55
з҃	7	о҃	70	ѱ҃	700	з҃	70000	з҃і	17	о҃є	75
и҃	8	п҃	80	ѿ	800	г҃	300000	и҃	18	р҃а	101
ѳ҃	9	ч҃	90	ц҃	900	є҃	500000	ѳ҃	19	р҃к҃в	122

Рис. 1.21. Древнерусские цифры

Практикум 10. Инопланетные истории

История 1

Прилетел как-то к земной девушке, красавице писаной, ухажер с планеты Double Fingers и ну замуж ее звать и похваляться, что де и зарабатывает он 1100000 долларов в месяц, и апартаменты у него общей площадью 10100 кв. м, и одних машин у него 10 штук. Однако девица наша была с умом и учла, что все это в двоичной системе. А сколько же по-нашему будет?

История 2

Прилетели теперь наши на далекую планету Gamma Rays. И разговорились с одним почтенным отцом семейства. Много детей, рассказал он, у него — тридцать три сына и пятьдесят дочерей. Всего детей — сто тринадцать, а годиков ему, соответственно, 101. А на самом-то деле, сколько чего?

История 3

А теперь представьте, что вы с ценным товаром — спичками — прилетели на планету Кин-дза-дза. У вас 10 000 упаковок по 20 рублей за упаковку. У них там шестнадцатеричная система счисления. Они, не разобравшись, все раскупают по 20 своих кин-дза-дзовских рублей, которые вы потом спокойно меняете на свои десятичные. Какова же неожиданная прибыль?

Практикум 11. Перевод чисел из одной системы счисления в другую

Предлагается в каждом из 14 вариантов (табл. 1.4) проделать следующие действия:

- Десятичное число из второго столбца надо перевести сначала в восьмеричную систему, а затем, воспользовавшись восьмеричными триадами, перевести из восьмеричной в двоичную, после чего с помощью шестнадцатеричных тетрад перевести в шестнадцатеричную. В ответе должно получиться три числа: восьмеричное, двоичное и шестнадцатеричное.

  Покажем на примере, как это делать.

  Имеется десятичное число 125_{10}. Делением на 8 переводим его в восьмеричную систему. Получится 175_8. Заменяем каждую цифру полученного числа восьмеричной триадой. Получаем $001\ 111\ 101_2$. Отбрасываем незначащие нули слева. Получаем двоичное число 1111101_2. Справа отсчитываем по 4 цифры. Получаем шестнадцатеричные тетрады — $0111\ 1101_2$. Заменяем тетрады шестнадцатеричными цифрами и получаем $7D_{16}$.

- Двоичные числа из третьего и четвертого столбцов перевести в десятичные (либо обычным способом, либо через восьмеричные или шестнадцатеричные).

  Покажем на примере, как это делать.

  Имеется число 1111101_2.

  Стандартный перевод: $1111101_2 = 2^6+2^5+2^4+2^3+2^2+1=$
  $=64+32+16+8+4+1=125_{10}$.

  Через восьмеричные: $1111101_2 = 001\ 111\ 101_2 = 175_8=1\times8^2+7\times8+5=125_{10}$.

  Аналогично осуществляется перевод и через шестнадцатеричные числа.

- Восьмеричные числа из пятого и шестого столбцов перевести в десятичную систему стандартным способом.

- Шестнадцатеричное число из седьмого столбца перевести в десятичную систему счисления.

Таблица 1.4. *Варианты заданий к практикуму 11*

№ вар.	Десят. число	Двоичное число	Двоичное число	Восьм. число	Восьм. число	Шестн. число
1	362	10011110110	101101110	666	543	E32
2	412	11011011011	111000111	362	412	DF3

Таблица 1.4 *(окончание)*

№ вар.	Десят. число	Двоичное число	Двоичное число	Восьм. число	Восьм. число	Шестн. число
3	289	10101010101	101011100	435	355	ABC
4	343	11100011100	111110000	353	401	39F
5	391	10111011010	110001111	444	423	48A
6	317	11010111111	101101011	416	457	36E
7	350	11100100110	100011110	427	466	FF2
8	328	10110111011	101101011	436	477	3CC
9	355	11001100110	100111000	447	300	F77
10	377	11101110111	101010101	466	306	F16
11	388	10011100010	111000111	421	406	FC8
12	344	10110101111	100110000	476	360	23D
13	322	11001100111	110001111	470	400	EE7
14	399	10001110011	101110000	401	436	AAA

Практикум 12. Перевод в стандартные коды ASCII и наоборот

Используя расширенную таблицу кодов ASCII (см. рис. 1.18), закодируйте предложенное слово из задания 1 и декодируйте слово из задания 2.

Вариант 1

1. Закодировать данный текст: IBM PC
2. Декодировать данный текст: 8A AE AC AF EC EE E2 A5 E0

Вариант 2

1. Закодировать данный текст: Автоматизация
2. Декодировать данный текст: 50 72 6F 67 72 61 6D

Вариант 3

1. Закодировать данный текст: Информатика
2. Декодировать данный текст: 50 72 6F 63 65 64 75 72 65

Вариант 4

1. Закодировать данный текст: Computer
2. Декодировать данный текст: 84 88 91 8A 8E 82 8E 84

Вариант 5

1. Закодировать данный текст: Printer
2. Декодировать данный текст: 43 4F 4D 50 55 54 45 52

Вариант 6

1. Закодировать данный текст: компьютеризация
2. Декодировать данный текст: 50 52 49 4E 54

Вариант 7

1. Закодировать данный текст: YAMAHA
2. Декодировать данный текст: 4D 4F 44 45 4D

Вариант 8

1. Закодировать данный текст: световое перо
2. Декодировать данный текст: 4C 61 73 65 72

Вариант 9

1. Закодировать данный текст: Микропроцессор
2. Декодировать данный текст: 88 AD E4 AE E0 AC A0 E2 A8 AA A0

Вариант 10

1. Закодировать данный текст: Принтер
2. Декодировать данный текст: 42 69 6E 61 72 79

Практикум 13. Суммирование положительных чисел в разных системах счисления

Вариант 1

а) в двоичной системе счисления: 1010110011 + 1100110011 =

б) в восьмеричной системе счисления: 3456 + 1706 =

в) в шестнадцатеричной системе счисления: АВСД + 7653 =

Вариант 2

а) в двоичной системе счисления: 00101100101 + 10011011111 =

б) в восьмеричной системе счисления: 3476 + 2563 =

в) в шестнадцатеричной системе счисления: F458 + 20DC =

Вариант 3

а) в двоичной системе счисления: 101100110 + 011011001 =

б) в восьмеричной системе счисления: 2376 + 5462 =

в) в шестнадцатеричной системе счисления: 2FB6 + 48DA =

Вариант 4

а) в двоичной системе счисления: 1101101101 + 0101011101 =

б) в восьмеричной системе счисления: 5264 + 3507 =

в) в шестнадцатеричной системе счисления: ED20 + 36AB =

Вариант 5

а) в двоичной системе счисления: 0010111011 + 1011011010 =

б) в восьмеричной системе счисления: 3264 + 5270 =

в) в шестнадцатеричной системе счисления: A0B1 + DACB =

Вариант 6

а) в двоичной системе счисления: 10110111000 + 11000101100 =

б) в восьмеричной системе счисления: 3207 + 2677 =

в) в шестнадцатеричной системе счисления: 57FE + AB0D =

Вариант 7

а) в двоичной системе счисления: 1100011110 + 1011011001 =

б) в восьмеричной системе счисления: 5670 + 3456 =

в) в шестнадцатеричной системе счисления: B0D6 + ABCD =

Вариант 8

а) в двоичной системе счисления: 11000111010 + 10110111101 =

б) в восьмеричной системе счисления: 7621 + 1352 =

в) в шестнадцатеричной системе счисления: 25BA + D0BF =

Вариант 9

а) в двоичной системе счисления: 0100001111 + 1101110101 =

б) в восьмеричной системе счисления: 3256 + 7061 =

в) в шестнадцатеричной системе счисления: A0B8 + D08A =

Вариант 10

а) в двоичной системе счисления: 11100101101 + 10110110111 =

б) в восьмеричной системе счисления: 7025 + 1777 =

в) в шестнадцатеричной системе счисления: 50DB + ADBE =

Вариант 11

а) в двоичной системе счисления: 1010110011 + 1100110011 =

б) в восьмеричной системе счисления: 3456 + 1706 =

в) в шестнадцатеричной системе счисления: ABCД + 7653 =

Вариант 12

а) в двоичной системе счисления: 00101100101 + 10011011111 =

б) в восьмеричной системе счисления: 3476 + 2563 =

в) в шестнадцатеричной системе счисления: F458 + 20DC =

Вариант 13

а) в двоичной системе счисления: 101100110 + 011011001 =

б) в восьмеричной системе счисления: 2376 + 5462 =

в) в шестнадцатеричной системе счисления: 2FB6 + 48DA =

1.5. Основы логической алгебры

Если бы компьютер умел только быстро вычислять, то он был бы большим калькулятором, что, собственно, для нас не так интересно. Но управляя подключенными к нему устройствами и выполняя команды пользователя, компьютеру приходится решать много логических задач. Тут-то и приходит на помощь алгебра логики или *булева алгебра* (по имени ее создателя — английского математика Дж. Буля).

В основу ее положено так называемое *элементарное логическое высказывание*. Таким высказыванием будем называть высказывание, которое может быть только *истинным* (true, истина) или *ложным* (false, ложь).

Например, высказывание "Москва — столица России" истинно, а высказывание "Сборная Франции — чемпион мира по футболу 2002 года" ложно.

Для упрощения действий элементарные высказывания обозначают буквами, а истину и ложь логическими единицей и нулем соответственно. Тогда простые элементарные высказывания можно связывать между собой с помощью логических функций и, зная как они работают, рассчитывать их.

Только надо понимать, что не все высказывания могут быть логическими. Например, высказывание "Москва — лучший город на Земле" — не логическое высказывание, потому что не имеет однозначного ответа. Для кого-то оно истинно, а для кого-то ложно.

Основные функции (логические операции) алгебры логики следующие:

- *Конъюнкция* (от лат. *conjunctio*, связываю):
  - в естественном языке соответствует союзу *и*;
  - обозначение: &;
  - в языках программирования обозначается: `and`;
  - иное название: *логическое умножение.*

  Конъюнкция — это логическая операция, ставящая в соответствие каждым двум простым высказываниям составное высказывание, являющееся истинным тогда и только тогда, когда оба исходных высказывания истинны (табл. 1.5).

Таблица 1.5. *Конъюнкция*

A	B	A&B
0	0	0
0	1	0
1	0	0
1	1	1

  Например, если есть высказывание А. "Москва — столица России" и В. "Москва самый большой город России", то их можно связать конъюнкцией: союзом "И", получится "Москва — столица И самый большой город России". Результат этой конъюнкции будет истинным, т. к. оба элементарных высказывания, входящие в ее состав, истинны.

- *Дизъюнкция* (от лат. *disjunctio*, различаю):
  - в естественном языке соответствует союзу *или*;
  - обозначение: $\vee$;
  - в языках программирования обозначается: `or`;
  - иное название: *логическое сложение.*

Дизъюнкция — это логическая операция, которая каждым двум простым высказываниям ставит в соответствие составное высказывание, являющееся ложным тогда и только тогда, когда оба исходных высказывания ложны, и истинным, когда хотя бы одно из двух образующих его высказываний истинно (табл. 1.6).

Таблица 1.6. *Дизъюнкция*

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

Высказывание "Я хочу кошку пушистую ИЛИ белую" будет истинным в трех случаях:

- Кошка пушистая, но не белая.
- Кошка не пушистая, но белая.
- Кошка и пушистая, и белая.

Во всех этих трех случаях я кошку беру. И только, если она ни белая, ни пушистая — при всей своей любви к животным, не возьму.

❒ *Инверсия* (от лат. *inversio*, переворачиваю):

- в естественном языке соответствует словам *Неверно, что...* и частице *не*;
- обозначение: $\overline{A}$;
- в языках программирования обозначается: `not`;
- иное название: *отрицание*.

Инверсия — это логическая операция, которая каждому простому высказыванию ставит в соответствие составное высказывание, заключающееся в том, что исходное высказывание отрицается (табл. 1.7).

Таблица 1.7. *Инверсия*

A	$\overline{A}$
0	1
1	0

Например, "Неверно то, что первым человеком в космосе был представитель Соединенных Штатов Америки".

Практикум 14. Установите, какие из следующих предложений являются логическими высказываниями, а какие — нет (объясните почему)

"Солнце вращается вокруг Земли".

"3 + 2 = 4".

"Сегодня отличная погода".

"В романе Л. Н. Толстого “Война и мир” 3 432 536 слов".

"Лондон расположен на Темзе".

"Музыка Брамса слишком сложна".

"Первая космическая скорость равна 7,8 км/с".

"Золото — металл".

"Если один угол в треугольнике прямой, то треугольник будет тупоугольным".

"Если сумма квадратов двух сторон треугольника равна квадрату третьей, то он прямоугольный".

Практикум 15. Определите значения истинности высказываний

"Наличия аттестата о среднем образовании достаточно для поступления в институт".

"Наличие аттестата о среднем образовании необходимо для поступления в институт".

"Если целое число делится на 6, то оно делится на 3".

"Подобие треугольников является необходимым условием их равенства".

"Подобие треугольников является необходимым и достаточным условием их равенства".

"Треугольники подобны только в случае их равенства".

"Треугольники равны только в случае их подобия".

"Равенство треугольников является достаточным условием их подобия".

"Для того чтобы треугольники были неравны, достаточно, чтобы они не были подобны".

"Для того чтобы четырехугольник был квадратом, достаточно, чтобы его диагонали были равны и перпендикулярны".

Определение истинности составных высказываний

Рассмотрим, например следующие простые высказывания:

A = {Принтер — устройство вывода информации},

B = {Процессор — устройство хранения информации},

C = {Монитор — устройство вывода информации},

D = {Клавиатура — устройство обработки информации}.

Попробуем теперь определить истинность составного высказывания:
$(\overline{A} \& \overline{B}) \& (C \vee D)$.

На основании знания устройства компьютера устанавливаем истинность простых высказываний: $A = 1$, $B = 0$, $C = 1$, $D = 0$.

Определяем истинность составного высказывания, используя таблицы истинности логических операций:

$(\overline{1} \& \overline{0}) \& (1 \vee 0) = (0 \& 1) \& (1 \vee 0) = (0 \& 1) \& (1 \vee 0) = 0 \& 1 = 0$

Практикум 16. Определите истинность составных высказываний

Даны простые высказывания:

A = {Принтер — устройство ввода информации};

B = {Процессор — устройство обработки информации};

C = {Монитор — устройство хранения информации};

D = {Клавиатура — устройство ввода информации}.

Определите истинность составных высказываний:

а) $(A \& B) \& (C \vee D)$;

б) $(A \& B) \& (C \vee D)$;

в) $(A \vee B) \& (C \& D)$;

г) $\overline{A} \& \overline{B}$.

Практикум 17. Определите истинность составных высказываний

Даны простые высказывания:

$A = \{5>3\}$, $B = \{2=3\}$ и $C = \{4<2\}$.

Определите истинность составных высказываний:

а) $(A \vee B) \& C \& (A \& C) \vee (B \& C)$;

б) $(A \& B) \vee C \& (A \vee C) \& (A \& B)$.

Практикум 18. Вычислите значения следующих логических функций при *A*=1, *B*=0, *C*=1, *D*=0.

1) $Y = D \vee \overline{A \& \overline{\overline{B \& C}}}$;

2) $Y = A \& B \vee C \& \overline{\overline{D}}$;

3) $Y = C \& (C \vee B \vee \overline{A \& D}) \vee \overline{B}$;

4) $Y = \overline{A \& (B \vee C)} \vee \overline{A \& (B \vee \overline{\overline{D}})}$;

5) $Y = A \& (B \vee C) \& D \vee (\overline{A} \vee \overline{B} \vee \overline{C} \vee \overline{\overline{D}})$.

Практикум 19. Логические задачи

Задача 1. Пять офицеров

В одной из горячих точек служили 5 офицеров: генерал, полковник, майор, капитан и лейтенант. Один из них сапер, другой — пехотинец, третий — танкист, четвертый — связист, а пятый — артиллерист. У каждого из них есть сестра. И каждый из них женат на сестре своего однополчанина.

Вот что еще известно об этих офицерах.

- По меньшей мере, один из родственников связиста старше его по званию.
- Капитан никогда не служил в Хабаровске.
- Оба родственника-пехотинца и оба родственника-танкиста служили раньше в Мурманске. Ни один родственник генерала в Мурманске не был.
- Танкист служил в Твери вместе с обоими своими родственниками, а лейтенант там не служил.
- Полковник служил в Махачкале вместе со своими родственниками.
- Танкист не служил в Махачкале. Там служил только один из его родственников.
- Генерал служил с обоими своими родственниками в Хабаровске, а в Махачкале он не бывал.
- Артиллерист не служил ни в Хабаровске, ни в Твери.

Определите, кто из офицеров какое звание имеет?

Задача 2. Разоблачение оракула

Давным-давно в одной из восточных стран был знаменитый оракул. В отличие от остальных оракулов, его устами вещало не одно божество, а целых три: бог Правды, бог Лжи и бог Дипломатии. Эти божества изображались

совершенно одинаковыми фигурами, расположенными в ряд за алтарем, перед которым преклоняли колени люди, ищущие совета. Боги всегда охотно отвечали на вопросы. Но т. к. они были похожи друг на друга, никто не мог определить, то ли отвечает бог Правды, которому надо верить, то ли бог Лжи, который говорит всегда неправду, то ли бог Дипломатии, который может либо солгать, либо сказать правду. Такое положение было на руку жрецам, ибо любой ответ оракула можно было толковать как угодно.

Но однажды нашелся кощунственный смельчак, который задумал совершить то, что не удавалось самым большим мудрецам. Он решил опознать каждого из богов.

Смельчак вошел в храм и спросил бога, стоявшего слева:

— Кто стоит рядом с тобой?

— Бог Правды, — ответил тот.

Тогда смельчак спросил бога, стоявшего в центре:

— Кто ты?

— Бог Дипломатии, — был ответ.

Последний вопрос смельчак задал богу, стоявшему справа:

— Кто стоит рядом с тобой?

— Бог Лжи, — ответил бог.

— Теперь все понятно, — довольно сказал смельчак.

Что же он понял из ответов богов?

Задача 3. Находчивый комендант

Комендант переселял студентов на время ремонта общежития. Дело это не простое. Посудите сами. На очередную комнату было 8 кандидатов, а поселить в нее можно было только четырех. Пошел комендант расспрашивать студентов, кто с кем жить хочет. Вот что он услышал:

- Андрей согласен на любых соседей.
- Борис без Кости не переселится.
- Костя не хочет жить в одной комнате с Василием.
- Василий согласен жить с кем угодно.
- Дима не будет переселяться без Юры.
- Федя не будет без Гриши жить в одной комнате с Димой, а без Димы не будет жить в одной комнате с Костей.
- Гриша не хочет, чтобы его соседями были и Борис, и Костя вместе, а кроме того, он не желает жить в одной комнате ни с Андреем, ни с Василием.

- Юра даст согласие переехать в новую комнату, если туда же переберутся либо Борис, либо Федя. Кроме того, Юра не будет жить в одной комнате с Костей, если туда не переедет Гриша, и не желает жить в одной комнате ни с Андреем, ни с Василием.

"Задали они мне задачу", — подумал комендант. Но в конце концов сумел учесть все пожелания. Каким образом?

Задача 4. Четыре "если"

Левин, Митерев и Набатов работают в банке в качестве бухгалтера, кассира и счетовода.

Если Набатов — кассир, то Митерев — счетовод. Если Набатов — счетовод, то Митерев — бухгалтер. Если Митерев — не кассир, то Левин — не счетовод. Если Левин — бухгалтер, то Набатов — счетовод. Кто какую должность занимает?

Задача 5. Любители птиц

В одном городе живут 7 любителей птиц. И фамилии у них птичьи. Каждый из них — "тезка" птицы, которой владеет один из его товарищей. У троих из них живут птицы, которые темнее, чем пернатые "тезки" их хозяев.

- "Тезка" птицы, которая живет у Воронова, женат.
- Голубев и Канарейкин — единственные холостяки из всей компании.
- Хозяин грача женат на сестре жены Чайкина.
- Невеста хозяина ворона очень не любит птицу, с которой возится ее жених.
- "Тезка" птицы, которая живет у Грачева, — хозяин канарейки.
- Птица, которая является "тезкой" владельца попугая, принадлежит "тезке" той птицы, которой владеет Воронов.
- У голубя и попугая оперение светлое.

Кому принадлежит скворец?

Задача 6. Финальный забег

Каждый второй пассажир в автобусе с увлечением читал спортивный раздел газеты, а остальные оживленно обсуждали последние спортивные новости. Иван Михайлович не успел купить газету, и ему не оставалось ничего другого, как заглядывать в газеты, развернутые другими пассажирами, и ловить доносившиеся до него обрывки разговоров.

Главной новостью дня был состоявшийся накануне финал эстафеты 4×100 м для мужчин. В финал после упорной борьбы вышли команды шести стран:

европейские команды A и B, африканские команды C и D и 2 команды — представительницы американского континента E и F.

Иван Михайлович охотно узнал бы, как распределились места среди участников финала, но сделать это оказалось непросто. В тот день Ивану Михайловичу особенно не везло: стоило ему пристроиться к кому-нибудь, чтобы заглянуть через плечо, как счастливый обладатель спортивной газеты тотчас переворачивал страницу, а доносившиеся со всех сторон реплики знатоков и ценителей спорта были маловразумительными.

Выйдя из автобуса, Иван Михайлович смог восстановить в памяти лишь следующие крохи информации.

1. Команда A одержала победу над командой B.
2. Африканская команда получила золотые медали.
3. Команда B одержала победу над командой D.
4. По всему было видно, что первое и второе места достанутся американским командам, и вдруг в последний момент между ними вклинилась европейская команда.
5. Африканская команда отстала от всех остальных участников финала.
6. Первыми финишировали 3 африканских бегуна.
7. Команда F одержала победу над командой B.
8. Команда E одержала победу над командой F.
9. В составе европейских команд не было африканских спортсменов.

Располагая этими отрывочными сведениями, Иван Михайлович попытался восстановить, как распределились места между шестью командами, участвовавшими в финале эстафетного бега, но тщетно.

Наконец, после тщательного анализа Иван Михайлович понял, что одна из девяти перечисленных выше посылок ложная. Он либо что-то не так понял, либо плохо разглядел, либо неправильно вспомнил.

Все остальные посылки истинны.

Как распределились места между шестью командами, принимавшими участие в финальном забеге?

Задача 7. Чья зебра?

В пяти соседних домах, окрашенных в разные цвета, живут пять человек различных национальностей. У каждого из них есть свое любимое животное, своя манера курить и свой любимый напиток.

- Англичанин живет в красном доме.
- У испанца есть собака.

- Кофе пьют в зеленом доме, который находится рядом с белым домом и справа от него.
- Француз пьет чай.
- У того, кто курит большие сигары, есть попугайчики.
- Маленькие сигары курят в желтом доме.
- Молоко пьют в среднем доме.
- Швед живет в крайнем доме слева.
- Тот, кто курит сигареты, живет в доме, соседнем с тем домом, где держат обезьяну.
- Тот, кто курит маленькие сигары, живет рядом с владельцем кошки.
- Тот, кто курит трубку, пьет апельсиновый сок.
- Итальянец вообще не курит.
- Швед живет рядом с голубым домом.

Кому принадлежит зебра (рис. 1.22)?

Рис. 1.22. Так чья же это зебра?

Глава 2

Компьютер

2.1. История развития вычислительной техники

В развитии вычислительной техники можно выделить предысторию и четыре поколения электронных вычислительных машин. У нас на глазах рождается пятое поколение! Развитие компьютеров, по-видимому, ярче всего отражает динамику научно-технического прогресса второй половины XX начала XXI веков.

Предыстория развития вычислительной техники начинается с глубокой древности. Одним из первых средств счета были китайские *счеты* (суан-пан), греческие счеты (абаки) и русские счеты, которые до сих пор кое-где пытаются конкурировать с современной вычислительной техникой. (рис. 2.1).

Рис. 2.1. Разновидности абаков и счетов

Прошло много лет, прежде чем появилась первая счетная машина, которую в 1642 году изобрел французский математик Блез Паскаль. Эта машина была

построена на *основе зубчатых колес* и могла суммировать десятичные числа. Впечатление о "способностях" этой машины высказал сам Паскаль, который сказал, что "арифметическая машина производит действие, приближающееся к нам больше, чем все, что делают животные".

Первую арифметическую машину, выполняющую четыре арифметических действия, создал в 1673 году немецкий математик Лейбниц. Эта арифметическая машина послужила прототипом арифмометров, которые начали производиться серийно с 1820 года и использовались вплоть до 60-х годов XX века.

Одновременно с широким распространением арифмометров появилась идея создания универсальной программируемой счетной машины, выдвинутая в 1823 году английским математиком Чарльзом Бэббиджем. Задуманный им проект машины содержал все основные устройства вычислительных машин: память, арифметическое устройство и устройство управления, устройство ввода и устройство печати. Проект этой машины реализовывался 70 лет, но его воплощение так и не было завершено. Однако вычислительные программы для этой машины были созданы! Их составила дочь Джорджа Байрона, герцогиня Ада Лавлейс, которая по праву считается первой женщиной-программистом. В ее честь назван язык программирования Ада.

Из-за сложности и механического износа деталей проект Бэббиджа, опережавший технические возможности своего времени, так и остался нереализованным. И только через 100 лет, в 40-х годах XX века, удалось создать программируемую счетную машину на *основе электромеханических реле.* Реле — это элемент, имеющий два рабочих состояния: "включено" и "выключено". Важно отметить, что при проектировании этих электромеханических счетных машин использовался аппарат математической логики.

Именно в 40-е годы начался бурный прогресс научных и технических новшеств в промышленности и вычислительной технике. Не успели начать серийно выпускать электромеханические счетные машины, как появились первые ЭВМ, в которых логические элементы были реализованы на *основе радиоламп.*

Первая электронная вычислительная машина "ЭНИАК" (ENIAC, Electronic Numeral Integrator and Calculator) была создана после Второй мировой войны, в 1946 году, в США. В группу создателей этой первой ЭВМ входил один из самых выдающихся ученых XX века — Джон фон Нейман, который и предложил основные принципы построения и функционирования универсальных программируемых вычислительных машин. Именно в соответствии с его идеями современные ЭВМ состоят из процессора, арифметического устройства, устройств ввода/вывода и памяти для хранения данных и программ.

Одновременно над проектами электронных вычислительных машин работали в Англии, где первая универсальная ЭВМ появилась в 1949 году, и в СССР, где в 1950 году была разработана первая электронно-вычислительная машина (рис. 2.2), получившая название МЭСМ (малая электронно-счетная

машина). Первая советская большая ЭВМ — БЭСМ была создана в 1952 г. под руководством профессора С. А. Лебедева (рис. 2.3).

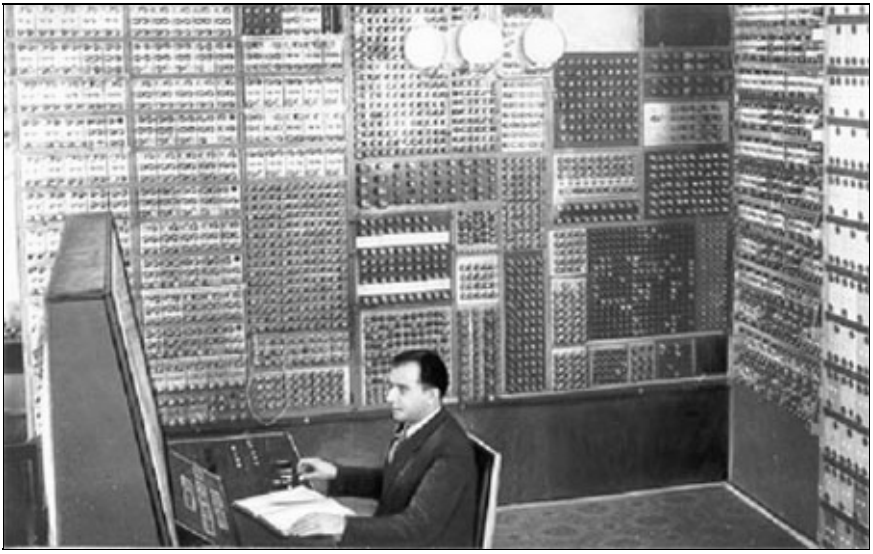


Рис. 2.2. МЭСМ

Рис. 2.3. С. А. Лебедев — отец первых советских ЭВМ

ЭВМ первого поколения изготовлялись на основе вакуумных электронных ламп. Эти ЭВМ размещались в нескольких больших металлических шкафах, занимавших целые залы, и требовали сложнейшей системы охлаждения. Работа на ЭВМ производилась за пультом, где можно было видеть состояние каждой ячейки памяти и любого регистра. Программы для ЭВМ первого поколения составлялись в машинных кодах — в виде длинных последова-

тельностей двоичных чисел. Занимались этим исключительно математики, выполнявшие на ЭВМ сложнейшие расчеты.

Следующее, второе поколение ЭВМ появилось через 10 лет — в 60-х годах XX века. В этих ЭВМ логические элементы реализовывались уже не на радиолампах, а на базе полупроводниковых приборов — *транзисторов*. Это позволило значительно увеличить надежность машин, сократить их размеры и потребление электроэнергии. Тем самым открылся путь для серийного производства ЭВМ.

В составе ЭВМ второго поколения появились печатающие устройства для вывода, телетайпы для ввода и магнитные накопители для хранения информации. Диалог человека с ЭВМ стал более естественным благодаря появлению языков программирования высокого уровня, таких как Фортран, Алгол, Бейсик и др. Начали создаваться первые автоматизированные системы на базе ЭВМ.

Лучшей по тем временам электронно-вычислительной машиной была БЭСМ-6 (рис. 2.4).

Рис. 2.4. БЭСМ-6

Для появления третьего поколения ЭВМ вновь понадобилось всего лишь около 10 лет. Основу этих компьютеров составляли *интегральные микросхемы*, содержавшие на одной полупроводниковой пластинке сотни и даже тысячи транзисторов. Благодаря этому уменьшились размеры, потребление электроэнергии и стоимость ЭВМ.

В состав ЭВМ третьего поколения были включены удобные устройства ввода/вывода и накопления информации (дисплеи) на основе электронно-лучевых трубок, накопители на магнитных лентах и дисках, графопостроители и т. п. Количество компьютеров к этому времени достигло уже десят-

ков и сотен тысяч. К работе с ними стал подключаться широкий круг специалистов: инженеры, техники. Вычислительные машины появились в университетах и институтах. Начали создаваться операционные системы, базы данных, языки структурного программирования, первые системы "искусственного интеллекта", стали внедряться системы автоматизированного проектирования и управления.

Еще через 10 лет, на рубеже 80-х годов, были созданы и выпущены в массовое производство ЭВМ четвертого поколения. Их элементной базой стали *большие интегральные схемы* (БИС), в которых на одном кристалле кремния размещались уже десятки и сотни тысяч логических элементов. Такие интегральные схемы позволяют создавать на одном-единственном кристалле программируемые блоки управления различными устройствами. Малые габариты и слабые токи, необходимые для их работы, позволяют устанавливать эти процессоры в любое техническое изделие: в телевизоры, стиральные машины, автомобили и т. д. Тем самым открылась возможность создания принципиально новых, программно управляемых технических устройств.

Наиболее яркими представителями ЭВМ четвертого поколения служат персональные компьютеры, габариты которых позволяют устанавливать их на любом рабочем месте и дома. В состав этих компьютеров включаются удобные средства накопления, ввода и предоставления информации: накопители на гибких магнитных дисках, цветные графические дисплеи, графические планшеты, компактные печатающие устройства.

Массовое распространение персональных ЭВМ изменило требования к программам. Главными из этих требований стали: простота правил работы, эстетичность, надежность программ, универсальность их функций, простота обучения работе на компьютерах. Десятки и сотни миллионов персональных компьютеров, устанавливаемых в службах сервиса и управления, на производстве и в образовании, требуют овладения компьютерной грамотностью от всего взрослого населения, а также подготовки специалистов по созданию, развитию и применению компьютеров.

Тест 1. Знаете ли вы историю развития вычислительной техники

Ответов на этот тест мы не приводим, потому что, во-первых, бóльшую часть ответов можно найти в вышеприведенном материале, а, во-вторых, этот тест используется нами как задание по поиску информации в Интернете (о таких заданиях речь еще пойдет ниже).

1. Первоначальный смысл английского слова "компьютер":

   а) разновидность телескопа;

   б) электронный аппарат;

в) электронно-лучевая трубка;

г) человек, производящий расчеты.

2. Общим свойством машины Бэббиджа, современного компьютера и человеческого мозга является способность обрабатывать:

а) числовую информацию;

б) текстовую информацию;

в) звуковую информацию;

г) графическую информацию.

3. Двоичную систему счисления впервые предложил:

а) Блез Паскаль;

б) Готфрид Вильгельм Лейбниц;

в) Чарльз Бэббидж;

г) Джордж Буль.

4. Первую вычислительную машину изобрел:

а) Джон фон Нейман;

б) Джордж Буль;

в) Норберт Винер;

г) Чарльз Бэббидж.

5. Первая программа была написана:

а) Чарльзом Бэббиджем;

б) Адой Лавлейс;

в) Говардом Айкеном;

г) Полом Алленом.

6. Основы теории алгоритмов были впервые заложены в работе:

а) Чарльза Бэббиджа;

б) Блеза Паскаля;

в) С. А. Лебедева;

г) Алана Тьюринга.

7. Современную организацию ЭВМ предложил:

а) Джон фон Нейман;

б) Джордж Буль;

в) Ада Лавлейс;

г) Норберт Винер.

8. Основные принципы цифровых вычислительных машин были разработаны:

   а) Блезом Паскалем;

   б) Готфридом Вильгельмом Лейбницем;

   в) Чарльзом Бэббиджем;

   г) Джоном фон Нейманом.

9. Под термином "поколение ЭВМ" понимают:

   а) все счетные машины;

   б) все типы и модели ЭВМ, построенные на одних и тех же научных и технических принципах;

   в) совокупность машин, предназначенных для обработки, хранения и передачи информации;

   г) все типы и модели ЭВМ, созданные в одной и той же стране.

10. Первая ЭВМ появилась:

    а) в 1823 году;

    б) в 1946 году;

    в) в 1949 году;

    г) в 1951 году.

11. Первая ЭВМ называлась:

    а) МИНСК;

    б) БЭСМ;

    в) ЭНИАК;

    г) IBM.

12. Первые ЭВМ были созданы в XX веке:

    а) в 40-е годы;

    б) в 60-е годы;

    в) в 70-е годы;

    г) в 80-е годы.

13. Языки высокого уровня появились:

    а) в первой половине XX века;

    б) во второй половине XX века;

    в) в 1946 году;

    г) в 1951 году.

14. Машины первого поколения были созданы на основе:
    а) транзисторов;
    б) электронно-вакуумных ламп;
    в) зубчатых колес;
    г) реле.
15. Элементной базой ЭВМ второго поколения являются:
    а) электронные лампы;
    б) полупроводники;
    в) интегральные микросхемы;
    г) БИС, СБИС.
16. В каком поколении машин появились первые программы?
    а) в первом поколении;
    б) во втором поколении;
    в) в третьем поколении;
    г) в четвертом поколении.
17. Для машин какого поколения потребовалась специальность "оператор ЭВМ"?
    а) первого поколения;
    б) второго поколения;
    в) третьего поколения;
    г) четвертого поколения.
18. В каком поколении машин появились первые операционные системы?
    а) в первом поколении;
    б) во втором поколении;
    в) в третьем поколении;
    г) в четвертом поколении.
19. Основной элементной базой ЭВМ третьего поколения являются:
    а) БИС;
    б) СБИС;
    в) интегральные микросхемы;
    г) транзисторы.
20. Основной элементной базой ЭВМ четвертого поколения являются:
    а) полупроводники;
    б) электромеханические схемы;

в) электровакуумные лампы;

г) СБИС.

21. Первая ЭВМ в нашей стране появилась:

а) в XIX веке;

б) в 60-х годах XX века;

в) в первой половине XX века;

г) в 1951 году.

22. Какая из отечественных ЭВМ была лучшей в мире ЭВМ второго поколения?

а) МЭСМ;

б) Минск-22;

в) БЭСМ;

г) БЭСМ-6.

23. Основоположником отечественной вычислительной техники является:

а) Сергей Алексеевич Лебедев;

б) Николай Иванович Лобачевский;

в) Михаил Васильевич Ломоносов;

г) Пафнутий Львович Чебышев.

24. Машины какого поколения позволяют нескольким пользователям работать с одной ЭВМ?

а) первого поколения;

б) второго поколения;

в) третьего поколения;

г) четвертого поколения.

25. Что представляет собой большая интегральная схема (БИС)?

а) транзисторы, расположенные на одной плате;

б) кристалл кремния, на котором размещаются от десятков до сотен тысяч логических элементов;

в) набор программ для работы на ЭВМ;

г) набор ламп, выполняющих различные функции.

26. Фирма IBM первоначально занималась:

а) автоматическим статистическим учетом иммигрантов;

б) производством калькуляторов;

в) производством запасных частей к автомобилям "Ford";

г) производством швейных машин.

27. Первый компьютер "Apple", который уже можно было назвать персональным, продавался в 1976 году по цене:

а) 777 долларов;

б) 666 долларов;

в) 555 долларов;

г) 1313 долларов.

28. Массовое производство персональных компьютеров началось в XX веке:

а) в 40-е годы;

б) в 90-е годы;

в) в 50-е годы;

г) в 80-е годы.

29. Первая ЭВМ в нашей стране называлась:

а) Стрела;

б) МЭСМ;

в) IBM PC;

г) БЭСМ.

30. В настоящее время в мире ежегодно компьютеров производится:

а) около 1 млн.;

б) около 500 млн.;

в) около 10 млн.;

г) около 100 млн.

2.2. Персональные компьютеры. Классификация

Компьютеры получили широкое распространение практически во всех сферах нашей жизни. Для решения различных задач используются разные типы компьютеров. На данный момент существует следующая классификация компьютеров.

- *IBM-совместимые персональные компьютеры*. Сейчас это наиболее распространенный тип компьютеров. Они являются своего рода универсальными машинами и могут применяться практически для любых задач. Однако в некоторых случаях, связанных с обработкой больших объемов информации, их применение будет нецелесообразным.

- *Персональные компьютеры Macintosh*. Компьютеры, предназначенные для тех же целей, что и компьютеры IBM, но не совместимые с ними. Правда в последнее время делаются шаги навстречу друг другу.
- *Рабочие станции*. Более мощные компьютеры, чем персональные. Используются обычно для хранения данных.
- *Серверы*. Как правило, самые мощные персональные компьютеры с большим дисковым пространством. Используются в компьютерных сетях для централизованного хранения и обработки данных.
- *Мейнфреймы*. Считаются наиболее мощными (не считая суперкомпьютеров) вычислительными системами общего назначения, обеспечивающими непрерывный круглосуточный режим эксплуатации. Они могут оснащаться одним или несколькими процессорами.
- *Суперкомпьютеры* — компьютеры, созданные для обработки больших объемов информации. Очень часто они используются для работы с трехмерным моделированием и анимацией. Наибольшее применение имеют в оборонных целях.

2.2.1. Основные устройства персональных компьютеров

Все устройства ПК делятся на *основные*, без которых работа компьютера невозможна, и *дополнительные*, которые расширяют его функциональные возможности.

К основным частям (блокам) относятся: *системный блок*, *монитор* и *клавиатура*. Монитор предназначен для вывода графической и текстовой информации. Клавиатура — основное устройство ввода данных и команд. К системному блоку подключаются все устройства: и монитор, и клавиатура, и *мышь*, и *принтер*, и *модем*, и т. д. (рис. 2.5).

Рис. 2.5. Персональный компьютер

В системном блоке располагаются:

- *материнская плата*, на которой находятся все основные электронные элементы: микропроцессор, оперативная память, а также разъемы расширения, в которые вставляются остальные платы (звуковая, видео и др.);
- *микропроцессор* — выполняет арифметико-логические действия и управляет всеми другими устройствами;
- *память* — внутренняя (оперативная и постоянная) и внешняя (дисководы, винчестер, CD-ROM).

2.2.2. Основные характеристики IBM-совместимых компьютеров

Фирма IBM (Ай-Би-Эм, International Business Machines) уже несколько лет является мировым лидером на рынке персональных компьютеров. Вот ее краткая история.

IBM, американская электронная корпорация, крупнейший мировой производитель всех видов компьютеров и программного обеспечения и один из крупнейших провайдеров глобальных информационных сетей. Корпорация занимает шестое место в списке крупнейших компаний мира. Штаб-квартира расположена в городе Армонк (штат Нью-Йорк). Была основана в 1911 году. Современное название получила в 1924 году. С середины 1950-х годов заняла ведущее положение на компьютерном рынке. В 1981 году создала свой первый персональный компьютер. К середине 1980-х годов контролировала около 60% мирового производства ЭВМ.

Несмотря на огромное значение рынка персональных компьютеров, интересы IBM простираются гораздо шире. Традиционно сильны позиции корпорации в производстве мэйнфреймов. В 1995 году IBM получила престижный заказ американского правительства на создание самого мощного в мире суперкомпьютера для Ливерморской лаборатории — центра ядерных исследований в США. В 1996—1997 году детище IBM — шахматный компьютер Deep Blue вступил в единоборство с чемпионом мира по шахматам Гарри Каспаровым. IBM выпускает также собственные микропроцессоры, а ее операционная система OS/2 применяется в каждом третьем банке США.

Исследования ученых в научных лабораториях IBM выходят далеко за рамки чисто коммерческих интересов и имеют значение для всей мировой науки. В 1986 году сотрудники IBM Г. Бинниг (Германия) и Г. Рорер (Швейцария) получили Нобелевскую премию по физике за создание растрового туннельного микроскопа, а в 1987 году Нобелевской премии по физике были удостоены сотрудники IBM Й. Г. Беднорц (Германия) и К. А. Мюллер (Швейцария) за открытие новых сверхпроводящих мате-

риалов. IBM занимает первое место среди компаний США по количеству полученных патентов на изобретения. В 1996 году IBM запатентовала 1867 изобретений. К концу 1990-х годов корпорация тратила около 5 млрд. долларов в год на научные исследования.

В 1993 году новый председатель совета директоров Луис Герстнер выбрал новое стратегическое направление корпорации — создание сетевого компьютера и развитие сетевых технологий. Первый образец такого компьютера появился в 1996 году, а 31 декабря того же года IBM, Mastercard и Датская платежная система объявили о первой транзакции (платеже) через Интернет с использованием протокола SET. Ближайшими своими задачами IBM видит создание надежных систем для электронного бизнеса. IBM принадлежит 95% рынка программного обеспечения для работы банкоматов. Являясь самым крупным провайдером услуг Интернета, компания обслуживает более 50 тыс. корпоративных клиентов в 950 городах в более чем ста странах мира.

Микропроцессор — важнейший элемент компьютера, т. к. им определяется скорость выполнения машиной программ пользователя. Со времени появления персональных компьютеров (ПК) сменилось несколько поколений процессоров, что составляет следующий ряд в порядке увеличения скорости: 8088, 286, 386SX, 386DX, 486SX, 486DX, 486DX2, Pentium, Pentium Pro и др.

Параметры процессора:

- *разрядность* — ширина "такта", по которому передается компьютерная информация: 8, 16, 32 или 64 бита;
- *тактовая частота*, характеризующая число команд, выполняемых процессором за одну секунду (измеряется в мегагерцах, МГц). Обычно тактовая частота соответствует 160—200 МГц.

Микропроцессор включает в себя:

- *арифметико-логическое устройство* (АЛУ), которое производит операции (микрооперации), необходимые для выполнения команд микропроцессора;
- *устройство управления* (УУ) — управляет всеми частями компьютера посредством принципов программного управления;
- *микропроцессорная память* (МПП). В микропроцессоре есть несколько ячеек собственной памяти, они называются *регистрами*. Некоторые из них предназначены для хранения операндов — величин, участвующих в текущей операции. Такие регистры называются *регистрами общего назначения* (RON). *Регистр команд* (RK) предназначен для хранения текущей команды. В *регистре-счетчике команд* (СК) хранится текущий адрес. Перед выполнением программы необходимо задать ее начальный адрес — записать его в счетчик команд.

Через *интерфейсную систему*, основу которой составляет системная шина персонального компьютера, микропроцессор соединяется с *основной памятью* и *внешней памятью* компьютера.

Основная память компьютера, состоит из:

- *оперативного* запоминающего устройства (ОЗУ), которое хранит работающую программу и данные;
- *постоянного* запоминающего устройства (ПЗУ), которое хранит информацию, которая необходима для постоянной работы.

ОЗУ и ПЗУ разбиты на ячейки, каждой из которых присвоен порядковый номер (адрес).

Внешняя память компьютера сосредоточена в накопителях на жестких магнитных дисках — устройствах с несъемными носителями (винчестеры). *Жесткие диски* служат для постоянного хранения в компьютере программ и данных. Выполнение многих современных прикладных программ без них невозможно. Большинство жестких дисков, в отличие от имеющих меньшую емкость дискет, нельзя снять, поэтому их и называют несъемными дисками. Можно, однако, приобрести и съемные жесткие диски. Они ценны, когда нужно сохранять конфиденциальность информации, и как средство переноса больших объемов данных между компьютерами. Емкость современных накопителей на жестких магнитных дисках может достигать нескольких десятков Гбайт.

Все бóльшую популярность благодаря большой емкости и надежности приобретают *накопители на оптических дисках.* Неперезаписываемые лазерные оптические диски обычно называют компакт-дисками ПЗУ — Compact disk ROM (CD-ROM). Они имеют емкость до 1,5 Гбайт, время доступа — от 30 до 300 мс.

Каждый компьютер оснащен также *накопителем на гибких магнитных дисках.* Для данных накопителей носителями являются дискеты (флоппи-диски).

Кроме устройств памяти, микропроцессор с помощью системной шины соединяется и с другими компонентами компьютера:

- с генератором тактовых импульсов, который генерирует последовательность электрических импульсов, частота которых определяет тактовую частоту машины (ее быстродействие);
- с таймером — внутренними электронными часами, которые подключены к автономному источнику питания (аккумулятору), и продолжают работать даже после отключения машины от питающей сети;
- с каналом связи через сетевой адаптер. Сетевой адаптер дает возможность подключить компьютер к локальной сети. При этом пользователь может получать доступ к данным, находящимся на других компьютерах;

- с клавиатурой через специальный разъем;
- с монитором через видеоадаптер;
- с принтером через адаптер принтера;
- с источником питания.

Вопросы

1. Перечислите основные устройства компьютера.
2. Что размещается на материнской плате? Какие фирмы — производители современных материнских плат вы знаете? Какую плату вы предпочли бы иметь в своем домашнем ПК?
3. Опишите эволюцию процессоров ПК. Какие процессоры сейчас конкурируют на рынке? В чем различие процессоров Celeron и Intel?
4. Что такое системная шина? Зависит ли от нее быстродействие компьютера? Что такое тактовая частота?
5. Как подразделяется память компьютера? Какая из них является энергонезависимой, а какая — энергозависимой?
6. Какие накопители информации вы знаете? Проведите сравнительный анализ гибких дисков 5,25" и 3,5", жестких дисков (винчестеров), CD-ROM, CD-RW, DVD.
7. Расскажите об основных характеристиках мониторов. Перспективы их развития.
8. Что такое регистры?
9. Что такое Hardware и Software?
10. Ниже приведены краткие характеристики двух ПК. Что вы можете сказать об этих характеристиках? Какой предпочли бы купить и почему?

    а) Intel Celeron 1000/ MSI 6337EPT Lite/ 128/ ATX/ 40 Gb Seagate 5400/ 32 Mb GeForce2 MX200/ Audio on board/ CD 52 Asus/ 15"Samsung 56E;

    б) Intel Pentium 4 1,7/ Asus TUSL2 i815/ 512/ ATX/ 120 Gb IBM 7200/ 64 Mb Ati Radeon 8500/ Creative Live! 5.1/ CD 40 Teac/ Жидкокрист.18"LG.

2.2.3. Внешние (периферийные) устройства персонального компьютера

Различные типы *периферийных устройств*, подключаемых к компьютерной системе, играют важную роль в ее работе. Они в значительной степени определяют возможности использования компьютеров и их технические характеристики. Широкий ассортимент выпускаемых периферийных устройств

позволяет выбирать те из них, с которыми профессиональные компьютеры в различных областях деятельности используются наиболее эффективно.

В зависимости от функций, выполняемых компьютерной системой, периферийные устройства могут подразделяться на две основные группы. К первой относятся те периферийные устройства, наличие которых абсолютно необходимо для функционирования компьютерной системы. Их обычно называют *системными периферийными устройствами.* К этой группе относятся видеомонитор, клавиатура, накопитель на гибком магнитном диске (НГМД), накопитель на жестком магнитном диске (НЖМД) и печатающее устройство (принтер).

Ко второй группе периферийных устройств относятся накопители на магнитной ленте, устройства для ввода графической информации (дигитайзеры), устройства для вывода графической информации (плоттеры), модем, сканер, аудиоплата, мышь или трекбол, коммуникационные адаптеры и др. Они предоставляют профессиональному компьютеру дополнительные возможности. Однако наличие их в его конфигурации определяется конкретной областью деятельности. В связи с этим данная группа носит название *дополнительных периферийных устройств.*

Многие периферийные устройства подсоединяются к компьютеру через специальные гнезда (разъемы), находящиеся обычно на задней стенке системного блока компьютера. Кроме монитора и клавиатуры такими устройствами являются:

- принтер — устройство для вывода на печать текстовой и графической информации;
- мышь — устройство, облегчающее ввод информации в компьютер;
- джойстик — манипулятор в виде укрепленной на шарнире ручки с кнопкой, употребляется в основном для компьютерных игр,

а также другие устройства.

Некоторые устройства, например, многие разновидности сканеров (приборов для ввода в компьютер рисунков и текста), используют смешанный способ подключения: в системный блок компьютера вставляется только электронная плата (контроллер), управляющая работой устройства, а само устройство подсоединяется к этой плате кабелем.

В настоящее время разрабатываются все более новые и совершенные периферийные устройства.

Вопросы (продолжение)

11. Какие типы принтеров вы знаете? Проведите их сравнительный анализ. Какие фирмы — производители принтеров вы знаете?
12. Графопостроитель и плоттер. Каково их предназначение?

13. Сканеры ручные и планшетные. Можно ли с помощью сканера распознавать печатный текст? А рукописный?
14. Модемы встроенные и внешние. Основные характеристики. Что влияет на их скорость передачи данных? Как называется единица скорости передачи данных?
15. Мыши и трекболы. Сходство и различия.
16. Что такое мультимедийный компьютер? Какие устройства для него нужны?
17. Разделите нижеприведенный список на два — на устройства ввода и устройства вывода:

    принтер, джойстик, микрофон, графопостроитель, сканер, мышь, клавиатура, плоттер, трекбол, кассеты с магнитной лентой, жесткий магнитный диск, CD-ROM, монитор, колонки, модем, принтер, стример.

Тест 2. Периферийные устройства

1. Плоттер — это устройство для:

   а) сканирования информации;

   б) считывания графической информации;

   в) вывода;

   г) ввода.

2. Какое устройство ЭВМ относится к внешним?

   а) арифметико-логическое устройство;

   б) центральный процессор;

   в) принтер;

   г) оперативная память.

3. Что является характеристикой монитора?

   а) цветовое разрешение;

   б) тактовая частота;

   в) дискретность;

   г) время доступа к информации.

4. Устройство ввода предназначено для:

   а) передачи информации от человека машине;

   б) обработки вводимых данных;

   в) реализации алгоритмов обработки, накопления и передачи информации;

   г) передачи информации в Глобальную сеть.

5. Аппаратное подключение периферийного устройства к магистрали производится через:

   а) регистр;

   б) драйвер;

   в) контроллер;

   г) стример.

6. Устройством вывода является:

   а) клавиатура;

   б) стример;

   в) факс-модем;

   г) дискета.

7. Манипулятор "мышь"— это устройство:

   а) сканирования информации;

   б) вывода;

   в) считывания информации;

   г) ввода.

8. Устройством ввода является:

   а) сканер;

   б) принтер;

   в) дисплей;

   г) стример.

9. Устройство вывода предназначено для:

   а) обучения, игры, расчетов и накопления информации;

   б) программного управления работой вычислительной машины;

   в) передачи информации от машины человеку;

   г) передачи информации в Глобальную сеть.

10. Какую функцию выполняют периферийные устройства?

    а) управление работой ЭВМ по заданной программе;

    б) хранение информации;

    в) ввод и выдачу информации;

    г) обработку информации.

Тест 3. Устройства компьютера

И еще один тест по устройствам компьютера.

1. Какие виды памяти используются в ПК?

   а) только оперативная память (ОЗУ, или RAM);

   б) только постоянная память (ПЗУ) ROM;

   в) только внешняя память (дискета, компакт-диск, жесткий диск и т. д.);

   г) все перечисленные ответы.

2. Какие функции выполняет центральный процессор (CPU)?

   а) руководит всей работой ПК, осуществляя связь между всеми частями компьютера;

   б) осуществляет связь между персональным компьютером и периферийными устройствами;

   в) выполняет только низкоуровневые команды;

   г) используется для осуществления связи между электронными компонентами ПК.

3. Скорость работы ПК в целом определяется:

   а) скоростью вычислений процессора;

   б) скоростью обмена данными с периферийными устройствами и оперативной памятью;

   в) скоростью процессора и обмена данными с периферийными устройствами и оперативной памятью;

   г) только разрядностью системной шины.

4. Оперативная память (ОЗУ, или RAM) служит для:

   а) временного хранения данных и очищается при выключении питания ПК;

   б) временного хранения данных и при выключении питания ПК не очищается;

   в) временного хранения данных, от состояния питания ПК не зависит;

   г) долговременного хранения данных.

5. Что из перечисленного не является внешней памятью?

   а) накопитель на гибком магнитном диске (дисковод 3,5");

   б) накопитель на лазерном диске (CD-ROM);

   в) накопитель на жестком магнитном диске (жесткий диск);

   г) BIOS.

6. Привод CD-ROM позволяет:

   а) только считывать информацию с компакт-диска;

   б) только записывать информацию на компакт-диск;

в) считывать и записывать информацию на компакт-диск;

г) только проигрывать музыкальные файлы.

7. Какое из перечисленных устройств быстрее обменивается информацией с ЦП (CPU)?

а) жесткий диск;

б) привод компакт-диска (CD-ROM);

в) дисковод для гибких магнитных дисков;

г) принтер.

8. Какой тип принтеров является наиболее производительным?

а) матричный принтер;

б) струйный принтер;

в) лазерный принтер;

г) фотопринтер.

9. Манипулятор "мышь" является:

а) внешним устройством;

б) устройством вывода;

в) устройством ввода (управления);

г) устройством обработки информации.

10. С помощью сканера можно:

а) вводить в ПК фотографии, тексты и рисунки;

б) вводить только фотографии;

в) выводить в цифровом виде фотографии, рисунки и тексты;

г) печатать текстовую информацию.

11. Какое из перечисленных устройств не является устройством ввода?

а) мышь;

б) сканер;

в) принтер;

г) клавиатура.

12. Какое устройство компьютера моделирует мышление человека?

а) оперативная память;

б) процессор;

в) внешняя память;

г) регистры процессора.

13. Для подключения внешних устройств рекомендуется:

    а) завершить работу с приложениями и сохранить файлы;

    б) проверить наличие заземления и надежность контактов в подключаемых кабелях и разъемах;

    в) подсоединение внешних устройств, замену плат производить при выключенном питании ПК;

    г) отсоединить монитор от системного блока.

14. Для долговременного хранения пользовательской информации служит:

    а) постоянное запоминающее устройство;

    б) дисковод;

    в) процессор;

    г) внешняя память.

15. Что такое "метка диска"?

    а) надпись на дискете с помощью фломастера;

    б) цифровая надпись;

    в) обозначение серийного номера;

    г) штрих-код.

16. Для чего необходима защелка на дискете?

    а) для предотвращения чтения информации;

    б) для предотвращения чтения и записи;

    в) для запрета несанкционированного удаления или изменения информации (файлов);

    г) для красоты.

17. Для организации связи в Глобальной компьютерной сети необходимы следующие ресурсы:

    а) компьютер, модем, программное обеспечение для поддержки работы в сети;

    б) компьютер, сетевой адаптер, канал связи, соответствующее программное обеспечение;

    в) компьютер, модем, сетевой адаптер;

    г) компьютер, телефонный аппарат, интернет-карта.

18. Модем обеспечивает:

    а) модуляцию (преобразование двоичной информации в аналоговую);

    б) демодуляцию (преобразование аналоговой информации в двоичную);

в) модуляцию и демодуляцию;

г) усиление сигнала.

19. В высказывание "Программа, хранящаяся во внешней памяти, после вызова на выполнение попадает в ... и обрабатывается ..." вместо многоточий выбрать правильные выражения:

а) устройство ввода, процессором;

б) процессор, регистрами процессора;

в) постоянное запоминающее устройство, процессором;

г) оперативную память, процессором.

20. Что из нижеперечисленного никогда не использовалось в вычислительной техники в качестве носителя информации:

а) магнитная лента;

б) магнитный диск;

в) магнитный барабан;

г) магнитная пирамида.

Тест 4. Знание клавиатуры

Как ни смешно, а это — большая беда. Даже в старших классах учащиеся часто неграмотно пользуются клавиатурой, не могут самостоятельно переключить раскладку клавиатуры ЛАТ/РУС, подолгу ищут знаки препинания, не умеют редактировать текст, не знают назначения клавиш. Проверим!

1. Нажатие какой клавиши (не всегда, но часто) вызывает выход из режима программы или отмену предыдущей команды, действия?

а) <Esc>;

б) <Tab>;

в) <Enter>;

г) <Delete>.

2. При нажатии на какую клавишу происходит перемещение курсора на одну позицию влево с удалением находящегося в этой позиции символа?

а) <Tab>;

б) <Enter>;

в) <Backspace>;

г) <Delete>.

3. Нажатие какой клавиши в режиме редактирования (в DOS) перемещает курсор на 8 позиций вправо (с клавишей <Shift> — влево)? (В других

программах ее назначение может быть иным: переход на другие панели, окна.)

а) <Tab>;

б) <Enter>;

в) <Alt>;

г) <Home>.

4. Назначение какой клавиши состоит в завершении набора текущей строки и передаче набранной команды для исполнения?

а) <PrintScreen>;

б) <Enter>;

в) <Alt>;

г) <Esc>.

5. Укажите клавишу, которая служит для перемещения курсора в начало текущей строки?

а) <F1>;

б) <Home>;

в) <Esc>;

г) <End>.

6. Какая клавиша обеспечивает перемещение курсора в конец текущей строки.

а) <Enter>;

б) <End>;

в) <Tab>;

г) <Home>.

7. С помощью какой клавиши компьютер переводится в режим вставки или замещения?

а) <Tab>;

б) <Insert>;

в) <Esc>;

г) <Backspace>.

8. Посредством какой клавиши производится удаление с экрана того символа, на который указывает курсор (в DOS)?

а) <Esc>;

б) <Delete>;

в) <Tab>;

г) <Insert>.

9. Какая клавиша обеспечивает перелистывание текста на один экран (страницу) вверх?

а) <PgUp>;

б) <Home>;

в) <PgDn>;

г) <End>.

10. Какая клавиша обеспечивает перелистывание текста на один экран (страницу) вниз?

а)<PgUp>;

б) <End>;

в) <PgDn>;

г) <Home>.

11. Нажатие какой клавиши позволяет вводить только заглавные буквы?

а) <CapsLock>;

б) <Shift>;

в) <Ctrl>;

г) <NumLock>.

12. Какая клавиша переключает режим малой цифровой клавиатуры с цифр на действия и обратно?

а) <Shift>;

б) <Insert>;

в) <NumLock>;

г) <CapsLock>.

13. Для получения дополнительных значений других клавиш необходимо:

а) удерживая клавишу <Shift>, нажать другую клавишу;

б) удерживая клавишу <Alt>, нажать другую клавишу;

в) удерживая клавишу <Ctrl>, нажать другую клавишу;

г) удерживая клавишу <Shift>, <Alt> или <Ctrl>, нажать другую клавишу.

14. Клавиши <F1>—<F12> позволяют:

а) быстро выполнять переход к заданной строке;

б) вызывать помощь по работе с программой;

в) выполнять запрограммированные команды одним нажатием;

г) переключать регистры компьютера.

15. Переход с русского на латинский регистр в системе Windows 95/2002 по умолчанию осуществляется:

а) с помощью сочетания левой клавиши <Alt> плюс левой клавиши <Shift>;

б) с помощью сочетания правой клавиши <Ctrl> плюс правой клавиши <Shift>;

в) только с помощью индикатора клавиатуры;

г) при одновременном нажатии обоих клавиш <Shift>.

2.3. DOS и файловая система

Для начала краткие сведения. Одной из первых операционных систем (ОС) была MS-DOS. Она появилась на рынке еще в 1981 году вместе с первыми компьютерами фирмы IBM.

MS-DOS состоит из следующих составных частей:

- программа *начальной загрузки* (Boot Record);
- *базовая система ввода/вывода* (BIOS), состоящая из двух частей:
  - BIOS — записана на жесткий диск и содержит набор подпрограмм нижнего уровня, осуществляющих непосредственный доступ к аппаратуре;
  - файл Io.sys, содержащий подпрограммы ввода/вывода для конкретной реализации, которые используют или заменяют программы, находящиеся в BIOS. Файл Io.sys вместе с файлом Msdos.sys составляют системное ядро ОС;
- файл Msdos.sys — *часть системного ядра*, отвечающая за:
  - управление файлами;
  - управление ресурсами сети;
  - обработку ошибок;
  - запуск и завершение выполнения программы.
- *командный процессор* — файл Command.com, который организует диалог с пользователем путем обработки команд, выдаваемых ему пользователем (на клавиатуре в виде командной строки) и прикладных программ;
- *драйверы* — загружаются в память компьютера при загрузке операционной системы, их имена указываются в специальном файле — Config.sys. Такая схема облегчает добавление новых устройств и позволяет делать это, не затрагивая системные файлы DOS;
- файл Autoexec.bat, служащий для загрузки прикладных программ ОС (например, Norton Commander) сразу же после загрузки.

Тест 5. Операционная система DOS

1. Классы существующих программных средств:

   а) системные программы, системы программирования, прикладные программы;

   б) операционные системы, системы программирования, игровые программы;

   в) системы программирования, игровые программы, редакторы;

   г) программы системного и прикладного назначения.

2. Операционная система — это:

   а) комплекс программ, организующих управление работой компьютера и его взаимодействие с пользователем;

   б) совокупность основных устройств компьютера;

   в) техническая документация компьютера;

   г) совокупность устройств и программ общего пользования.

3. Операционная система загружается:

   а) только с жесткого диска (винчестера);

   б) только с дискеты;

   в) только с CD-ROM;

   г) ОС может быть загружена со всех вышеперечисленных устройств.

4. Какие команды MS-DOS доступны пользователю?

   а) только внутренние;

   б) только внешние;

   в) внутренние и внешние команды;

   г) никакие команды пользователю недоступны.

5. Для чего нужна системная дискета?

   а) для аварийной загрузки компьютера, установки ОС и проверки на вирусы;

   б) для установки ОС;

   в) для проверки ПК на наличие вирусов;

   г) для низкоуровневого форматирования CD-ROM.

6. Для чего необходимы файлы Autoexec.bat и Config.sys?

   а) для проверки списка файлов и каталогов при загрузке DOS;

   б) для проверки ПК на наличие вирусов;

   в) для запуска программ при загрузке ПК;

   г) для загрузки ОС с установленными параметрами (настройками).

7. Драйверы — это:

   а) техническое устройство;

   б) программы для согласования работы внешних устройств и компьютера;

   в) носители информации;

   г) программы для ознакомления пользователя с принципами устройства компьютера.

8. Командный процессор — это файл ОС:

   а) Command.com;

   б) Autoexec.bat;

   в) Config.sys;

   г) Norton commander.exe.

9. Где находится BIOS?

   а) в оперативном запоминающем устройстве (ОЗУ);

   б) на винчестере;

   в) на CD-ROM;

   г) в постоянном запоминающем устройстве (ПЗУ).

10. Какие функции выполняет файл Command.com?

    а) обрабатывает команды, вводимые пользователем;

    б) хранит все команды операционной системы;

    в) обрабатывает команды и программы, выполняемые при каждом запуске компьютера;

    г) хранит все команды, которые использует пользователь в своей работе.

11. Загрузчик операционной системы служит для:

    а) загрузки программ в оперативную память ЭВМ;

    б) обработки команд, введенных пользователем;

    в) считывания в память модулей операционной системы Io.sys и Msdos.sys;

    г) подключения устройств ввода/вывода.

12. BIOS — это:

    а) игровая программа;

    б) диалоговая оболочка;

    в) базовая система ввода/вывода;

    г) командный язык операционной системы.

13. В состав ОС не входит:
    а) BIOS;
    б) программа-загрузчик;
    в) драйверы;
    г) ядро ОС.
14. Какой из нижеперечисленных файлов не входит в ядро DOS:
    а) Io.sys;
    б) Config.sys;
    в) Command.com;
    г) Msdos.sys.
15. Что из нижеперечисленного не является операционной системой:
    а) Linux;
    б) OS/2;
    в) UNIX;
    г) MS Office.

Тест 6. Файловая структура

1. Информация внутри ПК хранится в виде:
   а) файлов, находящихся в каталогах (папках);
   б) в цифровом виде;
   в) рисунков, текстов, звуков и видео;
   г) только в текстовом виде.
2. Файловая система на диске имеет:
   а) линейную структуру;
   б) иерархическую структуру;
   в) не имеет структуры;
   г) связную линейно-кусочную структуру.
3. Что такое каталог?
   а) место на диске, отведенное для хранения файлов и каталогов;
   б) единица хранения информации;
   в) поименованная совокупность файлов и каталогов;
   г) периферийное устройство.

4. Кто присваивает имена файлам и каталогам?
   а) только операционная система;
   б) файлы и каталоги не имеют имени;
   в) пользователь, а также некоторые программы;
   г) программы, при установке (инсталяции) на диск.
5. Файл — это:
   а) программа;
   б) область внешней памяти;
   в) именованная совокупность данных;
   г) текстовая информация.
6. Расширение имени файла:
   а) указывает тип файла;
   б) характеризует информационный объем файла;
   в) содержит любое количество символов;
   г) может характеризовать тип файла и способ представления информации.
7. Расширение файла должно содержать:
   а) 8 символов;
   б) обязательно 3 символа;
   в) меньше или равно 3 символам;
   г) не более 256 символов.
8. Исполняемые файлы имеют расширение:
   а) com exe bat;
   б) com txt pas;
   в) exe com txt;
   г) exe com bak.
9. Какое имя файла и расширения правильное для MS-DOS?
   а) AN21CC.TEXT;
   б) 1310NN1712.TUT;
   в) LBN42.TXT;
   г) RTR.ORT.TV.
10. В высказывание "Каталог содержит информацию о ..., хранящихся в ..." вместо многоточия вставить правильные выражения.
   а) файлах, оперативной памяти;
   б) программах, внешней памяти;

в) файлах, внешней памяти;

г) файлах, ПЗУ.

11. Какие буквы и символы можно использовать в имени файла (каталога) в MS-DOS?

а) длина имени файла и каталога не ограничена, можно использовать любые символы;

б) только специальные символы и знаки препинания;

в) только латинские буквы, цифры, знак подчеркивания и дефиса;

г) только буквы русского алфавита и цифры.

12. Из чего состоит полное имя файла?

а) из букв латинского алфавита и цифр;

б) только из букв латинского алфавита и разделителя (точки);

в) из имени, разделителя и расширения;

г) файлы имеют только короткие имена.

13. Что такое путь?

а) указание местонахождения файла или каталога при помощи указания диска и последовательного перечисления вложенных каталогов;

б) перечисление списка дисков;

в) перечисление каталогов;

г) список файлов и каталогов.

14. Запись USSR?.* означает:

а) все файлы с именем USSR;

б) все файлы с именем USSR с любым расширением;

в) все файлы из 5 символов, у которых первые 4 символа USSR;

г) ничего не означает.

15. Какие файлы соответствуют маске ??P*.A??:

а) PPEPSI.ABC;

б) PEDDY.A1;

в) PEPPER.ARJ;

г) PEPSI.A1;

д) PEPPY.A7F;

е) CAPITAL.A3A;

ж) SUPPORT.A1;

з) TUPIK.ABC.

16. Все файлы с именем, начинающимся с DEK и состоящим не более чем из 6 символов, независимо от расширения обозначаются как:
    а) DEK*.*
    б) DEK???.?
    в) DEK.*
    г) DEK???.*
    д) DEK?
    е) DE????.*
17. Текущий каталог — это каталог:
    а) в котором хранятся все программы операционной системы;
    б) объем которого изменяется при работе компьютера;
    в) с которым работает или работал пользователь на данном диске;
    г) в котором находятся файлы, созданные пользователем.
18. Определите тип файла по его расширению:

а) q*.txt	з) q*.bat
б) q*.mp3	и) q*.zip
в) q*.avi	к) q*.sys
г) q*.bmp	л) q*.mpg
д) q*.wav	м) q*.hlp
е) q*.c	н) q*.exe
ж) q*.xls	

2.3.1. Основные внутренние команды DOS

Внутренние команды DOS выполняет командный процессор Command.com.

- Команда смены текущего дисковода.
  - `a:` — переход на дисковод A:
  - `c:` — переход на дисковод C:
  - `d:` — переход на дисковод D:
- Команда вывода оглавления каталога.
  - `dir /p` — поэкранный (постраничный) вывод оглавления каталога, если оглавление очень большое.
  - `dir a:\` — команда вывода оглавления корневого каталога на диске A:.

- `dir>dir.txt` — команда позволяет сохранить оглавление текущего каталога в виде текстового файла Dir.txt.

❒ Команда смены текущего каталога.

`cd имя_каталога`

- `cd\` — переход в корневой каталог текущего диска.
- `cd..` — переход из подкаталога в предыдущий каталог.

❒ Команда создания каталога.

`md имя_каталога`

❒ Команда удаления каталога (удалить можно только *пустой* каталог!).

`rd имя_каталога`

❒ Команда вывода содержимого текстового файла на экран.

`type имя_файла`

❒ Команда очистки экрана монитора.

`cls`

❒ Команда копирования файлов.

- `copy имя_файла имя_каталога` — копирование файла в каталог.
- `copy имя_файла prn` — распечатка файла на принтере.
- Создание текстового файла.

  `copy con: имя_файла`, нажать <Enter>, ввести построчно текст, в конце каждой строки <Enter>, затем в конце <F6> или <Ctrl>+<Z> и <Enter>.
- Объединение содержимого двух и более текстовых файлов (конкатенация файлов).

  `copy имя_файла_1 + имя_файла_2 имя_нового_файла`

❒ Команда удаления файла.

`del имя_файла`

❒ Команда переименования файла.

`ren старое_имя_файла новое_имя_файла`

❒ Команда вывода версии DOS.

`ver`

❒ Команда вывода метки диска.

`vol`

❒ Команда ввода текущей даты.

`date`

❒ Команда ввода текущего времени.

`time`

2.3.2. Основные внешние команды DOS (для версии MS-DOS 6.22)

Внешние команды DOS выполняют вспомогательные программы (утилиты), расположенные в каталоге DOS на диске C:.

- `format имя_диска`
  - `format a:` — форматирование дискеты.
  - `format имя_диска: /s` — создание системного диска.

  Предупреждение

  Будьте внимательны: при форматировании диска (дискеты) вся информация на нем уничтожается!

- `fdisk`

  Разбиение жесткого диска на логические разделы (логические диски C:, D:, E: и т. д.).

  Предупреждение

  Будьте внимательны: при разбиении жесткого диска на разделы вся информация на нем уничтожается!

- `sys имя_диска`

  Перенос файлов ядра DOS на диск (дискету), создание системного диска (дискеты). Используется также для смены (обновления версии) DOS.

- `mem`

  Вывод карты оперативной памяти. Чаще используются команды `mem/c/p` или `mem/d/p` (ключ `p` — постраничный вывод информации).

- `chkdsk c: /f`

  Проверка файловой системы и восстановление потерянных кластеров.

- `scandisk`

  Проверка файловой системы и диска на наличие логических и физических сбоев.

- `label имя_диска`

  Создание или обновление метки на диске, дискете.

- `print имя_файла prn`

  Печать файла на принтере в фоновом режиме. Отмена фоновой печати: `print /t`

- `defrag c: /f`

  Устранение фрагментации файлов и оптимизация размещения информации на жестком диске.

Создание файла в MS-DOS

Для создания файла в MS-DOS необходимо:

1. Ввести команду в командную строку: `copy con: имя_файла` и нажать <Enter>.

   Пример: `copy con: c:\mark\proba.txt`

2. Ввести текст. В конце каждой строки нажимать <Enter>. В конце текста <Ctrl>+<Z> и <Enter>. Будет создан файл Proba.txt в каталоге MARK. Имя файла надо указывать с полным путем.

Если файл создается в текущем каталоге, то можно не указывать полный путь (маршрут).

Пример: `copy con: name.txt` и далее как указано выше в п. 2.

Практикум 1. Работа с диском

1. Вставьте диск в дисковод A:.
2. Просмотрите его содержимое.
3. Проверьте поверхность диска на наличие ошибок.
4. Дефрагментируйте диск.
5. Отформатируйте диск, сделав его системным.
6. Задайте метку диска.
7. Загрузите компьютер с созданной системной дискеты.
8. Установите дату 17 декабря 2003 года.
9. Установите время 10 часов 13 минут.
10. Просмотрите метку диска.
11. Перейдите на диск C:.
12. Создайте на нем каталог REPORT.
13. Создайте в этом каталоге текстовый файл Markfive.txt с кратким отчетом о проделанной работе.
14. Узнайте объем, занимаемый созданным файлом.
15. Скопируйте каталог REPORT на диск A:.
16. Покажите результат работы преподавателю.
17. Уничтожьте следы вашей работы на дисках A: и C:.

Практикум 2. Команды MS-DOS работы с дисками, файлами и каталогами

1. Если вы находитесь в Windows, то перезагрузите компьютер в режим эмуляции MS-DOS.
2. Перейдите на диск C:.
3. Очистите экран.
4. Просмотрите содержимое диска C:.
5. Создайте каталог ZADANIE1.
6. Откройте созданный каталог.
7. Просмотрите его содержимое и убедиться в том, что он пуст.
8. Создайте в нем два каталога — DOG и CAT.
9. Откройте каталог DOG.
10. Создайте в нем с клавиатуры файл под именем Bobik.txt. В качестве текста — фраза "Собака бывает кусачей только от жизни собачьей".
11. Выйдите из каталога DOG в предыдущий.
12. Откройте каталог CAT.
13. Создайте в нем с клавиатуры файл под именем Murzik.txt. В качестве текста — фраза "Кошки гуляют сами по себе".
14. Выйдите из каталога CAT в предыдущий.
15. Скопируйте файл Murzik.txt в каталог DOG и файл Bobik.txt в каталог CAT.
16. Откройте каталог DOG.
17. Переименуйте файл Murzik.txt в файл под именем Barsik.txt.
18. Выйдите из каталога DOG в предыдущий.
19. Откройте каталог CAT.
20. Переименуйте файл Bobik.txt в файл под именем Barbos.txt.
21. Выведите на экран содержимое файла Barbos.txt.
22. Выведите в корневой каталог диска C:.
23. Вставьте диск A:.
24. Перейдите на диск A:.
25. Просмотрите его содержимое.
26. Скопируйте каталог ZADANIE1 на диск A:.
27. Переименуйте каталог ZADANIE1 в CAT&DOG.

28. Просмотрите содержимое диска A:. Должна получиться следующая структура файлов и каталогов (рис. 2.6).

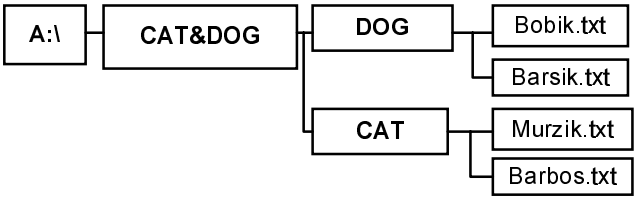


Рис. 2.6. Структура файлов и каталогов

29. Покажите результаты работы преподавателю.
30. Удалите все созданные файлы и каталоги.

Тест 7. Команды MS-DOS

1. Внутренние команды MS-DOS — это:

   а) команды, предназначенные для создания файлов и каталогов;

   б) команды, встроенные в DOS;

   в) команды, которые имеют расширения sys, exe, com;

   г) команды для работы с дисками.

2. Какие команды DOS называются внешними?

   а) команды, предназначенные только для работы с периферийными устройствами;

   б) команды, хранящиеся на диске и вызываемые по мере необходимости;

   в) все команды, которые можно реализовать с помощью DOS;

   г) команды для удаления информации.

3. Для проверки диска C: на наличие ошибок в системе MS-DOS необходимо использовать:

   а) `defrag c:/f`;

   б) `scandisk`;

   в) `format c:`;

   г) `autoexec c:`.

4. Как отформатировать дискету с переносом системных файлов?

   а) `dir a:/s`;

   б) `scandisk a:/s`;

в) `format a:/s`;

г) `sys c: a:`.

5. Для чего нужна системная дискета?

а) для аварийной загрузки компьютера, установки ОС и проверки на вирусы;

б) для установки ОС;

в) для проверки ПК на наличие вирусов;

г) для разбиения жесткого диска на несколько логических дисков.

6. Как запустить программу из командной строки?

а) перейти в каталог, где находится ПО, и ввести имя исполняемого файла, нажать ввод (<Enter>);

б) перейти в каталог ПО и нажать ввод (<Enter>);

в) нажать клавишу <Enter>, затем ввести имя файла, с которого запускается программа;

г) ввести команду `start`, а затем имя исполняемого файла.

7. Как установить дату с помощью команд MS-DOS?

а) ввести команду `date` и по образцу ввести необходимую информацию;

б) в командной строке напечатать текущую дату и нажать ввод (<Enter>);

в) ввести `time` и, нажав ввода (<Enter>), установить текущую дату с клавиатуры.

8. Как установить системное время с помощью команд MS-DOS?

а) ввести команду `time` и по образцу ввести необходимую информацию;

б) в командной строке напечатать текущее время и нажать ввод (<Enter>);

в) ввести `date`, нажать клавишу ввода (<Enter>) и ввести данные с клавиатуры;

г) командой `format time`.

9. Как перейти в корневой (главный) каталог диска?

а) `cd\`;

б) `cd dos`;

в) `dir`;

г) `time`.

10. Как вывести на экран список файлов и каталогов текущего каталога?

    а) `dir`;

    б) `time`;

    в) `cd\`;

    г) `cd dos`.

11. Как перейти в корневой каталог из текущего?

    а) `cd`;

    б) `cd/`;

    в) `cd.`;

    г) `cd\`.

12. Как сменить (перейти на другой) диск?

    а) для перехода на другой диск необходимо ввести команду `cd\`;

    б) для перехода на другой диск необходимо ввести имя диска и нажать ввод;

    в) для перехода на другой диск необходимо нажать ввод (<Enter>) и ввести имя диска;

    г) для перехода на другой диск необходимо воспользоваться командой `goto` с указанием имени диска.

13. Для выполнения полной оптимизации диска C: в системе MS-DOS необходимо использовать команду:

    а) `defrag c:/f`;

    б) `scandisk c:`;

    в) `format c:`;

    г) `aidstest c:`.

14. Что произойдет по команде `C:\>copy a:\DREAM\*.txt C:\REAL`?

    а) будут скопированы все файлы с диска A: в каталог REAL диска C:;

    б) будут скопированы все текстовые файлы с диска A: в каталог REAL диска C:;

    в) будут скопированы все текстовые файлы из каталога DREAM диска A: в каталог REAL диска C:;

    г) будет скопирован файл DREAM с диска A: на диск C: под именем REAL.

15. Что произойдет после выполнения команды `D:\>dir C:\DOS\f*.* /p`?

    а) на экран будут выведены все файлы каталога DOS диска C:;

    б) на экран будут выведена первая страница всех файлов, начинающихся на букву f каталога DOS диска C:;

в) на экран будут выведены все файлы каталога DOS, начинающиеся на букву f и имеющие в расширении букву p;

г) на экран будет выведено сообщение о некорректности введенной команды.

2.4. Оболочка Norton Commander

Основные функциональные клавиши Norton Commander:

- <F1> — Вызов помощи.
- <F2> — Меню пользователя.
- <F3> — Просмотр файла.
- <F4> — Редактирование файла.
- <F5> — Копирование файла (группы файлов).
- <F6> — Переименование или перенос файла (группы файлов).
- <F7> — Создание каталога.
- <F8> — Удаление файла или каталога (группы файлов).
- <F9> — Верхнее меню всех команд и возможностей Norton Commander.
- <F10> — Выход из Norton Commander.
- <Ctrl>+<F1> — Убрать/восстановить левую панель экрана.
- <Ctrl>+<F2> — Убрать/восстановить правую панель экрана.
- <Ctrl>+<F3> — Сортировка файлов по имени (по алфавиту).
- <Ctrl>+<F4> — Сортировка файлов по расширению.
- <Ctrl>+<F5> — Сортировка файлов по времени создания.
- <Ctrl>+<F6> — Сортировка файлов по размеру.
- <Ctrl>+<F7> — Сортировка файлов не устанавливается.
- <Ctrl>+<F9> — Печать файла.
- <Shift>+<F4> — Создание текстового файла с клавиатуры. Имя файла задается.
- <Shift>+F9> — Сохранение настроек Norton Commander.
- <Alt>+F1> (<Alt>+<F2>) — Переключение на другой дисковод на левой (правой) панели.
- <Alt>+<F5> — создание архива.
- <Alt>+<F7> — Быстрый поиск файла на диске.
- <Alt>+<F9> — Переключение режима монитора с 25 на 43 линии.

- <Insert> — Выделение файла (файлов) для объединения их в группу и выполнения над ними каких-либо действий.
- <*> (на малой цифровой клавиатуре) — Выделение всех файлов в каталоге или инверсия выделения файлов. Выделение группы файлов по шаблону.
- <серый +>. <Enter> — Выделение всех файлов в каталоге.
- <серый −>. <Enter> — Отмена выделения файлов.
- <Ctrl>+<O> — Убрать/восстановить панели экрана.
- <Ctrl>+<U> — Поменять панели местами.
- <Ctrl>+<P> — Убрать/восстановить не текущую панель экрана.
- <Ctrl>+<L> — Вывод информационной панели диска (дискеты).
- <Ctrl>+<Q> — Быстрый просмотр файла, информация о содержимом каталога.
- <Ctrl>+<E> — Вывод ранее выполненной команды из буфера History.
- <Ctrl>+<X> — Вывод следующей выполненной команды из буфера History.
- <Ctrl>+<J> (или <Ctrl>+<Enter>) — Перенос выделенного курсором файла в командную строку.
- <Ctrl>+<R> — Быстрое обновление информации на панели диска (дискеты).
- <Tab> — Перемещение курсора с левой панели на правую и обратно.
- <Esc> — Отмена команд в Norton Commander.
- Поиск подстроки при просмотре файла — <F3>, <F7>, ввести искомое слово и <Enter>.

Вопросы (продолжение)

18. В чем удобство работы в NC по сравнению с DOS?
19. Какие отличия вы знаете в оболочке Volkov Commander по сравнению с классическим Norton Commander?
20. Возможно ли в NC вызвать оглавление сразу трех дисков? Если да, то как?
21. Как отсортировать файлы в NC по размеру, по типу, по алфавиту?
22. Как найти текст в NC, например, как найти документы, содержащие словосочетание "Зенит — чемпион!"?
23. Как быстро найти все файлы, длина имени которых не более 5 символов, а второй и третий символы — "or"?

24. Зачем бывает нужно убрать панели NC с экрана? Как это сделать?
25. Как создать текстовый файл с клавиатуры? Как после его создания внести туда коррективы?
26. Как напечатать выбранный файл?
27. Почему Norton Commander так называется? Когда эта оболочка впервые появилась?

Практикум 3. Работа с Norton Commander

1. Вставьте диск в дисковод A:.
2. Вызовите оглавление диска A: в левой панели NC, а диска C: — в правой.
3. Определите объем диска A: и количество свободного дискового пространства.
4. Настройте панель с диском A: на отображение полной информации о файлах.
5. Создайте на диске C: каталог ANIMALS.
6. Создайте в нем два подкаталога WILD и HOME.
7. Создайте в подкаталоге WILD два текстовых файла − Wolf.txt (в качестве текста — фраза "Сколько волка ни корми, а он все равно в лес смотрит") и Lion.txt (в качестве текста фраза "Лев — царь зверей").
8. Создайте в подкаталоге HOME два текстовых файла − Goat.txt (в качестве текста − фраза "Любовь зла — полюбишь и козла") и Pig.txt (в качестве текста фраза "Посади свинью за стол — она и ноги на стол!").
9. Создайте в подкаталоге HOME еще один подкаталог FFF.
10. Найдите на диске C: все файлы, начинающиеся на f.
11. Скопируйте в подкаталог FFF три из найденных файлов, занимающих в общей сложности не более 500 Кбайт.
12. Переименуйте эти файлы последовательно в File1.txt, File2.doc и File3.wri
13. Отредактируйте файл Pig.txt, добавив туда фразу "Гусь свинье не товарищ". Переименуйте файл в Goose.doc.
14. Определите суммарный информационный объем каталога ANIMALS.
15. Перенесите его на диск A: под именем ZVERI.

На диске A: должна получиться следующая структура файлов и каталогов (рис. 2.7).

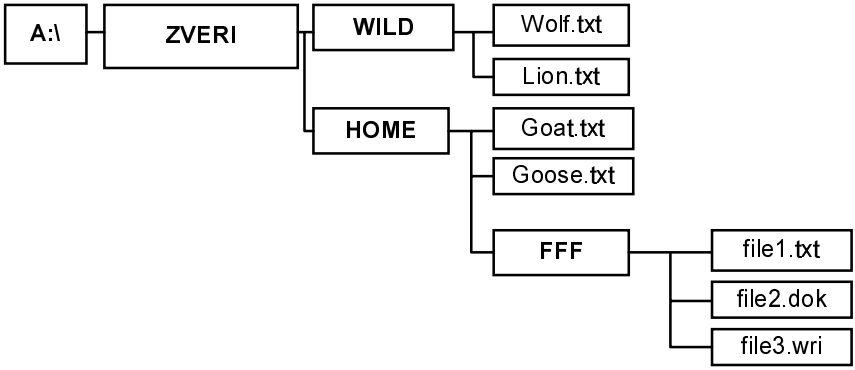


Рис. 2.7. Структура файлов и каталогов диска A:

16. Примените к файлам File?.* в подкаталоге FFF атрибут "скрытый", а к файлу Lion.txt – "только для чтения"
17. Заархивируйте каталог ANIMALS на диске C: под именем Paket.arj.
18. Сравните каталоги ANIMALS и ZVERI по объему.
19. Создайте текстовый файл Report.txt с кратким отчетом о проделанной работе.
20. Поместите его на диск A: в корневой каталог.
21. Покажите результаты преподавателю. Удалите следы вашего пребывания на дисках A: и C:.

Тест 8. Знаете ли вы NC?

1. Как запустить NC?

   а) ввести имя исполняемого файла в командной строке DOS и нажать ввод;

   б) NC запускается автоматически при старте ОС;

   в) для запуска NC никаких команд не требуется;

   г) запустить файл Autoexec.bat.

2. Как скопировать файл (каталог) в NC?

   а) с помощью команды меню **Файл | Копировать** или нажать <F5>;

   б) выделить объект, нажать <F5> и указать, в случае необходимости, путь копирования;

   в) воспользоваться клавишей <F1> (Помощь), далее следовать инструкциям;

   г) выделить объект, нажать <F6> и указать, в случае необходимости, путь копирования.

3. Как изменить имя файла (каталога)?

   а) выделить объект, далее клавиша <F6>, в диалоговом окне ввести новое имя;

   б) выделив файл (каталог), использовать клавишу <F5>;

   в) установить указатель на файл (каталог), ввести новое имя в командной строке и нажать <Enter>;

   г) воспользоваться клавишей <F1> (Помощь), далее следовать инструкциям.

4. Каким образом можно запускать программы в NC?

   а) только из командной строки (ввести команду и нажать <Enter>);

   б) только в каталоге, где находится исполняемый файл программы (перейти в каталог, выделить файл и нажать <Enter>);

   в) с помощью исполняемого файла или командной строки;

   г) найти исполняемый файл, войти в верхнее меню (<F9>) и выбрать там команду **Исполнить**.

5. Как сохранить настройки программы NC?

   а) использовать сочетание клавиш <Ctrl>+<O>;

   б) <Shift>+<F9> или воспользоваться верхним меню;

   в) настройки программы сохраняются автоматически;

   г) нажать <Alt>+<F7>, в появившемся диалоговом окне набрать команду `Remember` и нажать <Enter>.

6. Как можно быстрее всего выделить все файлы?

   а) с помощью мыши;

   б) используя клавишу <Insert>;

   в) использовать "серый" плюс;

   г) через меню пользователя (<F2>);

7. Как обратить (инвертировать) выделение?

   а) <Enter>;

   б) <Esc>;

   в) <Insert>;

   г) "серая" звездочка.

8. Как отменить появление диалогового окна?

   а) <Insert>;

   б) <Esc>;

в) <Enter>;

г) <Пробел>.

9. Как создать новый каталог?

а) выбрать нужный диск (каталог), нажать <F7>, ввести в диалоговом окне имя и нажать ввод;

б) нажать <F7>, ввести в диалоговом окне имя, подтвердить команду;

в) воспользоваться помощью — <F1>;

г) воспользоваться командой `copy con имя_каталога`.

10. Для выбора нужного диска в NC необходимо:

а) использовать сочетания <Alt>+<F1> или <Alt>+<F2> или вызвать команду из верхнего меню **Левая (Правая) панель | Сменить диск**;

б) использовать сочетания <Ctrl>+<F1> или <Ctrl>+<F2>;

в) ввести в командной строке команду `dir имя_диска`;

г) нажать <F10>, <Enter> и набрать имя нужного диска.

11. Как выделить только несколько файлов?

а) использовать <Insert> или правую кнопку мыши;

б) воспользоваться командами верхнего меню;

в) нажать <F1> и пометить нужные файлы;

г) выделить файлы с помощью клавиши <Delete>.

12. Как восстановить панели, не показываемые на экране?

а) <Ctrl>+<L>;

б) <Shift>+<F9>;

в) <Ctrl>+<O>;

г) <Alt>+<Enter>.

13. Как выключить и включить строку подсказки?

а) <Ctrl>+<B>;

б) <Ctrl>+<O>;

в) <Ctrl>+<L>;

г) <Alt>+<Enter>.

14. Как повторить команду, введенную ранее в командной строке?

а) <Ctrl>+<E>;

б) <Alt>+<F8>;

в) <Ctrl>+<E>;

г) <Shift>+<F9>.

15. Как найти файл (документ)?
    а) <Alt>+<F7>;
    б) <Alt>+<F1>;
    в) <Alt>+<F1>;
    г) только с помощью команды `find`.
16. Как скопировать имя текущего файла в командную строку?
    а) <Alt>+<Enter>;
    б) <Ctrl>+<Enter>;
    в) <Shift>+<Enter>;
    г) <Tab>+<Enter>.
17. Как очистить командную строку?
    а) <Ctrl>+<O>;
    б) <Esc>;
    в) <Alt>+<F1>;
    г) <Alt>+<F2>.
18. Как перейти из одной панели в другую?
    а) <Esc>;
    б) <Tab>;
    в) <Enter>;
    г) <Shift>+<Home> или <Shift>+<End>.
19. Как удалить сразу несколько каталогов?
    а) выделить и нажать <Delete>;
    б) выделить, <F8>, установить флажок **Включая подкаталоги** и <Enter>;
    в) выделить каталоги и нажать клавишу <F8>;
    г) при помощи команды `format c:\`.
20. Объясните принцип действия программ-архиваторов:
    а) переписывают содержимое файлов (данные) в один файл;
    б) сжимают объем файлов и сохраняют их в виде архивного файла;
    в) используются для копирования информации на дискеты;
    г) пересылают архивируемые файлы на сервер **www.fsb.ru**.
21. Сочетание <Ctrl>+<O> выполняет в NC следующую операцию:
    а) архивацию выделенных файлов (каталогов);
    б) убирает и восстанавливает панели на экране;

в) выключает панели совсем;

г) переводит компьютер в "спящий" режим.

2.5. Windows и программа Проводник

Windows — графическая многозадачная операционная система, обладающая огромными возможностями для работы в совершенно разных областях компьютерных технологий и очень простая в освоении. Поэтому не будем долго останавливаться на сведениях о ней — тут нужно говорить либо очень много, либо ничего. Надеемся, на уроках теории вы это уже слышали, а мы при помощи вопросов, практикумов и тестов проверим усвоение вами полученных знаний.

Вопросы (продолжение)

28. Какая версия Windows появилась первой? Каковы были ее достоинства и недостатки? Каковы были системные требования для ее работы?
29. Что такое многозадачность?
30. Какая версия Windows имела рабочее название Chicago?
31. Какая версия Windows имела рабочее название Memphis?
32. Как называется версия Windows 2002?
33. Что является основными элементами графического интерфейса Windows?
34. Что такое окно? Какие типы окон в Windows бывают?
35. В чем различие между ярлыком и значком?
36. В чем различие приложения, документа, задачи?
37. Какие способы запуска приложений вы можете назвать?
38. Сколько способов создания папок вы знаете?
39. Сколько способов копирования файлов и папок вы можете назвать и продемонстрировать?
40. Как изменить свойства окна? Рабочего стола?
41. Какие стандартные программы, входящие в состав Windows, вы знаете? Зачем нужны служебные программы?
42. Входит ли в состав Windows пакет прикладных программ Microsoft Office?
43. Какие мультимедиа-приложения входят в состав Windows?
44. Как вызвать контекстное меню?

45. Как можно добавить новую раскладку клавиатуры, изменить сочетание клавиш для переключения раскладок?
46. При перетаскивании объектов при помощи мыши какую клавишу нужно удерживать, чтобы получилось перемещение (Move), а не копирование (Copy)?
47. Какова последовательность действий при установке Windows XP, если у вас стоит Windows 98?
48. Входят ли в состав Windows программы-архиваторы WinZip, WinRar?

Практикум 4. Создание системной дискеты

1. Произведите полное форматирование дискеты A: без переноса системных файлов.
2. Убедитесь по отчету о форматировании, что дискета не содержит плохих секторов.
3. Используя служебные программы обслуживания дисков, проверьте поверхность дискеты и оптимизируйте их.
4. Используя возможности Windows, создайте системную дискету для аварийной загрузки компьютера в случае неполадок в ее работе. Пометьте ее как SYSTEM.
5. Загрузите компьютер с системной дискеты.
6. Просмотрите ее содержимое.
7. Создайте на диске C: текстовый файл Report.txt о тех файлах системной дискеты, назначение которых вам известно.
8. Сдайте работу преподавателю.

Практикум 5. Создание папок и документов стандартными средствами Windows

1. Создайте на диске C: папку под именем `FRIENDS`.
2. Создайте в папке FRIENDS две не вложенные друг в друга папки и назовите их соответственно ПУШКИН и ГОГОЛЬ.
3. Переименуйте папку FRIENDS в Мои друзья.
4. Создайте на диске C: в папке Мои документы папку, которая будет называться датой сегодняшнего дня, например 171203.
5. Переместите папку Мои друзья в созданную папку.
6. Скопируйте папки ПУШКИН и ГОГОЛЬ с диска C: на диск A: за одно действие.

7. Узнайте, сколько подпапок в папке Мои документы диска С:, не открывая ее.
8. Запустите программу-приложение Блокнот.
9. Создайте текстовый файл (в качестве текста — загадка "Сто одежек и все без застежек"), сохраните его под именем Одна загадка в папке ПУШКИН.
10. Переименуйте созданный файл, дав ему имя Моя загадка.
11. Откройте файл Моя загадка, добавьте еще одну загадку: "Два конца, два крыльца — посередине гвоздик" и сохраните текст под именем Две загадки в папку ГОГОЛЬ.
12. Узнайте информационный объем обоих файлов.
13. Отправьте файл Моя загадка в Корзину.
14. Восстановите удаленный в Корзину файл.
15. Сделайте ярлык для файла Две загадки. Вытащите его на рабочий стол.
16. Сдайте работу преподавателю.

Практикум 6. Создание ярлыков

1. Создайте ярлык для стандартной игры "Сапер" (Winmine).
2. Вытащите его на рабочий стол.
3. Измените значок. Проверьте, работает ли программа.
4. Создайте ярлык для любой программы DOS.
5. Настройте его на полноэкранный режим, автоматическое закрытие окна после окончания работы.
6. Измените значок на свое усмотрение.
7. Проверьте работоспособность ярлыка.
8. Сдайте работу преподавателю.

Практикум 7. Программа Проводник

Выполните *Практикум 3* из *главы 2* по пункт 16 включительно применительно к Windows.

Практикум 8. Элементы окна Windows

Посмотрите на рис. 2.8 и напишите, как правильно называются помеченные номерами элементы окна Windows.

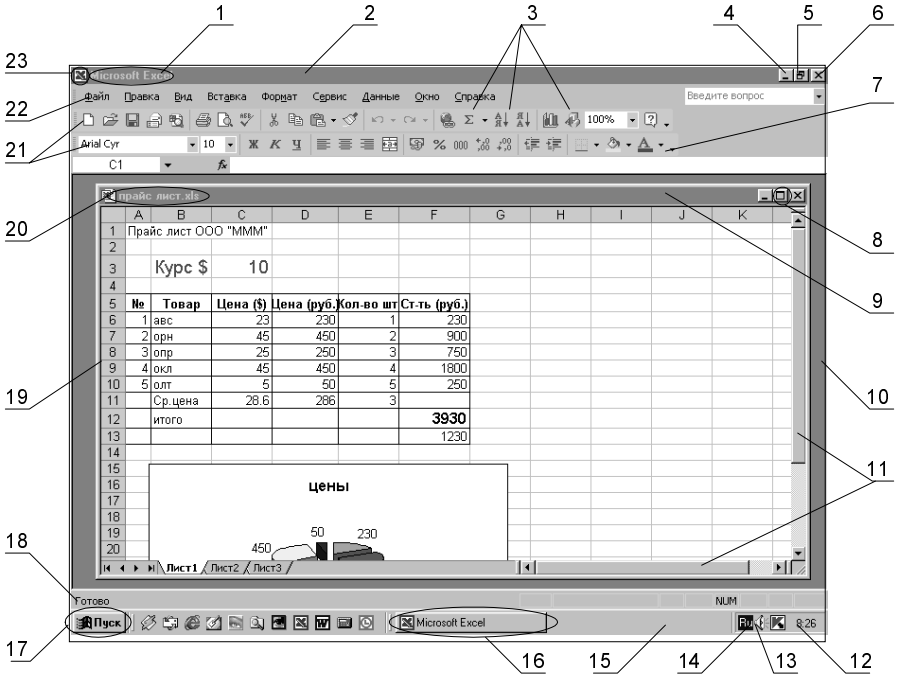


Рис. 2.8. Элементы окна Windows

Тест 9. Знаете ли вы Windows?

1. Стандартный интерфейс ОС Windows не имеет:

   а) рабочего поля, панели инструментов;

   б) справочной системы;

   в) элементов управления (**Свернуть**, **Развернуть**, **Закрыть** и т. д.);

   г) строки ввода команды.

2. Технология Plug and Play:

   а) позволяет синхронизировать работу компьютера и устройства;

   б) позволяет новым устройствам автоматически настраиваться под конфигурацию данного компьютера;

   в) используется вместо внешних устройств;

   г) позволяет проигрывать музыкальные CD.

3. Ярлык — это:

   а) копия файла, папки или программы;

   б) каталог;

в) графическое изображение файла, папки или программы;

г) перемещенный файл, папка или программа.

4. ОС Windows предоставляет возможность работать с мультимедиа-информацией. К таким программам не относится:

а) VolumeControl (Регулятор звука);

б) Scan Disk (Диагностика);

в) Sound Recorder (Фонограф);

г) CD-Player (Лазерный проигрыватель).

5. Папка, в которую временно попадают удаленные объекты, называется:

а) Корзина;

б) Оперативная;

в) Портфель;

г) Блокнот.

6. ОС Windows поддерживает длинные имена файлов. Длинным именем файла считается:

а) любое имя файла без ограничения на количество символов в имени файла;

б) любое имя файла латинскими буквами, не превышающее 255 символов;

в) любое имя файла, не превышающее 255 символов;

г) любое имя файла с расширением doc.

7. К функциональным возможностям ОС Windows не относится;

а) поддержка мультимедиа;

б) технология Plug and Play;

в) охлаждение процессора;

г) многозадачность.

8. При удалении файлов активизируется Корзина. Что происходит?

а) при удалении файлы стираются с диска, а в Корзине хранится лишь их список;

б) файлы не удаляются с диска, они хранятся в папке Корзина, но в папке, где они находились, их уже нет;

в) файлы не удаляются с диска, они хранятся в той же папке, что и раньше;

г) файлы перекодируются в кодировку КОИ-8.

9. Как перейти с русского на латинский шрифт и обратно?

   а) с помощью стандартного сочетания левых клавиш <Alt>+<Shift> (или <Ctrl>+<Shift>) или через индикатор клавиатуры на панели задач;

   б) только с помощью стандартного сочетания клавиш;

   в) только с помощью индикатора клавиатуры;

   г) нажать клавишу <NumLock>.

10. Сколько раскладок клавиатуры может быть установлено?

    а) количество, необходимое пользователю;

    б) только две (русская и английская);

    в) не более трех (обязательно включая русскую и английскую);

    г) не более любых четырех.

11. Для управления состоянием окна используются опции:

    а) **Закрыть**, **Развернуть**, **Свернуть**, **Восстановить**;

    б) **Закрыть**, **Открыть**, **Переместить**;

    в) только **Открыть** и **Закрыть**;

    г) **Закрыть**, **Развернуть**, **Изменить размеры**.

12. Как развернуть окно на весь экран?

    а) щелкнуть по кнопке **Пуск** на панели задач;

    б) двойной щелчок в заголовке окна или кнопка **Развернуть**;

    в) только кнопкой **Восстановить**;

    г) только сочетанием клавиш <Alt>+<F4>.

13. Как восстановить свернутое окно?

    а) с помощью щелчка на кнопке программы на панели задач, кнопка **Восстановить** или контекстное меню;

    б) только с помощью кнопки **Восстановить**;

    в) только сочетанием клавиш <Alt>+<F4>;

    г) выбрать команду **Выход** в меню **Файл**.

14. Какое окно считается активным?

    а) с выделенным заголовком или нажатой кнопкой на панели задач;

    б) окно, развернутое на весь экран;

    в) окно, ближнее к верхней части экрана;

    г) окно, ближнее к клавише **Пуск** на панели задач.

15. Как изменить размеры окна с помощью манипулятора "мышь"?

    а) установив курсор мыши на уголок или любую из границ окна, изменить размер;

    б) операция доступна только через меню управления окном;

    в) операция выполняется с помощью кнопок управления окном;

    г) при помощи клавиш <PgUp> и <PgDn>.

16. Как переместить окно на рабочем столе?

    а) установить курсор мыши на заголовок и, не отпуская левую кнопку мыши, переместить в нужную позицию;

    б) с помощью клавиш управления курсором указать нужную позицию;

    в) указать позицию и нажать клавишу <Enter>;

    г) перемещение окна не предусмотрено.

17. Какие элементы (кнопки и значки) могут находиться на панели задач Windows?

    а) **Пуск**, быстрый запуск, кнопки программ, индикатор клавиатуры, регулятор звука, часы;

    б) только кнопка **Пуск**, индикатор клавиатуры и регулятор звука;

    в) только кнопка **Пуск**;

    г) наличие и отсутствие элементов регулирует пользователь.

18. Как переключаться между программами?

    а) с помощью <Alt>+<Tab>, диспетчера задач или щелчка на кнопке программы, находящейся на панели задач;

    б) только с помощью диспетчера задач;

    в) только с помощью сочетания клавиш <Alt>+<Tab>;

    г) с помощью щелчка на кнопке программы, находящейся на панели задач.

19. Как производится обмен данными между различными приложениями Windows?

    а) с помощью файлов (документов);

    б) с помощью преподавателя;

    в) посредством команд меню **Импорт/Экспорт**;

    г) с помощью буфера обмена.

20. Назначение пункта меню **Правка** стандартного окна Windows:

    а) работа с буфером обмена и выделение объектов;

    б) вывод справки по работе с буфером обмена;

в) выполнение операций копирования и выделения объектов с помощью сочетаний клавиш;

г) изменение внешнего вида окна.

21. Как изменить свойства рабочего стола?

а) вызвать контекстно-зависимое меню, выбрать пункт **Свойства** или через панель управления;

б) изменение свойств рабочего стола запрещено ОС;

в) только с помощью панели управления (команда **Пуск | Настройка | Панель управления | Экран**);

г) у рабочего стола нет свойств.

22. Что такое разрешение экрана?

а) величина, определяющая количество пикселов и, соответственно, размеры экрана;

б) количество цветов, отображаемых на рабочем столе Windows;

в) оформление рабочего стола и его размеры;

г) количество окон, которые одновременно могут быть открыты.

23. Как отформатировать дискету?

а) **Мой компьютер | Диск 3,5** | команда **Форматировать**;

б) в папке Мой компьютер щелкнуть правой кнопкой мыши на значке **Диск 3,5**, выбрать из контексного меню команду **Форматировать** или в папке Мой компьютер выполнить команду **Файл | Форматировать**;

в) только с помощью команды меню **Пуск | Выполнить** `format a:`;

г) выйти в режим эмуляции DOS и выполнить команду `format a:`.

24. Как скопировать документ на дискету наиболее коротким способом?

а) правая кнопка мыши на документе, в контекстно-зависимом меню выбрать команду **Отправить | Диск 3,5**;

б) выделить документ, выполнить команду **Правка | Копировать**, выбрать место вставки (**Диск 3,5 | Вставить**);

в) с помощью кнопки **Копировать** на панели инструментов;

г) перетащить выделенный документ в предварительно открытое окно с диском 3,5.

25. Как очистить Корзину за наименьшее число шагов?

а) открыть Корзину, выбрать команду **Файл | Очистить Корзину**;

б) правая кнопка мыши — в контекстно-зависимом меню выбрать команду **Очистить Корзину**, подтвердить удаление объектов;

в) любым из перечисленных способов;

г) перетащить Корзину на панель задач.

26. Как восстановить объект, отправленный в Корзину из окна работы с файлами?

    а) открыть Корзину, выделить объект и вытащить его из Корзины на рабочий стол;

    б) открыть Корзину, выделить объект, выбрать команду **Файл | Восстановить**;

    в) щелкнуть правой кнопкой мыши на Корзине, выбрать команду **Проводник | Восстановить все**;

    г) перетащить Корзину на панель задач.

27. Как выделить несколько значков, идущих не подряд?

    а) с помощью сочетания <Ctrl>+<A>;

    б) выбрать команду **Правка | Выделить все**;

    в) удерживая <Shift>, щелкнуть по нужным объектам;

    г) выделить первый, нажать <Ctrl>, щелкнуть мышью на следующем и т. д.

28. Как создать документ, папку?

    а) только с помощью команды контекстно-зависимого меню **Создать**, затем выбрать нужный объект;

    б) только с помощью команды меню **Файл | Создать | Выбрать тип объекта**;

    в) с помощью команды меню **Файл | Создать** или контекстно-зависимого меню;

    г) нарисовать новую папку, используя инструмент **Карандаш**.

29. Как найти документ (папку, компьютер)?

    а) выбрать команду **Пуск | Найти**, ввести критерии поиска;

    б) из папки Мой компьютер перейти в нужную папку (диск) и выделить нужный объект;

    в) функции поиска в Windows не предусмотрены;

    г) через сетевое окружение.

30. Как выполнить проверку диска в Windows?

    а) в папке Мой компьютер выбрать нужный диск, выполнить команду контекстно-зависимого меню **Свойства | Сервис** или выполнить команду **Пуск | Программы | Стандартные | Служебные | Проверка**;

    б) в папке Мой компьютер выбрать нужный диск, выполнить команду контекстно-зависимого меню **Свойства | Сервис | Очистка диска**;

    в) функции проверки доступны только администратору сети;

    г) выйти в режим эмуляции DOS и выполнить команду `format c:`.

31. Можно ли проверить качество дискеты при выполнении операции форматирования?

    а) да, по количеству поврежденных секторов;

    б) да, по количеству свободного места на дискете;

    в) нет, при операции форматирования данные недоступны;

    г) да, по наличию метки на диске.

32. Как вызвать панель инструментов, если ее нет в окне?

    а) выполнить команду меню **Вид | Панель инструментов** или щелкнуть правой кнопкой мыши на строке меню;

    б) выполнить команду меню **Вид | Обновить**;

    в) выполнить команду меню **Вид | Крупные значки**;

    г) только выполнить команду меню **Вид | Панель инструментов**.

33. Что находится в заголовке любого окна?

    а) имя программы и ее значок;

    б) имя файла и значок программы;

    в) значок программы, имя документа, имя программы, кнопки управления окном;

    г) слово "заголовок".

34. Доступ к каким типам сетей может быть организован на базе ОС Windows?

    а) к любой местной внешней электрической сети;

    б) к внутренней и наружной телефонной сети;

    в) к локальной и глобальной компьютерной сети;

    г) только к Глобальной информационной сети Интернет.

35. Какие возможности доступны с помощью правой кнопки мыши (ПКМ)?

    а) ПКМ не добавляет новых возможностей при работе в Windows;

    б) вызов контекстно-зависимого меню;

    в) изменение свойств у некоторых объектов;

    г) выключение компьютера.

36. Что такое ярлык?

    а) ссылка на удаленный ресурс;

    б) значок со стрелкой, позволяющий изменять свойства объекта;

    в) значок со стрелкой, осуществляющий переход к ресурсам ПК и служащий для запуска программ;

    г) стандартный элемент ОС Windows.

37. Для создания ярлыка необходимо:

    а) воспользоваться пунктом меню **Справка**, далее следовать инструкциям;

    б) перетащить с помощью правой кнопки мыши (ПКМ) значок программы или ресурса на рабочий стол;

    в) скопировать объект с помощью ПКМ;

    г) с помощью ПКМ перетащить значок программы или ресурса, выбрать команду **Создать ярлык**, или использовать команду контекстного меню.

38. Как завершить работу с программой (приложением)?

    а) <Alt>+<F4> или команда **Файл | Выход**;

    б) кнопка **Закрыть** или команда **Файл | Выход**;

    в) <Alt>+<F4>, кнопка **Закрыть** и команда **Файл | Выход**;

    г) <Alt>+<Tab>.

39. Как переименовать документ?

    а) ввести новое имя документа в адресной строке;

    б) использовать команду меню **Файл | Переименовать**;

    в) выделить документ, выполнить команду меню **Переименовать** или выбрать соответствующую команду в контекстно-зависимом меню;

    г) только с помощью контекстно-зависимого меню **Переименовать**.

40. Какое действие выполняет команда **Отправить** из контекстно-зависимого меню?

    а) перемещает выделенный объект на диск 3,5", в электронное письмо или принтер;

    б) создает ярлык к указанному объекту на рабочем столе;

    в) копирует выделенный объект на диск 3,5", отправляет адресату или создает ярлык на рабочем столе;

    г) отправляет документ в Корзину.

2.6. Инсталляция программ

Вопросы (продолжение)

49. Чем отличается простое копирование от инсталляции программы?
50. Что такое дистрибутив?
51. Какой файл необходимо запустить для установки программы?
52. Зачем дистрибутиву серийный номер?
53. Существует ли в нашей стране уголовная ответственность за компьютерные преступления?

Практикум 9. Установка программного обеспечения

1. Установите любую доступную по системным требованиям и дисковому пространству программу на жесткий диск вашего компьютера.
2. Настройте программу для работы, создав ей ярлык на рабочем столе.
3. Проверьте ее работу.
4. Покажите результат преподавателю.
5. Грамотно деинсталлируйте программу.

2.7. Архиваторы

Архиваторы — это программы, позволяющие паковать информацию, находящуюся в виде файлов на магнитных или лазерных дисках, с целью уменьшения их объема и снижения вероятности порчи при пересылке по электронной почте или копировании на другие диски. Также архивация — это один из способов защиты информации.

Вопросы (продолжение)

54. Какие архиваторы вы знаете для DOS?
55. Как можно запаковать файлы для переноса на дискетах, если их суммарный объем даже после архивирования составляет 6 Мбайт?
56. Как можно защитить информацию при помощи архивирования?
57. Какие действия мы можем выполнять с архивами?
58. Что такое SFX-архив?
59. Какие архиваторы сейчас являются наиболее популярными под Windows?

Практикум 10

1. Используя имеющуюся на диске A: после выполнения *Практикума 7 главы 2* папку ZVERI, запакуйте ее при помощи архиватора WinRar под именем Zoo.rar на рабочем столе.
2. А теперь создайте из той же папки ZVERI SFX-архив Zoo.exe с электронной подписью для защиты от несанкционированного доступа (она, правда, доступна только в лицензионной зарегистрированной версии).
3. Сравните объемы обоих архивов.
4. Найдите папку размером примерно 10 Мбайт и сделайте многотомный архив (размер одного тома — 1 457 500 байт).
5. Проделайте то же самое при помощи архиватора WinZip.
6. Сравните полученные результаты работы двух архиваторов.
7. Подготовьте файл-отчет Report.txt о проделанном практикуме и выводах.

8. Сдайте работу преподавателю.
9. Уничтожьте следы вашего пребывания на жестком диске.

Тест 10. Что вы знаете про архиваторы

1. ARJ, ZIP, RAR это:
   а) названия программ-архиваторов (упаковщиков);
   б) расширения обычных файлов;
   в) латинские буквы алфавита;
   г) команды MS-DOS.
2. Что такое архивация данных?
   а) увеличение объема (размера) файла по сравнению с исходным;
   б) уменьшение объема (сжатие) данных и помещение в архив;
   в) временное хранение информации в виде особого файла;
   г) упаковка файла с отправкой его адресату по электронной почте.
3. Что такое разархивация данных?
   а) извлечение (распаковка) файлов из архива;
   б) помещение файлов в архив;
   в) временное хранение информации в виде особого файла;
   г) получение архивированного файла по электронной почте.
4. Где хранятся упакованные файлы?
   а) во временном каталоге;
   б) в специальном файле, имеющем расширение программы архиватора;
   в) место хранения определяет пользователь;
   г) на специальном сервере.
5. Как определить, работая в Windows, что это архивный файл?
   а) по цвету файла;
   б) по значку или по расширению;
   в) по имени файла;
   г) визуально не определяется.
6. Как проверить целостность архива?
   а) только с помощью специальных программ;
   б) распаковать архив или воспользоваться командой архиватора;
   в) только с помощью команды программы архиватора;
   г) целостность архива не проверяется.

7. Какие файлы пакуются плотнее всего:
   а) графические в формате JPG;
   б) музыкальные в формате MP3;
   в) текстовые в формате TXT;
   г) базы данных в формате MDB.

Тест 11. Простые правила по защите информации

Этот тест, в отличие от предыдущих, очень короток. Но однозначных ответов на поставленные вопросы, наверное, нет. Это тест, скорее для разговора, может быть, даже спора о методах защиты и сохранения информации.

1. Для возможности восстановления информации после сбоев нужно:
   а) проверять компьютер на наличие вирусов;
   б) архивировать данные (файлы) на жесткий диск;
   в) делать резервные копии файлов на внешние носители;
   г) иметь под рукой загрузочную дискету.
2. Для предотвращения потерь информации необходимо:
   а) проверять входящие дискеты и компакт-диски на наличие вирусов;
   б) проводить проверку жесткого диска на наличие ошибок;
   в) использовать антивирусные программы и обслуживать компьютер;
   г) использовать лицензионное программное обеспечение.
3. Как ограничить доступ к информации (файлам) посторонних лиц?
   а) установить пароль на загрузку компьютера;
   б) установить сигнализацию или пропускной режим;
   в) установить для файлов атрибут "только чтение";
   г) установить видеокамеру для наблюдения за помещением.

2.8. Антивирусные программы

О вирусах можно говорить долго, но, т. к. они множатся каждый день (это не преувеличение!), то пока эта книга увидит свет, многое устареет.

Краткая классификация

Вирусы можно разделить на классы по следующим основным признакам:

- среда обитания;
- операционная система (ОС);

- особенности алгоритма работы;
- деструктивные возможности.

По *среде обитания* вирусы можно разделить на:

- файловые;
- загрузочные;
- макро;
- сетевые.

Файловые вирусы либо различными способами внедряются в выполняемые файлы (наиболее распространенный тип вирусов), либо создают файлы-двойники (компаньон-вирусы), либо используют особенности организации файловой системы (link-вирусы).

Загрузочные вирусы ориентированы на конкретные форматы расположения системных данных в загрузочных секторах дисков и записывают себя либо в загрузочный сектор диска (boot-сектор), либо в сектор, содержащий системный загрузчик винчестера (Master Boot Record), либо меняют указатель на активный boot-сектор.

Макро-вирусы заражают файлы-документы и электронные таблицы нескольких популярных редакторов (форматов Word, Excel, других программ пакета MS Office).

Сетевые вирусы используют для своего распространения протоколы или команды компьютерных сетей и электронной почты.

Существует большое количество сочетаний: например, файлово-загрузочные вирусы, заражающие как файлы, так и загрузочные секторы дисков. Такие вирусы, как правило, имеют довольно сложный алгоритм работы, часто применяют оригинальные методы проникновения в систему, используют технологии стелс и полиморфик.

Другой пример такого сочетания — сетевой макро-вирус, который не только заражает редактируемые документы, но и рассылает свои копии по электронной почте. Заражаемая операционная система (вернее ОС, объекты которой подвержены заражению) является вторым уровнем деления вирусов на классы. Каждый файловый или сетевой вирус заражает файлы какой-либо одной или нескольких ОС: DOS, Windows, OS/2 и т. д.

Среди особенностей алгоритма работы вирусов выделяются следующие пункты:

- резидентность;
- использование стелс-алгоритмов;
- самошифрование и полиморфичность;
- использование нестандартных приемов.

Резидентный вирус при инфицировании компьютера оставляет в оперативной памяти свою резидентную часть, которая затем перехватывает обращения операционной системы к объектам заражения и внедряется в них. Резидентные вирусы находятся в памяти и являются активными вплоть до выключения компьютера или перезагрузки операционной системы.

Нерезидентные вирусы не заражают память компьютера и сохраняют активность ограниченное время.

Некоторые вирусы оставляют в оперативной памяти небольшие резидентные программы, которые не распространяют вирус. Такие вирусы считаются нерезидентными.

Резидентными можно считать макро-вирусы, поскольку они постоянно присутствуют в памяти компьютера на все время работы зараженного редактора. При этом роль операционной системы берет на себя редактор, а понятие "перезагрузка операционной системы" трактуется как выход из редактора.

В многозадачных операционных системах время "жизни" резидентного DOS-вируса также может быть ограничено моментом закрытия зараженного DOS-окна, а активность загрузочных вирусов в некоторых операционных системах ограничивается моментом инсталляции дисковых драйверов ОС.

Использование *стелс-алгоритмов* позволяет вирусам полностью или частично скрыть себя в системе. Наиболее распространенным стелс-алгоритмом является перехват запросов ОС на чтение/запись зараженных объектов. Стелс-вирусы при этом либо временно лечат их, либо "подставляют" вместо себя незараженные участки информации.

В случае *макро-вирусов* наиболее популярный способ — запрет вызовов меню просмотра макросов. Один из первых файловых стелс-вирусов — вирус "Frodo", первый загрузочный стелс-вирус — "Brain".

Самошифрование и *полиморфичность* используются практически всеми типами вирусов для того, чтобы максимально усложнить процедуру детектирования вируса. Полиморфик-вирусы (polymorphic) — это достаточно труднообнаружимые вирусы, не имеющие сигнатур, т. е. не содержащие ни одного постоянного участка кода. В большинстве случаев два образца одного и того же полиморфик-вируса не будут иметь ни одного совпадения. Это достигается шифрованием основного тела вируса и модификациями программы-расшифровщика.

Различные *нестандартные приемы* часто используются в вирусах для того, чтобы как можно глубже спрятать себя в ядре ОС (как это делает вирус "3APA3A"), защитить от обнаружения свою резидентную копию (вирусы "TPVO", "Trout2"), затруднить лечение от вируса (например, поместив свою копию в Flash-BIOS) и т. д.

По *деструктивным возможностям* вирусы можно разделить на:

- *безвредные*, т. е. никак не влияющие на работу компьютера (кроме уменьшения свободной памяти на диске в результате своего распространения);

- *неопасные*, влияние которых ограничивается уменьшением свободной памяти на диске и графическими, звуковыми и пр. эффектами;
- *опасные* вирусы, которые могут привести к серьезным сбоям в работе компьютера;
- *очень опасные*, в алгоритм работы которых заведомо заложены процедуры, которые могут привести к потере программ, уничтожить данные, стереть необходимую для работы компьютера информацию, записанную в системных областях памяти, и даже, как гласит одна из непроверенных компьютерных легенд, способствовать быстрому износу движущихся частей механизмов — вводить в резонанс и разрушать головки некоторых типов винчестеров.

Из антивирусных программ наиболее известны Norton Antivirus, DrWeb, AVP Касперского, Adinf, и т. д.

Говорить о преимуществах того или иного антивируса мы не будем. Это хорошее исследовательское и никогда не стареющее задание учащимся, среди которых всегда находятся юные хакеры. Почему-то тема компьютерных вирусов маняща для молодых дарований.

Вопросы (окончание)

60. Какими могут быть признаки заражения компьютерными вирусами?
61. Каковы должны быть действия пользователя при угрозе или уже после заражения вирусом?
62. Какие превентивные действия необходимо предпринимать во избежание заражения вирусами?
63. Как можно наиболее легко заразиться компьютерным вирусом?
64. К каким последствиям может привести заражение компьютерными вирусами?
65. Какие типы компьютерных вирусов вы знаете?
66. Чему больше угрожают вирусы: Hardware или Software?
67. Может ли пустая дискета стать источником заражения компьютера?

Практикум 11. Антивирусные программы

1. Узнать, какая антивирусная программа стоит на вашем компьютере.
2. Определить ее вирусную базу.
3. Сканировать дискету.
4. Получить отчет.

Практикум 12. Сравнительный анализ антивирусных программ

Провести маркетинговое исследование современных антивирусных программ по следующим критериям:

- вирусная база (количество распознаваемых вирусов);
- частота обновления вирусных баз;
- стоимость лицензионного продукта;
- скорость сканирования;
- возможность лечения оперативной памяти;
- успешность борьбы с "троянами" и "червями".

Написать отчет о сравнительном анализе современных антивирусных программ.

Тест 12. Антивирусные программы

1. Заражение компьютерными вирусами может произойти в процессе:
   а) работы с файлами;
   б) форматирования дискеты;
   в) выключения компьютера;
   г) печати на принтере.
2. Что необходимо иметь для проверки на вирус жесткого диска?
   а) защищенную программу;
   б) загрузочную программу;
   в) файл с антивирусной программой;
   г) дискету с антивирусной программой, защищенную от записи.
3. Какая программа не является антивирусной?
   а) AVP;
   б) Defrag;
   в) Norton Antivirus;
   г) DrWeb.
4. Какие программы не относятся к антивирусным?
   а) программы-фаги;
   б) программы сканирования;
   в) программы-ревизоры;
   г) программы-детекторы.

5. Как вирус не может появиться в компьютере?

   а) переместиться с гибкого диска;

   б) при просмотре сайтов Интернета;

   в) при отключении компьютера от сети;

   г) при работе с электронной почтой.

6. Как происходит заражение "почтовым" вирусом?

   а) при открытии зараженного файла, присланного с письмом по E-mail;

   б) при подключении к почтовому серверу;

   в) при подключении к Web-серверу, зараженному "почтовым" вирусом;

   г) при получении с письмом, присланном по E-mail, зараженного файла.

7. Как обнаруживает вирус программа-ревизор?

   а) контролирует важные функции компьютера и пути возможного заражения;

   б) отслеживает изменения загрузочных секторов дисков;

   в) при открытии файла подсчитывает контрольные суммы и сравнивает их с данными, хранящимися в базе данных;

   г) периодически проверяет все имеющиеся на дисках файлы.

8. Компьютерным вирусом является:

   а) программа проверки и лечения дисков;

   б) любая программа, созданная на языках низкого уровня;

   в) программа, скопированная с плохо отформатированной дискеты;

   г) специальная программа небольшого размера, которая может приписывать себя к другим программам и обладает способностью "размножаться".

9. Заражению компьютерными вирусами могут подвергнуться:

   а) графические файлы;

   б) программы и документы;

   в) звуковые файлы;

   г) видеофайлы.

10. Какие из перечисленных типов не относятся к категории вирусов?

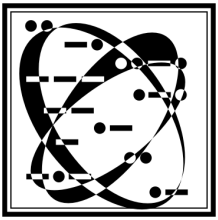
    а) загрузочные вирусы;

    б) type-вирусы;

    в) сетевые вирусы;

    г) файловые вирусы.

Глава 3



Компьютерное моделирование

3.1. Краткие сведения о моделях

Рассказывать, что такое *модель* и *моделирование*, какие модели бывают и как их на компьютере реализовать — дело весьма непростое. В одном только Большом энциклопедическом словаре для слова "модель" приведено девять значений.

Поэтому, как нам представляется, задача учителя в теоретической части — все вышесказанное учащимся прояснить, а здесь мы дадим ряд самостоятельных заданий, выполнение которых, на наш взгляд, очень тесно смыкается с информационными технологиями, а именно с электронными таблицами и базами данных. Да и вообще, компьютер позволяет моделировать практически все школьные предметные области. И это — огромное поле деятельности для учителей и учащихся.

Что же такое модель? Приведем несколько определений модели.

- Отражение наиболее существенных признаков, свойств и отношений явлений, объектов или процессов.
- Объект, используемый вместо другого объекта с целью его изучения или экспериментов над ним.
- Уменьшенная или увеличенная копия какого-либо материального объекта, имеющая с ним обязательное внешнее сходство.
- Описание какого-либо реального объекта или процесса на языке программирования высокого уровня или с использованием новейших компьютерных технологий.

Основные технологические цепочки решения задачи моделирования с использованием компьютера:

- построение модели — постановка задачи — разработка и исполнение алгоритма — анализ результатов;
- постановка задачи — построение модели — разработка и исполнение алгоритма — анализ результатов;

- постановка задачи — разработка и исполнение алгоритма — построение модели — анализ результатов;
- разработка и исполнение алгоритма — постановка задачи — построение модели — анализ результатов.

Классификация моделей:

- Модели с учетом фактора времени:
  - *статические* (фактор времени не учитывающие), например рост и вес учеников вашего класса на данный момент;
  - *динамические* (учитывающие фактор времени), например анализ изменений этих данных за 10 лет обучения в школе.
- Модели по области применения:
  - *научно-производственные* — для исследования явлений и процессов. Пример: термоядерная установка ТОКАМАК для исследования управляемой термоядерной реакции;
  - *игровые* — моделирование поведения объектов (в том числе и людей) в различных близких к жизненным ситуациях. У детей, например игра в "дочки-матери" или в "магазин". У взрослых — военные учения;
  - *имитационные* — не просто отражают основные свойства объекта или реальности, но и имитируют их. Например, сейчас вводится Единый Государственный Экзамен (ЕГЭ), но не сразу по всей стране, а в нескольких регионах, чтобы оценить все его плюсы и минусы;
  - *учебные* — всевозможные наглядные пособия и тренажеры для обучения. Например тренажер для вождения автомобиля или скелет в кабинете биологии;
  - *экспериментальные* — разрабатываются с целью прогнозирования поведения объекта в реальных условиях. Например модель "Титаника", видимо, плохо испытывали, что и обернулось трагедией.
- По способу представления модели делятся на:
  - *материальные* — всегда имеют реальное воплощение, которое можно пощупать — будь то модель самолета или игрушечный автомобиль;
  - *информационные* — оперируют только со сведениями, информацией об исследуемом объекте.

И вот как раз информационные модели мы и будем пытаться реализовать с помощью компьютера. Сделать это можно как на языках программирования высокого уровня, так и в электронных таблицах Excel и базах данных Access.

3.2. Задачи

Задача 1. Прямоугольный треугольник

Дан прямоугольный треугольник (рис. 3.1), у которого длины катетов могут меняться, но их сумма всегда есть число постоянное, равное 30 см. Минимальная длина катета 0,5 см, максимальная — 29,5 см, изменяться длины могут дискретно на 0,5 см. То есть пары возможны такие:

- A = 0,5 см, B = 29,5 см;
- A = 1 см, B = 29 см;
- A = 1,5 см, B = 28,5 см

и т. д.

Найти экспериментальным путем длины катетов, при которых площадь получаемого прямоугольного треугольника максимальна.

Рис. 3.1. Кто больше

Задача 2. Обои

Имеется прямоугольная комната 6×3 м = 18 м^2. Высота потолков 3 м 20 см. Площадь дверного проема 2 м^2. В комнате два квадратных окна 1,5×1,5 м. Комнату надо оклеить обоями. Имеются три варианта обоев. Первый — рулоны по 18 м шириной 0,45 м. Второй — рулоны по 12 м шириной по 0,5 м. Третий — рулоны по 9 м шириной по 0,65 м. Цена первых — 80 руб. за рулон, вторых — 60 руб. за рулон, третьих — 50 руб. за рулон. У первых на подгон рисунка и обрезки уйдет 20% обоев, у вторых — 15%, у третьих — 12%. Рассчитать количество и стоимость обоев по каждому варианту. Выбрать наиболее выгодный, если учесть, что все три варианта хозяйку устраивают.

Обобщить задачу для любой комнаты с вводимыми исходными данными: h — высота, a — ширина, b — длина, S_1 — площадь окон, S_2 — площадь двери.

Задача 3. Расстановка мебели

Средствами графического редактора построить план комнаты из Задачи 2, учитывая, что окна находятся на длинной и короткой сторонах комнаты, а дверь — в углу короткой без окна (рис. 3.2).

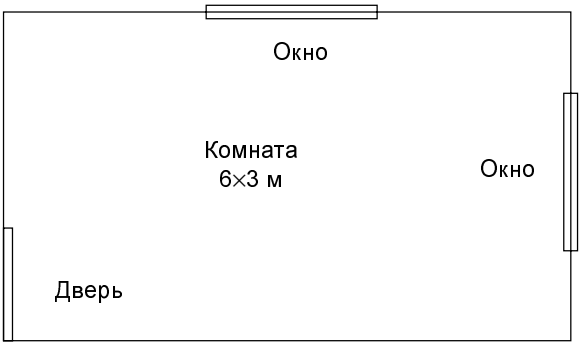


Рис. 3.2. План комнаты

Необходимо расставить (предварительно заготовив шаблоны) в комнате следующую мебель: стол круглый радиусом 0,7 м, шкаф 1,5×0,6 м, диван 1,6×0,8 м, компьютерный стол 1×0,8 м и два стула 0,5×0,5 м.

Найти оптимальную расстановку мебели, которая не загораживала бы окна, была бы эргономична и стильна. Если останется место, то можно в комнату добавить еще что-нибудь. А если есть талант, то можно сделать эскиз объемного вида со стороны двери.

Задача 4. Неотвратимость наказания

Следующая задача — у автора одна из самых любимых. На заре его преподавательской деятельности идея задачи была почерпнута у А. Г. Гейна. Решение ее требует некоего предварительного отступления.

Задача решается на основе многократных наблюдений за одним и тем же объектом. А наблюдать можно за чем угодно. Например Некто, педант и очень аккуратный человек, каждый день на завтрак пьет чашку кофе одного и того же объема (рис. 3.3) и ест бутерброд с сыром одной и той же величины.

Если в течение нескольких дней мы из любопытства понаблюдаем за этим человеком и подсчитаем количество затраченного им на завтрак времени в секундах, то наверняка у нас получатся близкие друг другу, но все же разные числа. Ясно, что на продолжительность вкушения завтрака влияют различные внешние факторы. К примеру, в одно утро кофе был излишне горячий, и завтрак затянулся. А на следующий день Некто опаздывал на работу,

и кофе был недопит. Можно сказать, что продолжительность завтрака — величина случайная. Проведя двадцать наблюдений, мы получим двадцать значений случайной величины. Пусть у нас образовалась следующая последовательность чисел (продолжительность завтрака в секундах):

672, 676, 674, 675, 673, 664, 680, 674, 677, 675, 676, 673, 675, 674, 673, 671, 675, 673, 674, 676.

Их удобнее всего расположить в виде линейной таблицы. Вычислим среднее арифметическое. Оно будет равно 674.

Рис. 3.3. Кофе подан

Чтобы в дальнейшем из многих показаний поглотителей такого же завтрака определить именно нашего Некто, математики ввели специальную величину и назвали ее *дисперсией* (от *dispersion*, разброс). Обозначим значения случайной величины d_1, ..., d_n, а среднее арифметическое этих значений — буквой S. Дисперсия — это среднее арифметическое квадратов разностей между значениями случайной величины и ее средним значением.

$$D = \frac{(d_1 - S)^2 + \ldots + (d_n - S)^2}{n}.$$

Из формулы видно, что чем меньше дисперсия, тем меньше отличаются результаты наблюдений от своего среднего значения и тем ближе среднее значение к истинному. В частности, если $D=0$, то все числа d совпадают между собой (и со своим средним значением).

Допустим теперь, что Некто сагитировал нескольких своих коллег провести тот же эксперимент. Через 20 дней каждый из них, в том числе и Некто, представили по 20 результатов наблюдений, не указав своих фамилий. Можно ли узнать, какие из результатов принадлежат Некто, а какие — нет? Да, можно!

Математики установили, что для этого, как правило, достаточно сравнить дисперсии и средние значения. Дисперсия и среднее значение почти так же индивидуальны, как отпечатки пальцев. Если наблюдения делались над од-

ним и тем же человеком, то дисперсии и средние значения будут близки, если над разными людьми — то далеки.

Осталось выяснить: какие значения считать близкими, а какие далекими? На этот вопрос дает ответ специальный раздел математики — статистика. Оказывается, достоверность ответа зависит от числа наблюдений. Если число наблюдений от 20 до 50, то дисперсии можно считать далекими, когда отношение большей дисперсии к меньшей больше 2. Чтобы говорить о близости средних значений двух последовательностей результатов, надо найти модуль разности средних и разделить его на квадратный корень из суммы дисперсий. Если полученное число больше 0,6, то средние значения считают далекими. В том случае, когда близки и дисперсии, и средние значения, можно сделать вывод, что наблюдения почти наверняка проводились над одним и тем же человеком.

У любознательного ученика может возникнуть вопрос: откуда эти числа (2 и 0,6) взялись? Отвечаем: из специальных таблиц, которые были составлены математиками. Их можно найти в любом справочнике по математической статистике.

Метод сравнения средних значений и дисперсий используется в самых разных областях человеческой деятельности. В медицине — для установления диагноза, в литературоведении — для определения автора произведений, в криминалистике — для розыска преступников.

При решении этой задачи вы как раз и займетесь расследованием одного преступления. Итак.

Органами ОБЭП задержан трейлер с арбузами, похищенными на овощной базе. В городе всего четыре овощных базы, каждая из них получает арбузы из своего сельскохозяйственного района. Определите, с какой базы были похищены арбузы. Расследование осложняется тем, что арбузы на всех базах одного сорта.

Воспользуемся методом средних значений и дисперсий. В каждом сельскохозяйственном районе свои условия произрастания арбузов, поэтому арбузы разных районов отличаются, скажем, удельным весом. Выберем по 30 арбузов (реально, конечно, их выбирается гораздо больше) на каждой овощной базе и из грузовика. У нас получится четыре последовательности по 30 значений в каждой (для каждой овощной базы) и еще одна (для грузовика), с которой мы и будем сравнивать первые четыре. Это наши исходные данные. Результатом явится номер овощной базы, где совершено хищение.

Чтобы добиться этого результата, нужно, как рассказано выше, вычислить средние значения и дисперсии всех пяти последовательностей и провести сравнение. Напомним, что сначала сравниваются дисперсии, а затем для последовательностей, имеющих близкие дисперсии, сравниваются средние

значения. Уже из сказанного ясно, что алгоритм будет достаточно сложным, так что мы построим его методом последовательной детализации.

Прежде всего организуем исходные данные. Последовательности для овощных баз и грузовика удобно записать в виде прямоугольной таблицы, имеющей 5 строк и 30 столбцов. Обозначим ее буквой *A* (последняя строка таблицы *A* соответствует грузовику). Тогда самый общий алгоритм выглядит так:

1. Организовать таблицу с исходными данными.
2. Определить среднее значение для каждой из 5 последовательностей и записать их в таблицу *C* (табл. 3.1).

Таблица 3.1. *Средние значения (C)*

	1-я база	**2-я база**	**3-я база**	**4-я база**	**Грузовик**
Среднее значение удельного веса арбузов					

3. Определить дисперсию для каждой из 5 последовательностей и записать их в таблицу *D* (табл. 3.2).

Таблица 3.2. *Дисперсии (D)*

	1-я база	**2-я база**	**3-я база**	**4-я база**	**Грузовик**
Дисперсия					

4. Для каждой из первых 4 дисперсий определить, близка ли она к пятой, и, если да, близки ли соответствующие средние значения.
5. Если близки и дисперсии, и средние значения, то сообщить номер овощной базы.

Задача 5. Бородино

Как известно, сражение при Бородино (рис. 3.4) пользуется огромным интересом у многих военно-исторических и реконструкторских клубов, которые каждый год, в годовщину сражения собираются на Бородинском поле и вновь разыгрывают это сражение. Это, конечно же, тоже модель, скорее, игровая. А вы попытайтесь, вспомнив историю, в форме слайд-шоу смоделировать ход знаменитого сражения в виде серии последовательных планов. Это, кстати, могло бы быть и мультимедийным проектом.

Рис. 3.4. Скажи-ка, дядя, ведь недаром...

Задача 6. Казино

Неизвестно, часто ли вы бываете в казино и много ли уже выиграли или хоть знаете ли приблизительно правила игры в рулетку (рис. 3.5), ту самую, в которую проигрывал свои денежки и рукописи сам Федор Михайлович Достоевский. Азартный был человек!

Рис. 3.5. Рулетка

Постановка задачи.

Имеется рулетка из 37 цифр, включая zero. После приема ставок выпадает случайное число от 0 до 36 включительно.

У нас будет упрощенный вариант. Изначально у трех игроков по 10 жетонов, каждый стоимостью по 100 рублей. 100 рублей — минимальная ставка.

Ставки принимаются на:

- красное-черное — 1:2 (ставишь 100 рублей, выигрываешь 200);
- чет-нечет — 1:2;
- первая половина (числа 1—18) — 1:2;
- вторая половина (числа 19—36) — 1:2;
- первая треть (числа 1—12) — 1:3 (ставишь 100 рублей, выигрываешь 300);
- вторая треть (числа 13—24) — 1:3;
- первая треть (числа 25—36) — 1:3.
- zero — все достается поставившему.

Модель "Казино" должна принимать ставки, сообщать номер и цвет выпавшего случайным образом шарика, выдавать выигрыши, забирать проигрыши и информировать игроков, сколько у них на счету осталось денег. Игра прекращается либо когда все игроки разорятся, либо когда один из игроков выиграет тройную начальную сумму — то есть 3000 рублей.

Далее следуют несколько логических моделей.

Задача 7. Большая семья

В одной семье было много детей и один компьютер на всех. Семеро из них любили стрелялки, шестеро — стратегии, пятеро — квесты. Четверо любили стрелялки и стратегии, трое — стрелялки и квесты, двое — стратегии и квесты. А еще один любил и то, и другое, и третье.

Сколько детей было в семье?

Задача 8. Три учителя

В одной школе уроки по биологии, географии, английскому языку, французскому языку, истории и математике вели три учителя — Морозов, Васильев и Токарев. Каждый из них преподавал два предмета.

- Учитель географии и учитель французского языка — соседи по дому.
- Морозов — самый младший из троих.

- Все трое — Токарев, учитель биологии и учитель французского языка — ездят из школы вместе.
- Учитель биологии старше учителя математики.
- В свободное время, если им удается найти четвертого партнера, учитель английского языка, учитель математики и Морозов обычно играют в NFS по сети.

Кто какие предметы преподает?

Задача 9. Встреча на конгрессе

На одном международном конгрессе встретились 4 делегата из разных стран. Каждый из них владел двумя языками из четырех (английский, французский, итальянский, немецкий). Однако оказалось, что не было такого языка, на котором они могли бы разговаривать вчетвером. И был только один язык, на котором могли вести беседу трое из них.

- Никто из делегатов не владеет французским и немецким языками одновременно.
- Хотя физик не говорит по-английски, он может служить переводчиком, если математик и биолог захотят поговорить друг с другом.
- Биолог говорит по-немецки и может говорить с химиком, хотя тот не знает ни одного немецкого слова.
- Физик, математик и химик не могут беседовать втроем на одном языке.

Какими двумя языками владеет каждый из них?

Задача 10. Кто с кем знаком?

В одном городе живут 5 человек. Их имена Леонид, Михаил, Николай, Олег и Петр. Их фамилии Атаров, Бартенев, Кленов, Данилин и Иванов.

- Бартенев знаком только с двумя из перечисленных мужчин.
- Петр знаком со всеми, кроме одного.
- Леонид знает только одного из всех.
- Данилин и Михаил незнакомы.
- Николай и Иванов знают друг друга.
- Михаил, Николай и Олег знакомы между собой.
- Атаров незнаком только с одним из всех. Только один из всех знаком с Кленовым.

Назовите имена и фамилии каждого. С кем знаком каждый из них?

Задача 11. Не так-то легко

Шестеро человек — назовем их А, Б, В, Г, Д и Е — кандидаты на посты министра обороны, заместителя министра обороны и секретаря министерства обороны только что получившего независимость государства Балтия. Но определить состав этой тройки оказалось не так-то легко. Судите сами:

- А не хочет входить в состав правительства, если Д не будет председателем.
- Б не хочет входить в состав правительства, если ему придется быть старшим над В.
- Б не хочет работать вместе с Е ни при каких условиях.
- В не хочет работать, если в состав правительства войдут Д и Е вместе.
- В не будет работать, если Е будет министром или если Б будет секретарем.
- Г не будет работать с В или Д, если ему придется подчиняться тому или другому.
- Д не хочет быть заместителем министра.
- Д не хочет быть секретарем, если в состав правительства войдет Г.
- Д не хочет работать вместе с А, если Е не войдет в состав правительства.
- Е согласен работать только в том случае, если министром будет либо он, либо В.

В конце концов были удовлетворены все пожелания капризных кандидатов.

Решите эту проблему и вы.

Задача 12. Едем на экскурсию

Используя свои навыки работы с электронными таблицами, выполните следующее задание.

На экскурсию едут 11 классов в четырех автобусах (табл. 3.3). Количество учащихся в классах дано. Класс может ехать в каком-либо автобусе только целиком. Если мы в ячейке, соответствующей автобусу, поставим 1, то это будет означать, что класс едет целиком в том автобусе, где стоит 1. Напротив одного класса может быть только одна единица, остальные должны быть нули.

Таблица 3.3. *Распределение классов по автобусам*

	A	B	C	D	E	F
1	Класс	Едут на экскурсию	Первый автобус	Второй автобус	Третий автобус	Четвертый автобус
2	6а	23	1			

Таблица 3.3 *(окончание)*

	A	B	C	D	E	F
3	6б	17		1		
4	6в	14				
5	6г	7	1			
6	7а	18				1
7	7б	5				
8	7в	19				
9	7г	22				
10	8а	21				1
11	8б	11				
12	8в	15				
13		Итого в автобусе:	30	17	0	39

Сейчас единички в таблице стоят просто для примера.

Перед тем, как начинать "рассаживание" классов по автобусам, необходимо заполнить формулами нижний этаж ячеек **C13**, **D13**, **E13**, **F13**. Так, в ячейке **C13** должна стоять формула (а не число 30):

`=C2*B2+C3*B3+C4*B4+C5*B5+C6*B6+C7*B7+C8*B8+C9*B9+C10*B10+C11*B11+C12*B12`.

Если все правильно сделать с адресацией ячеек, то формулу можно будет скопировать. Эти формулы позволят в дальнейшем контролировать загрузку автобусов.

Расставляя единички, надо добиться того, чтобы в каждом автобусе ехало не более 45 человек.

Задача 13. Распределение грузов

Используя свои навыки работы с электронными таблицами, выполните следующее задание.

Заполните табл. 3.4 как показано ниже.

Таблица 3.4. *Загрузка грузовиков*

	A	B	C	D	E	F
1	Наименование оборудование	Количество	Вес единицы груза	Первый грузовик	Второй грузовик	Третий грузовик
2	Станки (штуки)	11	850			
3	Трубы (упаковка)	4	1930			
4	Буровое оборудование (ящики)	2	1700			
5	Отделочный камень (ящики)	4	1250			
6	Промышленные электромоторы (штуки)	7	730			
7	Кабель (бухты)	5	1100			
8			Итого вес груза:	0	0	0

В столбцах **D**, **E** и **F** должны находиться числа, определяющие, сколько единиц данного груза берет соответствующий грузовик.

Перед тем как начинать распределение груза, необходимо заполнить формулами ячейки **D8**, **E8**, **F8**. Так, в ячейке **D8** должна стоять формула:

`=D2*C2+D3*C3+D4*C4+D5*C5+D6*C6+D7*C7`

Это позволит в дальнейшем контролировать загрузку грузовиков.

Кроме того, необходимо заполнить формулами ячейки с **F2** по **F7**. Предоставим компьютеру автоматически загружать третий грузовик теми товарами, которые не взяли первые два. Тогда, в ячейке **F2** должна стоять формула: `=B2-D2-E2`. Воспользуйтесь возможностью дублирования формулы вниз, как вас научили на уроке теории, чтобы заполнить оставшиеся ячейки с **F3** по **F7**.

Ваша задача — таким образом распределить товары, чтобы загрузка каждой из автомашин не превышала 12 тонн более чем на 100—150 килограммов.

Задача 14. Моделирование физических явлений

1. Построение линий напряженности электростатического поля

Электростатическое поле создается двумя неподвижными зарядами. Величины зарядов вводятся с клавиатуры в процессе диалога. Знак зарядов может быть как одинаковый, так и разный.

Размещение зарядов на экране произвольное, задается с помощью клавиш перемещения курсора или мыши.

2. Построение эквипотенциальных поверхностей электростатического поля, создаваемого пластиной и зарядом

Пластина представляет собой бесконечную плоскость. Величина заряда и потенциал плоскости вводятся с клавиатуры в процессе диалога. Знак зарядов может быть как одинаковый, так и разный. Размещение заряда на экране произвольное, задается с помощью клавиш перемещения курсора или мыши.

3. Измерение массы тела на рычажных весах

Случайным образом компьютер подбирает три тела, массу которых необходимо измерить.

На экране рисуются весы. Учащийся ставит тело на одну чашу весов. На другую чашу учащийся подбирает для уравновешивания гири из стандартного набора, адекватного школьному лабораторному набору.

4. Измерение объема тела

Случайным образом машина подбирает три тела различной формы и объема. На экране рисуется мензурка с налитой водой. Учащийся определяет начальный объем воды, а затем по очереди опускает тела в мензурку и определяет новый получившийся объем.

5. Определение плотности твердого тела

Случайным образом машина подбирает два тела различного объема, массы и формы. Одно тело из известного материала, другое — из неизвестного.

На экране рисуются весы, и учащийся поочередно определяет массу тел, пользуясь стандартным набором гирь, адекватных школьному лабораторному набору.

На экране рисуется мензурка с налитой водой. Учащийся поочередно определяет объем тел. Затем по полученным данным рассчитывается плотность вещества.

Для первого тела результат сверяется с таблицей. Для второго по таблице определяется тип вещества.

Задача 15. Волк кролику не товарищ. Экологическая модель

Волчий остров размером 20×20 заселен дикими кроликами, волками и волчицами. Имеется по несколько представителей каждого вида. Кролики довольно глупы: в каждый момент времени они с одинаковой вероятностью 1/9 перемещаются в один из восьми соседних квадратов (за исключением участков, ограниченных береговой линией) или просто сидят неподвижно. Каждый кролик с вероятностью 0,2 превращается в двух кроликов. Каждая волчица передвигается случайным образом, пока в одном из соседних восьми квадратов не окажется кролик, за которым она охотится. Если волчица и кролик оказываются в одном квадрате, волчица съедает кролика и получает одно очко. В противном случае она теряет 0,1 очка. Волки и волчицы с нулевым количеством очков умирают.

В начальный момент времени все волки и волчицы имеют 1 очко. Волк ведет себя подобно волчице до тех пор, пока в соседних квадратах не исчезнут все кролики. Тогда если волчица находится в одном из восьми близлежащих квадратов, волк гонится за ней. Если волк и волчица окажутся в одном квадрате и там нет кролика, которого нужно съесть, они производят потомство случайного пола.

Запрограммируйте предполагаемую экологическую модель, понаблюдайте за изменением популяции в течение некоторого периода времени.

Задача 16. Перевод температур

Реализовать математическую модель, позволяющую переводить градусы по шкале Цельсия в градусы по шкале Фаренгейта, Реомюра, Кельвина.

Задача 17. Десять городов

Десять городов связаны между собой прямыми шоссейными автомагистралями. Построить математическую модель, которая позволяет определить минимальное и максимальное расстояния между городами, предлагает маршруты по двум заданным городам, находит наикратчайший маршрут.

Задача 18. Страны Африки. Структурная информационная модель

Создать базу данных стран Африки со следующими полями:

- полное название страны;
- столица;
- население;
- площадь;
- государственное устройство;
- стабильность политической ситуации (коэффициент 0—10);
- валюта;
- государственный язык;
- флаг;
- климат.

Реализовать запрос к базе данных, позволяющий сделать выборку, например, по государственному языку (французский), численности населения (не более 20 млн.) и стабильности (выше 7).

Задача 19. Взлет ракеты с поверхности планеты

Это классическая задача динамики движения тела переменной массы, и поскольку ее невозможно в строгой постановке решить аналитически (так как сила тяжести, сила сопротивления атмосферы и масса ракеты меняются с высотой), она представляет живой интерес для компьютерного моделирования. Исходными данными в рассматриваемой задаче является масса и радиус планеты, масса ракеты, масса топлива, скорость сгорания топлива и скорость вылета частиц топлива из сопла ракеты, а также зависимость силы сопротивления атмосферы от высоты ракеты и ее скорости. Для того чтобы данная задача могла быть решена школьниками, их следует на уроках физики 9-го класса ознакомить с уравнением Мещерского, а на уроках информатики — с простейшими методами численного решения задачи Коши для обыкновенных дифференциальных уравнений, например, с явным методом Эйлера.

В рассматриваемой задаче о старте ракеты при построении пользовательского интерфейса на языке программирования Visual Basic придется учитывать возможность повторного задания массы ракеты и характеристик топлива, поскольку при некоторых значениях указанных параметров старт ракеты вообще невозможен. Для обеспечения необходимой для организации самых разнообразных компьютерных экспериментов (например, по выявлению влияния массы и радиуса планеты на условия старта и полета ракеты и пр.) максимальной гибкости в задании исходных данных, ввод их в этом проекте

можно организовать с помощью специальной формы (**VB. Form**), содержащей соответствующие текстовые поля (**VB. TextBox**). Указанный способ позволит скомпоновать вместе все исходные числовые данные рассматриваемой задачи, обеспечит их наглядность и легкость редактирования с помощью стандартных методов языка Visual Basic, связанных с клавиатурой и мышью.

Задача 20. Море, море... Крупнейшие зарубежные пассажирские суда XX века. Структурированная модель

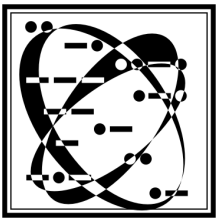
Имеется следующая неструктурированная информация. Необходимо структурировать ее и определить, какому государству принадлежит большинство крупнейших судов мира (рис. 3.6), самый скоростной лайнер и самый вместительный.

Рис. 3.6. "Титаник"

"Дейчланд" построен в 1900 году. "Лузитания" принадлежала компании "Кунард Лайн". "Бисмарк" был водоизмещением 56 551 т. "Титаник" мог принять на борт 2584 пасссажира. У "Олимпика" мощность двигателей составляла 46 000 л. с. "Дейчланд" был водоизмещением 16 502 т. "Нормандия" построена в 1935 году во Франции. "Куин Мэри" могла развить скорость 28 узлов. "Куин Элизабет" принадлежала компании "Кунард Лайн". У "Юнайтед Стейтс" мощность двигателей составляла 235 000 л. с. "Куин Элизабет II"

могла развить скорость 28,5 узлов. "Дейчланд" построен в Германии. "Куин Элизабет II" принадлежала компании "Кунард Лайн". "Бисмарк" принадлежал компании "Гамбург-Америка-Лайн". "Юнайтед Стейтс" мог развить скорость 30 узлов. "Нормандия" могла принять на борт 1972 пассажира. "Титаник" построен в 1912 году в Великобритании. У "Куин Мэри" мощность двигателей составляла 81 235 л. с. "Олимпик" принадлежал компании "Уайт Стар Лайн". "Лузитания" построена в 1907 году. "Куин Элизабет II" могла принять на борт 2025 пассажиров. У "Куин Элизабет" мощность двигателей составляла 160 000 л. с. "Бисмарк" построен в Германии. У "Дейчланд" мощность двигателей составляла 36 000 л. с. "Нормандия" принадлежал компании "Френч Лайн". "Лузитания" построена в Великобритании. "Юнайтед Стейтс" был водоизмещением 53 332 т. "Куин Элизабет II" построена в 1969 году. "Бисмарк" мог принять на борт 2095 пассажиров. У "Нормандии" мощность двигателей составляла 160 000 л. с. "Куин Мэри" построена в 1936 году в Великобритании. "Титаник" принадлежал компании "Уайт Стар Лайн". "Дейчланд" мог развить скорость 23,5 узлов. "Куин Элизабет II" построена в Великобритании. "Юнайтед Стейтс" принадлежал компании "Юнайтед Стейт Лайнс". "Дейчланд" принадлежал компании "Гамбург-Америка-Лайн". "Бисмарк" построен в 1914 году. "Куин Мэри" принадлежал компании "Кунард Лайн". "Юнайтед Стейтс" мог принять на борт 1962 пассажира. "Олимпик" построен в 1911 году в Великобритании. У "Куин Элизабет II" мощность двигателей составляла 110 000 л. с. "Лузитания" была водоизмещением 31 550 т. "Титаник" был водоизмещением 46 300 т. "Куин Элизабет" построена в 1939 году в Великобритании. "Куин Элизабет II" была водоизмещением 65 863 т. "Дейчланд" мог принять на борт 1283 пассажира. У "Бисмарка" мощность двигателей составляла 100 000 л. с. "Нормандия" была водоизмещением 83 423 т. "Юнайтед Стейтс" построен в 1952 году в США. "Лузитания" могла развить скорость 25 узлов. "Нормандия" могла развить скорость 28 узлов. "Куин Элизабет" была водоизмещением 83 673 т. "Бисмарк" мог развить скорость 23 узла. "Куин Элизабет" могла принять на борт 2283 пассажира. "Титаник" мог развить скорость 21,5 узлов. У "Титаника" мощность двигателей составляла 46 000 л. с. "Лузитания" построена в Великобритании. "Куин Мэри" мог принять на борт 2139 пассажиров. "Куин Элизабет" мог развить скорость 28,5 узлов. "Лузитания" могла принять на борт 2200 пассажиров. "Олимпик" мог принять на борт 2584 пассажира. "Олимпик" был водоизмещением 46 359 т. У "Олимпика" мощность двигателей составляла 46 000 л. с. У "Лузитании" мощность двигателей составляла 68 000 л. с. "Олимпик" мог развить скорость 21 узел.

Глава 4



Алгоритмизация и программирование

4.1. Алгоритмы. Их типы и свойства

Все, что бы мы ни делали, будь то за компьютером или в жизни, чаще всего имеет какую-либо цель. И не всегда эта цель достигается. Но если как следует сформулировать желаемый для себя результат, а потом продумать четкий план его достижения, то, по крайней мере, на компьютере цель будет достигнута (в жизни, к сожалению, не все так однозначно).

Так вот, четкая, ясная и однозначная последовательность действий, приводящая к достижению результата, называется *алгоритмом*.

Примеры алгоритмов из жизни. Это и правила перехода улицы, и рецепт пирога, и инструкция по изготовлению ядерной бомбы в домашних условиях и т. д.

Сразу попробуйте подумать и ответить, что из нижеперечисленного является алгоритмом, а что нет:

- правила игры в футбол;
- политическая карта мира;
- телефонный справочник;
- файл Readme, содержащий информацию об инсталляции программы;
- вычисление корней квадратного уравнения;
- приготовление завтрака;
- решение шахматной задачи;
- инструкция по пользованию видеомагнитофоном.

4.1.1. Виды алгоритмов

Алгоритмы бывают трех основных видов, которые и являются базовыми при написании программ.

Первый тип — *линейный* алгоритм; такой, в котором все действия выполняются в строгом порядке, последовательно, одно за другим. Типичный жизненный пример такого алгоритма — рецепт пирога.

Второй тип — *разветвляющийся* алгоритм; такой, в котором выполняются те или иные действия в зависимости от выполнения или невыполнения некоего условия. Пример из жизни — правило перехода улицы по светофору. Если горит красный — стоим, если горит зеленый — идем.

Наконец, третий тип — *циклический* алгоритм; такой, в котором присутствуют повторяющиеся действия с какой-либо изменяющейся величиной, так называемым *параметром*. Пример — колка дров. Берем полено — ставим на попа, колем топором, берем второе полено и т. д., пока поленья не закончатся, и эта работа нам не надоест.

4.1.2. Свойства алгоритмов

Результативность (или конечность) — выполнение алгоритма должно приводить к результату за конечное количество шагов.

Дискретность (или детализация) — алгоритм поддается расчленению на элементарные (дискретные) шаги, которые могут быть исполнены при помощи системы команд исполнителя.

Однозначность — каждый шаг исполнителя может и должен быть истолкован одним и только одним способом.

Понятность — алгоритм должен быть составлен только из команд, входящих в систему команд исполнителя.

Массовость — алгоритм должен решать однотипные задачи с различными исходными данными.

Переносимость (или совместимость) — алгоритм не должен зависеть от типа используемой вычислительной техники или выбранного языка программирования.

Теперь ответим на несколько вопросов:

1. Назовите основные виды алгоритмов и приведите свой пример из жизни для каждого вида.
2. Перечислите и поясните свойства алгоритмов.
3. Придумайте систему команд для исполнителя алгоритма чистки картошки.
4. А теперь для заварки чая.

5. А теперь предложите свой алгоритм действий для достижения следующего результата — не пойти завтра утром в школу.
6. В приведенных ниже последовательностях каждый следующий элемент получен по некоторому строгому алгоритму. Разгадав его, продолжите ряд:
   - а, в, д, ё, з, й;
   - 1, 2, 4, 8, 16, 32;
   - 1, 4, 9, 16, 25;
   - 1, 1, 2, 3, 5, 8, 13, 21, 34;
   - 1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 0, 1, 1, 1, 2, 1;
   - победа, обеда, беда, еда;
   - о, д, т, ч, п, ш, с, в, д, д.

И парочка заданий на нестандартное мышление:

7. 2, 3, 4, 5, 6, 7, 8, 9, 10.
8. 1, 11, 21, 1211, 111221, 312211, 13112221.

Черные ящики

Черным ящиком будем называть объект, устройство и алгоритм действий которого заранее неизвестны. Чтобы понять алгоритм его работы, надо поставить ряд опытов — подать информацию на вход и посмотреть, что получится на выходе в результате ее обработки (рис. 4.1). На основании нескольких опытов можно выдвинуть гипотезу об алгоритме работы черного ящика. Гипотезу необходимо проверить и подтвердить новыми опытами. Чем их будет больше и чем они будут более разнообразны, тем скорее гипотеза подтвердится или опровергнется.

Рис. 4.1. Черный ящик

Допустим, есть Исполнитель, запрограммированный на несколько десятков алгоритмов по работе с целыми числами, русскими и латинскими буквосочетаниями и номерами букв в алфавите.

Сначала выбирается номер алгоритма.

Затем экспериментатор ставит ряд опытов, т. е. дает Исполнителю разнообразную входную информацию, пытаясь выяснить алгоритм его действий.

Исполнитель после постановки опыта выдает либо выходную информацию, либо одно из двух сообщений.

- *Не понимаю* — в случае, если на вход подана информация не того типа (допустим, этот алгоритм обрабатывает числовую информацию, а на вход было подано буквосочетание, и наоборот).
- *Не могу* — в случае, если Исполнитель не может выполнить алгоритм с введенной информацией (например, Исполнитель делит введенное число на 2, а введено число 3 — ведь Исполнитель работает только с целыми положительными числами).

Приведем ряд примеров, где опыты уже были поставлены. Вам остается лишь понять алгоритм происходящего в черном ящике.

Пример 1.

Вход	А	1	3	7	5
Выход	Не могу	2	4	8	6

Ну, тут не то что в 11-ом, а даже и в 10-м классе понимают, что Исполнитель просто прибавляет единицу к входному числу. Хотя... Если продолжить опыты, то легко может оказаться, что гипотеза неверна. Поставим еще ряд опытов. Дадим на вход 4. На выходе — 3! Вот это да! Ставим еще опыты.

Вход	16	23	34	42	57
Выход	15	22	35	41	58

Особо проницательные после этой серии опытов выдвинут дополненную гипотезу: Исполнитель к нечетным числам добавляет единицу, а из четных вычитает. И вот это, скорее всего, верно. Хотя, как говорится, возможны варианты.

Пример 2.

Вход	Пар	Коля	Кркркркрк	Ru	True
Выход	3	4	9	2	4

Тут вроде бы без подвохов. Не шибко умный, но старательный Исполнитель аккуратно считает количество букв во введенных словах.

А теперь порешаем задачи.

Задачи

Замечание

Нумерация задач по всей главе сквозная.

Найти алгоритм работы черного ящика.

Принять во внимание, что в алгоритмах со словами используется русский алфавит без буквы "ё", т. е. из 32 букв.

1.

Вход	23	5	12	21	132	56	9	20	100	24
Выход	47	11	25	43	265	113	19	41	201	49

2.

Вход	23	5	12	21	132	56	9	20	100	24
Выход	45	9	23	41	263	111	17	39	199	47

3.

Вход	Корона	Село	Винт	Молоко	Революция
Выход	2	1	0	3	1

4.

Вход	25	11	7	98	34
Выход	100	44	28	392	136

5.

Вход	Доллар	Зоопарк	Клен	Кооперация	Рубль
Выход	1	1	0	2	0

6.

Вход	Квас	Змея	Мама	Коля	Три
Выход	Лгбт	Инеа	Нбнб	Лпма	Усй

7.

Вход	26	33	21	8	126
Выход	Не могу	11	7	Не могу	42

8.

Вход	10	13	40	100	54
Выход	4	Не могу	19	49	26

9.

Вход	Квас	Гвоздь	Поток	Одеколон	Молоток
Выход	2	3	Не могу	4	Не могу

10.

Вход	Квас	Гвоздь	Поток	Одеколон	Молоток
Выход	1	1	2	4	3

11.

Вход	Краб	Морс	Яблоко	Парта	Нора
Выход	р	С	я	т	Р

12.

Вход	12	327	500	66	1234	41	3355	98	89	809
Выход	1	4	5	0	3	3	2	1	1	9

13.

Вход	12	327	500	66	1234	41	3355	98	89	809
Выход	3	12	5	12	10	5	16	17	17	17

14.

Вход	13	32	12	500	89	16	18	80	64	24
Выход	Не могу	1	3	125	Не могу	1	9	5	1	3

15.

Вход	мама	ил	я	ай	байт
Выход	28	21	32	11	32

16.

Вход	12	327	500	66	1234	41	3355	98	89	809
Выход	6	654	250	33	617	82	6710	49	178	1618

17.

Вход	12	327	500	66	1234	41	3355	98	89	809
Выход	23	438	611	77	2345	42	4466	09	90	910

18.

Вход	12	34	20	9	28	100	200	27	41	60
Выход	0	1	2	0	1	1	2	0	2	0

19.

Вход	Башня	Як	Мама	Ты	Арбуз
Выход	2	33	13	19	1

20.

Вход	12	327	5	66	9	41	33	98	89	7
Выход	14	507	5	102	11	51	41	142	131	7

21.

Вход	23	2	10	3	20	2002	19	35	11	6
Выход	21	0	9	3	18	2001	18	33	9	6

22.

Вход	1	2	34892	43	791
Выход	1	1	5	2	3

23.

Вход	Я	Буря	Краб	Шея	Гвоздь
Выход	1+0	2+2	1+3	2+1	1+5

24.

Вход	У	Жук	Нога	Арка	Шалаш
Выход	У	Куж	Агон	Акра	Шалаш

25.

Вход	1	8	16	2002	1998	100	108	6	7	23
Выход	0	2	1	2	4	2	3	1	0	0

Найти алгоритм решения следующих задач.

26. Перевозчик.

На берегу реки находятся лодка, лодочник, коза, волк и капуста. В лодку помещается лодочник и еще только один из объектов (рис. 4.2). Как перевезти всех в целости и сохранности на другой берег, если известно, что нельзя оставлять наедине (без лодочника) волка с козой или козу с капустой?

Рис. 4.2. Перевозчик

27. Мужья и жены.

Задача, похожая на предыдущую, но более сложная.

Три достойных мужа, каждый со своей красавицей-женой, собрались съездить на пикник на островок, расположенный посередине реки. Есть лодка, которая может вместить только двух человек. Плавать никто из них не умеет, да и течение довольно сильное. Как же могут переправиться на островок шестеро достойных людей при условии, что оказавшись отдельно от своего мужа, ни одна дама не находилась бы при этом в обществе других мужей (рис. 4.3)?

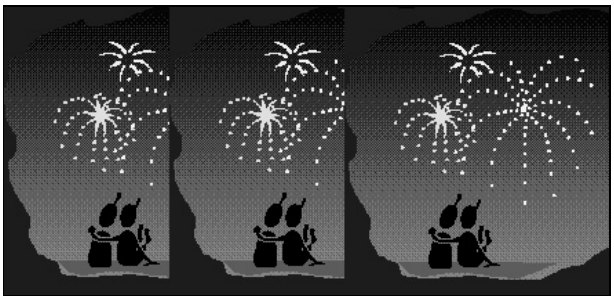


Рис. 4.3. Пикник

28. Переливашка.

Имеются три сосуда — А, В и С. Объем первого — 8 литров, второго — 5, третьего — 3. Первый заполнен до краев. Остальные пусты. Переливать позволяется из сосуда в сосуд ровно столько, сколько в него помещается (рис. 4.4). Например, первым ходом можно перелить из А в В 5 литров, в первом тогда останется 3. За наименьшее количество переливаний добиться того, чтобы в первом сосуде остался 1 литр.

А как, при соблюдении начальных условий, разделить эти 8 литров пополам?

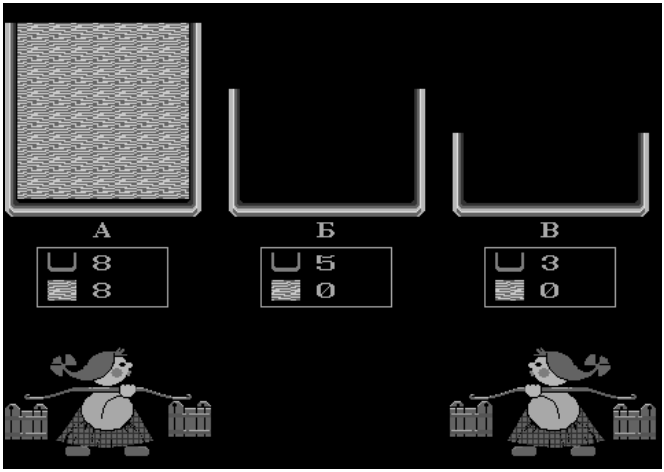


Рис. 4.4. Переливашка

29. Ханойские башни.

Есть такая притча. Давным-давно буддийские монахи задумали узнать, когда же наступит конец света. Явился им Будда и дал три золотых стержня, на одном из которых было 256 золотых колец, нанизанных как в детской пирамидке — от большого внизу к самому маленькому наверху. И сказал Будда: "Конец света наступит тогда, когда вы сможете переложить все кольца с первого стержня на третий, используя второй как промежуточный, по следующим правилам. Первое — за один раз можно брать только одно кольцо и только самое верхнее. Переносить его можно на второй или на третий стержень. Второе правило — большое кольцо нельзя класть на маленькое. И третье — на каждый перенос одного кольца вы не должны тратить больше 1 секунды". Испугались монахи, что так скоро конец света наступит, но ослушаться Будды не могли и немедленно приступили к работе. И вот уже 2000 лет, ежесекундно, не прерываясь и непрестанно меняя друг друга, перекладывают монахи кольца, а конца их работе все не видно.

Разработайте алгоритм переноса колец при начальном количестве колец 3 (рис. 4.5). Потом сделайте то же самое для 4-х колец, потом для пяти. Каково минимальное количество шагов для каждого из этих алгоритмов? Выведите общую формулу минимального количества шагов для любого исходного количества колец и попробуйте, воспользовавшись ею, хотя бы приблизительно определить, когда же нас ждет конец света?

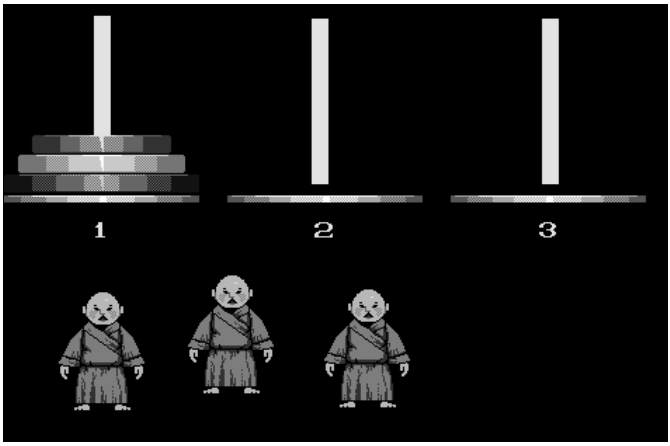


Рис. 4.5. Ханойские башни

30. Сумматор.

Предположим, что есть такой не шибко умный исполнитель команд — Сумматор, который может исполнять только две команды (зато очень быстро!) — прибавлять к предыдущему числу 1 или умножать предыдущее число на 2 (рис. 4.6). Изначально у него на индикаторе светится число ноль. Разработать алгоритм получения числа 100 за минимальное количество шагов.

31. Шахматные кони 1.

На доске 3×3 стоят 6 шахматных коней, как показано на рис. 4.7. Конь, как известно, ходит буквой "Г". Исходя из этого, запишите в шахматной нотации алгоритм, который меняет коней разного цвета местами. (Для тех, кто шахмат вообще никогда в глаза не видел, объясняем — конь с поля A1 может в данном случае пойти только на C2. Запись такая: 1. A1-C2 и т. д.)

32. Шахматные кони 2.

А теперь коней стало несколько меньше (рис. 4.8). Казалось бы, задача только облегчилась, но — впечатление очень обманчиво! Этот алгоритм гораздо сложнее, и опыт показывает, что терпения на его разработку хватает далеко не у всех.

Рис. 4.6. Сумматор

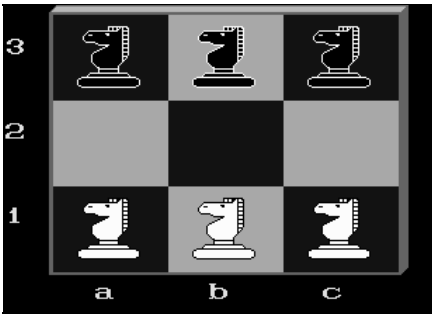


Рис. 4.7. Шесть коней

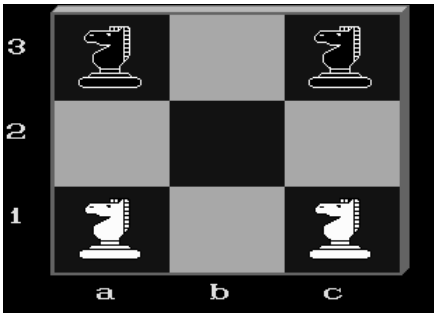


Рис. 4.8. Четыре коня

33. Взвешивание монет.

Еще один алгоритм — взвешивание на чашечных весах (рис. 4.9). Сначала пример. Допустим, у нас есть 6 монет. Одна из них фальшивая и весит легче, чем настоящие. Сколько взвешиваний на чашечных весах потребуется, чтобы однозначно определить фальшивую монету? Конечно, можно по очереди класть все монеты на весы, и тогда, если нам повезет, мы можем с первого раза определить фальшивую монету, а если очень не повезет, то только с пятого. Если взвешивать монеты парами — то опять же, в случае крупного невезения, мы определим монету уже на третий раз. Прогресс! Но если действовать, как подобает Homo Sapiens, то можно решить задачу за два действия! Первым действием кладем на чашки весов по три монеты. Какая-то из чашек окажется легче — значит в этой группе и есть фальшивая монета. Берем из этой группы любые две монеты и кладем их на весы по одной. Если весы в равновесии, значит фальшивая монета — третья из этой группы. Если какая-то из чашек весов оказалась легче — значит она там.

Ну, а теперь самостоятельно. (Кстати, сэр Исаак Ньютон долгое время был главным казначеем при королевском дворе и, в частности, добился больших успехов в борьбе с фальшивомонетничеством! А чем мы хуже?)

Итак, сначала попроще. Если вместо шести монет будет при тех же прочих условиях семь, то сколько взвешиваний понадобится? А теперь у нас 9 монет. Сколько взвешиваний понадобится?

Усложним условия. Пусть заранее неизвестно, легче или тяжелее фальшивая монета, известно только, что она отличается по весу. Как теперь будем взвешивать?

И наконец, еще одна алгоритмическая головоломка из этой серии, над которой автор в студенчестве поломал голову недели две, но пари выиграл. Задача такая. Имеется 12 монет, фальшивая отличается по весу, но неизвестно, в какую сторону. Надо определить ее за 3 и только 3 взвешивания. Более того, задача решается и для 13-ти монет! Дерзайте!

Рис. 4.9. Взвешивание монет

34. Угадай-ка.

Это такая игра, в которой один играющий загадывает число от 1 до 100, а второй пытается его отгадать, называя какие-нибудь числа из этого интервала. Ведущий отвечает, больше загаданное число названного или меньше. При правильном алгоритме наименьшее количество ходов, за которое при любом загаданном числе можно его угадать — семь. Найдите этот алгоритм. А каково минимальное количество шагов, если будет загадано число от 1 до 1000? А до 1 000 000? Не так уж и много. Проверьте!

35. Игра Баше.

Тоже очень древняя игра, которая алгоритмически решается в пользу одного из игроков. Правила таковы. Вначале имеется нечетное число предметов, количество которых описывается формулой $N = 4k - 1$, где

k — любое натуральное число. (Таким образом, N может быть равно 15, 19, 23 и т. д.) Играют двое. За один раз игрок может взять из кучи один, два или три предмета. Берут по очереди. Проигрывает тот, кто берет последний предмет. При правильной игре всегда будет выигрывать второй игрок. Найдите этот алгоритм.

36. Бочонок.

Имеется бочонок, который может вращаться вокруг вертикальной оси. В крышке бочонка проделаны четыре совершенно одинаковых и симметричных относительно центра отверстия, в любое из которых можно опустить руку. В бочонке под каждым отверстием помещен стакан (он стоит либо нормально, либо вверх дном). Увидеть что-либо через отверстие невозможно.

Правила игры следующие. В любые отверстия бочонка можно опускать руки (в одно отверстие — одну руку). Ими можно перевернуть один или два стакана. Можно не переворачивать.

После того как руки вытащены, бочонок вращают по или против часовой стрелки. Затем останавливают. Узнать те отверстия, куда опускались руки невозможно (напоминает наперсточников, правда?).

Затем в бочонок вновь опускают руки, и игра продолжается. Цель игры — добиться того, чтобы за минимальное количество ходов расположить все стаканы одинаково — либо все нормально, либо все вверх дном. В любой момент игрок может сказать, что задача выполнена, остановиться, и производится проверка расположения стаканов.

Казалось бы, алгоритма никакого быть не может — бочонок ведь движется случайным образом. Ан, нет! Мудрость человеческая (и, в частности ваша, уважаемый читатель) безгранична! Алгоритм есть. Ищите!

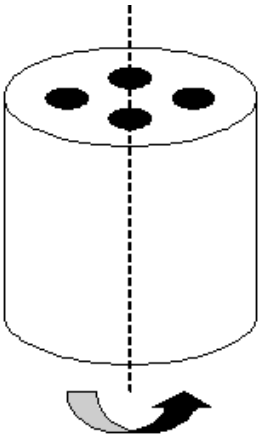


Рис. 4.10. Вращающийся бочонок с четырьмя стаканами внутри

Все дальнейшие задания предполагают преподавание основ программирования в школе на языках высокого уровня — Бейсике или Паскале (в данной книге алгоритмические конструкции рассмотрены на примере Бейсика), хотя сами задания практически не привязаны ни к какому конкретному языку и могут быть реализованы и на других алгоритмических языках.

Сначала введем понятие переменной.

4.2. Переменные и работа с ними

4.2.1. Переменные

Компьютер, как и вы, уважаемый читатель, обладает памятью. Она бывает разная. В процессе отладки вашей программы компьютер напрягает эту самую память: размещает в ней исходные данные, обрабатывает их, используя ваш алгоритм, получает результаты и доводит их до вашего сведения — чаще на экран монитора. Надо думать, для вас не секрет, что память эта называется ОЗУ (Оперативное Запоминающее Устройство), или по-английски RAM (Random Access Memory, память с произвольным доступом). Это, собственно, одно из основных устройств компьютера, имеющее, правда, ограниченный объем, измеряющийся в мегабайтах.

Если вы пишете достаточно сложную и уже нелинейную программу, то вам наверняка потребуются *переменные*, т. е. такие области этой самой оперативной памяти, которые имеют имя, данное нами, и значения, которые могут меняться. Имя переменной в ходе выполнения программы не меняется, а значение может меняться многократно. Этот процесс можно сравнить со сдаваемой на лето дачей. Дача — это участок (область памяти компьютера), имеющий уникальный, неповторимый и не меняющийся адрес, по которому ее можно найти (имя переменной), а каждое лето на дачу приезжают новые жильцы (значения переменной).

4.2.2. Арифметика в программировании

Прежде чем двигаться дальше ("Как трудно двигаться дальше" — поет Борис Гребенщиков), необходимо напомнить, что в те далекие времена, когда только зарождались алгоритмические языки, а словосочетание "персональный компьютер" вызывало у тех, кто его слышал, сомнения в здравомыслии его произносившего, так вот, в те самые времена считалось, что компьютер (от англ. *compute*, вычислять), т. е. "вычислитель", только и предназначен для того, чтобы считать в тысячи, нет — в миллионы раз быстрее человека. Что ж, по сути это действительно так, и если вы немного представляете себе физику происходящего в компьютере, то должны понимать: все, что вы ни делаете за компьютером — сочиняете стихи или музыку, рисуете картинки,

играете, общаетесь в Сети — все внутри этого "вычислителя" сводится к цифровой двоичной форме и к действиям, элементарным арифметическим действиям, над этими самыми числами.

Отсюда вывод — если хочешь быть программистом, надо дружить с математикой. Начнем?

Итак, компьютер умеет вычислять элементарные арифметические выражения. Но для того чтобы он смог это сделать, мы должны представить это самое выражение в понятном ему виде, а именно:

- в отличие от арифметики, выражение должно быть записано *в одну строку* безо всяких числителей и знаменателей;
- для записи арифметических действий допустимо использовать только перечисленные ниже знаки:
  - `+` (сложение, слева от клавиши <Backspace> или на малой цифровой клавиатуре <Серый плюс>);
  - `-` (вычитание, то же, что дефис, или на малой цифровой клавиатуре <Серый минус>);
  - `*` (умножение, там же, где цифра 8 на основной клавиатуре при нажатой клавише <Shift> или на малой цифровой клавиатуре <Серая звездочка>);
  - `/` (деление, на разных клавиатурах бывает в разных местах или на малой цифровой клавиатуре <Серый слэш>);
  - `^` (возведение в степень, при выбранном латинском шрифте там же, где цифра 6 на основной клавиатуре при нажатой клавише <Shift>);
  - `()` (скобки, там же, где цифры 9 и 0 на основной клавиатуре при нажатой клавише <Shift>);
- недопустим пропуск знака умножения между коэффициентом и переменной, как это возможно в алгебре (например, нельзя писать `2x`, а надо `2*X`, или нельзя `5d`, а надо `5*D`);
- дробная часть отделяется от целой *точкой*, а не запятой (нельзя писать `3,14`, а надо `3.14`);
- допустимо опускать в записи десятичной дроби ноль, стоящий перед точкой (вместо `0.123` можно `.123`).

Чтобы компьютер вычислил выражение правильно, необходимо помнить о приоритете выполнения действий. Тут все как в элементарной математике:

- сначала выполняются действия в скобках (в Бейсике скобки используются только круглые, в сложных выражениях они могут быть и двойные, и тройные, и т. д.);
- далее вычисляются функции, если они есть;

- затем выполняется возведение в степень;
- потом умножение и деление;
- в последнюю очередь — сложение и вычитание.

Действия одинаковой очередности выполняются слева направо.

Приведем ряд примеров перевода арифметических выражений в пригодный для Бейсика вид.

- В арифметике $\frac{1}{X}$ — на компьютере `1/X`
- В арифметике $2x^2$ — на компьютере `2*X^2`
- В арифметике $\frac{3+4}{8-6}$ — на компьютере `(3+4)/(8-6)`

Замечание

В последнем случае хочу обратить внимание на обязательность скобок как в числителе, так и в знаменателе. Потому что при записи `3+4/8-6` компьютер сначала произведет деление 4 на 8, затем прибавит 3 и вычтет 6. В ответе получится –2,5 вместо правильных 3,5. Будьте внимательны!

А чтобы потренироваться, вот вам ряд заданий на запись арифметических выражений в виде, пригодном для вычислений на компьютере.

37. $\frac{1}{5}+\frac{1}{6}+\frac{1}{7}+\frac{1}{8}+\frac{1}{9}$

38. $2^2+3^2+4^2+5^2+6^2$

39. $\frac{1+2}{3+4}:\frac{5+6}{7+8}$

40. $\frac{2^3+3^2+4^3}{9^3+5\times6\times7}$

41. $10\times\sqrt{25+11}$

Совет

Корень квадратный (впрочем, как и любой другой) всегда можно представить в виде степени. А вот как, узнайте на уроках математики! Да про скобки не забудьте!

42. $\cfrac{1}{2+\cfrac{3}{4+\cfrac{5}{6+\cfrac{7}{8}}}}$

43. $\sqrt{\sqrt[3]{2+\sqrt[4]{3}}}$

44. $\dfrac{\sqrt[4]{6}}{(5-\sqrt{8})^3}$

45. 81−0,25

46. $\sqrt{3+\sqrt{3+\sqrt{3+\sqrt{3}}}}$

47. $\sqrt{12^2+5^2}$

А теперь наоборот. Следующие компьютерные выражения запишите в обычной арифметической форме:

48. `(5^3*3-4)^.5`

49. `44/4*5+10-2^3/.3`

50. `5+2/6-7+3^2`

51. `5+2/6-7+3^2`

52. `(4*(5-4^2)^2`

53. `3+8/4-7*3^2`

Ну, и чтобы закончить с арифметикой, расскажу еще о двух действиях, которые вы навряд ли использовали в школе, а в программировании они могут оказаться весьма полезными. Эти действия применимы только к целым числам!

Первое из них — *деление нацело*. Для него используется знак `\` (так называемый обратный слэш). Компьютер в этом случае делит числа как обычно, но в качестве результата деления представляет только целую часть, отбрасывая дробную. Например:

- `41\4=10`
- `25\9=2`

и т. д.

И второе — *нахождение целого остатка от деления*. Для этого действия нет специального знака, и оно выполняется при помощи оператора `MOD`. Оператор `MOD` действует как и предыдущий, но в качестве результата представляет целочисленный остаток от деления. Например:

- `41 MOD 4 = 1`
- `25 MOD 9 = 7`

Действия деления нацело и нахождения целочисленного остатка выполняются до обычных умножения и деления.

Если это вам понятно, давайте вычислим несколько выражений.

54. `20\6`

55. `20 MOD 6`

56. `34\4`

57. `34 MOD 4`

58. `2\5`

59. `2 MOD 5`

60. `4*7\3 MOD 6/3`

61. `24 MOD 5\3`

4.2.3. Оператор присваивания

Представьте, что вы разработали алгоритм, продумали, какие в нем будут участвовать переменные, придумали им имена. И что же дальше? Как же сообщить компьютеру их значения? Как менять эти значения? Итак, мы знакомимся с первым оператором языка Бейсик — *оператором присваивания*. Знайте, что имя переменной после присваивания будет служить для компьютера своего рода ссылкой на адрес в памяти, где значение этой самой переменной разместилось.

В старых версиях Бейсика оператор присваивания всегда начинался со слова `LET`. Сейчас этого нет, и оператор присваивания выглядит просто как обычное математическое равенство. Но это вовсе не так!

Предупреждение

В левой части оператора присваивания может находиться только имя той переменной, в которую будет заноситься новое значение и больше ничего! В правой части оператора присваивания может находиться: конкретное значение, присваиваемое переменной (в зависимости от типа переменной — число, или символ, или строка); арифметическое или алгебраическое выражение, содержащее как конкретные числа, так и имена других переменных, уже имеющих значения; имя другой переменной, уже имеющей значение.

Обе части оператора присваивания соединяются знаком равенства.

Тип переменных в левой и правой частях оператора присваивания должен совпадать!

Когда интерпретатор языка Бейсик встречает в программе оператор присваивания, он обрабатывает его следующим образом: производятся все необходимые вычисления и операции в правой части оператора, находится конкретное значение (числовое или символьное), и это значение заносится в переменную, имя которой указано в левой части.

Предупреждение

При этом старое значение переменной, имя которой указано в левой части оператора присваивания, безвозвратно теряется. Значения переменных, участвующих в правой части, не изменяются.

Если вы в своей программе указываете впервые какую-либо переменную, не присвоив ей перед этим никакого значения, то интерпретатор по умолчанию считает ее значение равным нулю.

Синтаксис оператора присваивания

Произносится оператор присваивания следующим образом: "*Присвоить* Зет значение Икс минус три Игрек квадрат в скобках плюс семь" (рис. 4.11). Ни в коем случае нельзя говорить: "Зет равно" и далее по тексту.

Рис. 4.11. Пример использования оператора присваивания

Рассмотрим работу приведенного выше оператора присваивания на примере фрагмента маленькой программы:

```
X=15
Y=2
Z=(X-3*Y^2)+7
```

Вопрос. Чему будет равен `Z` после выполнения всех операторов присваивания?

Решение. Интерпретатор делает следующее:

- присваивает переменной `X` значение 15;
- присваивает переменной `Y` значение 2;
- вычисляет значение правой части третьего оператора присваивания `(15-3*2^2)+7`. Получается 10;
- присваивает переменной `Z` значение 10.

Ответ. После выполнения всех операторов присваивания значение `Z` будет равно 10.

Вопрос. А если бы первые два оператора присваивания отсутствовали, то чему было бы равно значение `Z`?

Ответ. Тогда интерпретатор принял бы значения `X` и `Y` по умолчанию равными 0, вычислил бы с этими нулями значение правой части и оказалось бы, что `Z` в этом случае равно 7.

Пока все было похоже на традиционную математику. Но есть один небольшой нюанс. Давайте рассмотрим такой фрагмент программы:

```
X=2
X=X+1
```

Если читать эту запись как математическую, то первый оператор совершенно обычен: "Икс равен двум". Но второй сразу заставляет бунтовать испорченную холодной логикой голову. Ведь так не может быть! Как это "Икс равен Икс плюс один"? Но что противоречит здравому смыслу в математике, легко поддается пониманию в программировании. Здесь это не математическое равенство, а рассматриваемый нами оператор присваивания. Разберем его работу.

Итак, *вопрос.* Чему будет равно значение `X` после выполнения двух операторов присваивания?

Решение. Сначала переменной `X` присваивается значение 2. Затем интерпретатор обращается к правой части второго оператора присваивания, находит в памяти значение `X`. Оно равно 2. Прибавляет к нему 1. Получается 3. Пересылает получившийся результат в правую часть, в ту же самую переменную `X`. Старое значение ее, равное 2, стирается, записывается новое: 3.

Ответ. После выполнения двух операторов присваивания значение переменной `X` будет равно 3.

Теперь упражнения. Если следующие операторы Бейсика написаны правильно, напишите слово "верно". Если нет, укажите на ошибку. Все операторы рассматривайте по отдельности, т. к. они не являются частью общей программы.

62. `3+6=Z`

63. `W=5+8`

64. `5+8`

65. `S+R=76`

66. `V=(X+Y+Z)/3`

67. `F+F-5`

68. `H=H*2`

69. `X=X*S`

70. `D=E=1`

71. `A+B=C-D`

72. Записать операторы, которые переменной `S` присваивают среднее арифметическое чисел `A`, `B` и `C`.

73. Записать операторы, которые переменной `S` присваивают расстояние между точками с координатами `X1`, `Y1` и `X2`, `Y2`.

74. Записать операторы, которые переменной `S` присваивают длину гипотенузы `C` по катетам `A` и `B`.

75. Записать операторы, которые переменной `S` присваивают площадь треугольника со сторонами `X`, `Y`, `Z`.

76. Записать оператор присваивания, который меняет знак у значения переменной `R`.

77. Чему будут равны значения переменных `X` и `Y` после выполнения следующих операторов?

```
X=3
Y=6
X=Y
Y=X
```

78. Поменять местами значения переменных `X` и `Y`.

79. Присвоить переменной `W` сумму цифр трехзначного числа `K`.

80. Присвоить переменной `S` значение длины светового года в километрах. Световой год — расстояние, которое свет проходит за астрономический год, т. е. за 365 суток. Скорость света принять равной 300 000 км/с.

81. Присвоить переменной `V` значение объема кирпича с размерами `A` — длина, `B` — ширина, `C` — высота.

82. Записать в форме оператора присваивания формулу перевода долларов США в рубли (курс по состоянию на июль 2002 г. — 31 руб. за 1 доллар).

4.2.4. Стандартные функции Бейсика

Теперь рассмотрим основные алгебраические функции, применяемые в языках программирования (опять-таки на примере Бейсика, хотя они почти аналогичны Паскалю).

Бейсик позволяет работать со следующими стандартными алгебраическими функциями (табл. 4.1).

Таблица 4.1. *Стандартные алгебраические функции Бейсика*

Функция	Описание	Пример	Результат
`ABS`	Возвращает абсолютную величину (модуль) аргумента	`? ABS(-2.56)`	2,56
`SQR`	Возвращает квадратный корень из аргумента	`? SQR(9)`	3
`SIN`	Возвращает значение синуса аргумента (аргумент указывается в радианах)	`? SIN(3)`	0,14112

Таблица 4.1 *(окончание)*

Функция	Описание	Пример	Результат
COS	Возвращает значение косинуса аргумента (аргумент указывается в радианах)	? COS(3)	0,98992599
TAN	Возвращает значение тангенса аргумента (аргумент указывается в радианах)	? TAN(3)	– 0,1425465
EXP	Находит значение показательной функции e^x, где x — аргумент, а e — основание натурального логарифма, равное ≈2,718...	? EXP(1)	2,718282
LOG	Возвращает логарифм по основанию e	? LOG(3)	1,09861229
INT	Возвращает целую часть аргумента (от англ. INTeger, целый)	? INT(30,1) ? INT(30,9)	30 30
SGN	Возвращает 1, если аргумент больше нуля; 0, если аргумент равен нулю; –1, если аргумент меньше нуля (от англ. SiGN, знак)	? SGN(3) ? SGN(0) ? SGN(-3)	1 0 –1

Предупреждение

Функция SQR применима только к положительным аргументам.

Любознательный читатель сразу заметит, что нет здесь милого его сердцу котангенса. Но ведь котангенс просто получить из тангенса, не правда ли?

Самые распространенные ошибки начинающих при написании стандартных функций (хотя вы все равно их допустите!):

- пишут AB**C** вместо AB**S** для модуля;
- пишут T**G** вместо T**AN** для тангенса.

Предупреждение

Главное правило при использовании стандартных функций — брать аргумент в скобки:

- нельзя SIN3, а надо SIN(3);
- нельзя SQR9, а надо SQR(9).

Стандартные функции могут входить в арифметические выражения. В качестве аргументов также можно использовать арифметические выражения.

Примеры правильной записи стандартных функций приведены в табл. 4.2.

Таблица 4.2. *Примеры записи стандартных функций*

Алгебра	**Бейсик**
$\sin 3 + \cos 6 + \mathrm{tg} 4$	`SIN(3)+COS(6)+TAN(4)`
$\sqrt{\lvert \sin 45^\circ - \cos^2 30^\circ \rvert}$	`SQR(ABS(SIN(45)-COS(30)^2))`

Перед упражнениями вспомним про *радианную меру углов*, потому что не все, к сожалению, любят тригонометрию. Итак,

1 радиан = 180°/π, отсюда 1 градус = π/180, следовательно sin30° надо записывать в Бейсике как `SIN(30*3.14/180)`, π≈3.14.

А вот теперь можно и поупражняться.

Запишите на Бейсике следующие выражения:

83. $\sqrt{1+z^3}$

84. $\lvert x+by \rvert$

85. $\cos^2 x^3$

86. 2^{2+x}

87. $\sqrt[3]{2+x}$

88. $\sqrt[7]{x^7+7^x}$

89. $\dfrac{a+\sin^3 b^2}{\cos 25 + \mathrm{ctg}\, 60}$

90. $\dfrac{(x^8+8^x)^{\frac{1}{8}}(a^2-(b+\sqrt[3]{\mathrm{ctg}\, x})}{3-\dfrac{4}{\sqrt{x}-\sin^2 a^3}}$

91. $\dfrac{xyz-3{,}3\lvert x-\sqrt[4]{y}\rvert}{10^7+\ln\sin^2 x}$

92. $\dfrac{(b+\cos^2 y^4)(ab+\mathrm{tg}(x+\sqrt[3]{y^2})}{\cos 11^0+|\mathrm{ctg}(y)|}$

93. $\dfrac{\sqrt{\dfrac{xb}{a}}+\cos^2(x+b)^3}{\dfrac{x^2(x+1)}{b}-\sin^2(x+a)}$

94. $\dfrac{\sin^3(x^2+a)^2-\sqrt[3]{\dfrac{x}{b}}}{\dfrac{x^2}{a}+\cos^3(x+b)}$

95. $\dfrac{m\,\mathrm{tg}(t)+\left|\sqrt[3]{c\sin(t)}\right|}{m\cos(bt\sin^2 t)+\mathrm{ctg}\,z}$

96. $b\,\mathrm{tg}^2 x-\dfrac{a}{\sin^2\dfrac{x}{a}}+\dfrac{a\mathrm{e}^{-\sqrt{a}}}{\cos(\dfrac{bx}{a})}$

Запишите в общепринятой математической форме:

97. `(-R+SQR(S^2-6*A*B))/(3*A)`

98. `X/Y*(D-F)+(X+Y)/D/F`

99. `X1+TAN(F2-V3)/3*ABS(X2-LOG(4)*Y3)/EXP(-2)`

100. `(LOG(a+x^2)+SIN(x/b)^2)*(x-SQR(ABS(x-a)))/(EXP(-c*x)SQR(x+a))`

101. `(a^(2*x)+b^(-x)*COS(a+b)*x)/(SQR(x^2+b)-b^2*SIN(x+a)^3/x`

102. `(a*x*SIN(2*x)+EXP(-2*x)*(x+b)^(1/3)/COS(x^3)^2-x/SQR(a^2)`

103. `(ABS(x^(y/x)-(y/x)^(1/3))/((y-x)*(y-1/TAN(z)/(y-x)/(1+(y-x)^2)`

104. `(SIN(x+m^3)^2/SQR(1+m^2*1/TAN(x)-c*m*LOG(m*x))/EXP(-b*x)`

105. `(EXP(-b*t)*SIN(a*t+b)-SQR(ABS(b*t+a)))/(b*SIN(t^2)^3-1)`

106. `(x^3*TAN(x+b)^2)^2+a/SQR(x+b)*EXP(a*x)*ABS(x^3-2)/(b-a^(1/4)`

4.3. Алгоритмическое программирование

4.3.1. Линейные алгоритмы

Для написания программ с линейным алгоритмом мы уже готовы. Подразумевается, что из уроков теории и некоторой практики вы уже знаете, как очистить экран, как поприветствовать пользователя по имени, запросить у

него необходимые данные, как решить вычислительные задачи, как вывести результат на экран в понятном и эстетичном для пользователя и проверяющего виде. Попробуем поупражняться. Следите все же за интерфейсом своих программ!

И сначала упражнения именно на это.

107. Напишите программу, запрашивающую ваш год рождения, год рождения вашей мамы и печатающую, во сколько лет мама вас родила.

108. Напишите дружественную по отношению к пользователю программу. Пусть она обратится к вам по имени, а затем предложит последовательно ввести прилагательное, существительное, наречие и глагол, чтобы в результате вывести на экран несложную фразу с использованием введенных слов.

109. Напишите программу, запрашивающую три стороны треугольника *A*, *B*, *C* и вычисляющую его площадь по формуле Герона:

$$S = \sqrt{p(p-a)(p-b)(p-c)},$$

где $p = (a+b+c)/2$ — полупериметр.

110. Напишите программу, запрашивающую высоту дома *h* (в метрах), ускорение свободного падения *g* и вычисляющую время падения кирпича *t* (в секундах) с крыши этого дома по формуле:

$$t = \sqrt{\frac{2h}{g}}.$$

Замечание

Помните, что в самой программе вы не указываете ни конкретных имен, ни конкретных слов, а лишь имена переменных, в которых они будут храниться после ввода с клавиатуры.

В следующих заданиях вам предлагается написать программы, производящие требуемые вычисления и выдающие на экран результаты этих вычислений.

111. Расстояние до ближайшей к Земле звезды Альфа Центавра 4,3 световых года. Скорость света принять 300 000 км/с. Скорость земного звездолета 100 км/с. За сколько лет звездолет долетит до звезды?

112. Вычислить количество прожитых составителем программы дней (учесть тот факт, что в високосном году 366 дней).

113. Известна теория биоритмов. С момента рождения жизнь человека подчиняется трем синусоидальным биоритмам. Физический цикл — 23 дня, эмоциональный — 28 дней и интеллектуальный — 33 дня. Первая половина каждого цикла — положительная, вторая — отрицатель-

ная. При переходе от положительной фазы к отрицательной в каждом цикле есть так называемый критический день. В физическом цикле — это 12-й день, в эмоциональном — 15-й, в интеллектуальном — 17-й. Когда два и более цикла находятся в отрицательной фазе или в критических днях, то в такие дни повышена вероятность физических недомоганий и травм, эмоциональных срывов и ссор, замедленности интеллектуальных процессов и реакции. В Японии, например, в крупных фирмах для каждого работника составлен график и в "плохие" дни их не допускают до работы, дают отдыхать, потому что оплата больничного или брак в работе обойдутся фирме дороже. Теперь к делу. Опираясь на результат предыдущего задания и используя операцию нахождения целочисленного остатка, рассчитайте, каков для вас сегодняшний день по всем трем циклам.

114. Запросите у пользователя курс доллара на сегодняшний день, затем имеющуюся у него рублевую сумму и рассчитайте, сколько долларов он может купить.

115. Дискета 3,5 дюйма вмещает 1,44 Мбайт. Рукопись содержит 450 страниц текста. На каждой странице 60 строк по 80 символов в каждой. Поместится ли рукопись на дискету, а если нет, то сколько таких дискет надо?

116. Документ содержит текст 32 строки по 60 символов в каждой и точечную черно-белую фотографию 10×15 см, каждый квадратный сантиметр содержит 300 точек, каждую точку описывают 4 бита. Каков общий информационный объем документа в килобайтах?

117. Запросите у пользователя длину ребра куба. Найти площадь грани, площадь полной поверхности и объем этого куба.

118. Есть притча о шахматах, где выигравший запросил у могущественного правителя, чтобы ему был выплачен выигрыш зернами пшеницы по следующим правилам: на первую клетку шахматной доски положить одно зерно, на вторую — два зерна, на третью — четыре, на четвертую — восемь и т. д., т. е. на каждую последующую в два раза больше зерен, чем на предыдущую. Сколько же зерен должен был бы получить выигравший? (Замечание: на каком-то этапе выполнения этого задания возможно переполнение памяти. Подумайте, как обойти это препятствие).

119. Кстати, для оценки предыдущего результата еще одно задание. В России ежегодно собирают около 90 млн. тонн зерновых. Масса одного зерна около 5 граммов. Сколько зерен в таком урожае, и сколько лет пришлось бы расплачиваться России по условиям предыдущего задания?

120. Продав квартиру, вы получили 22 000 долларов и положили их в банк. Банк начисляет 1% в первый месяц, а каждый следующий — тоже 1%, но уже с общей суммы с учетом начисленных процентов. Сколько денег будет в банке на вашем счету через год? (Нестабильностью нашей экономической системы пренебрегаем).

121. Допустим, вы получили наследство 1 000 000 долларов и хотите красиво пожить. После долгих раздумий вы решаете, что будете жить на 8000 долларов в месяц. На сколько лет вам хватит наследства?

122. Усложним задачу. А если при тех же условиях вы положите этот миллион в банк под 12% годовых?

123. Пушка стреляет под углом 30° к линии горизонта. Масса снаряда 30 кг, начальная скорость 500 м/с. Какова будет дальность полета снаряда? (Формулу вспомните из курса физики.)

124. Вычислить длину окружности, площадь круга и объем шара одного и того же радиуса.

125. Найти произведение цифр заданного четырехзначного числа.

126. Три сопротивления $R1$, $R2$ и $R3$ соединены параллельно. Найдите сопротивление соединения.

127. По координатам трех вершин некоторого треугольника найти его площадь и периметр.

128. Вычислить дробную часть среднего геометрического трех заданных положительных чисел.

129. Два поезда идут навстречу друг другу со скоростями 36 и 54 км/час. Пассажир, находящийся в первом поезде, замечает, что второй проходит мимо него в течение 6 секунд. Какова длина второго поезда?

130. Определить вес мальчика массой 40 кг, если он находится в центральной точке вогнутого моста, имеющего радиус кривизны 20 м. Скорость мальчика 10 м/с.

131. На вагонетку массой 800 кг, катящуюся со скоростью 0,2 м/с, погрузили мотоцикл массой 200 кг. На сколько при этом уменьшилась скорость вагонетки?

132. Расстояние до преграды, отражающей звук, равно 68 метров. Через сколько времени человек услышит эхо?

133. Кит плавает под водой со скоростью 27 км/ч. Он развивает мощность 150 кВт. Определить силу сопротивления воды.

134. Пуля, летящая со скоростью 400 м/с, ударяется в земляной вал и проникает в него на глубину 36 см. Сколько времени двигалась она внутри вала? С каким ускорением? Какова была ее скорость на глубине 18 см?

На какой глубине скорость пули уменьшилась в 3 раза? Движение считать равноускоренным. Чему будет равна скорость пули к моменту, когда она пройдет 99% своего пути?

135. Идет k-я секунда суток. Определить, сколько полных часов (H) и полных минут (M) прошло к этому моменту (например, $H = 3$ и $M = 40$, если $k = 13\ 237 = 3 \times 3600 + 40 \times 60 + 37$).

136. Определить угол (в градусах) α между положением часовой стрелки в начале суток и ее положением в H часов, M минут и S секунд ($0 \le H \le 11$, $0 \le M$, $S \le 59$).

137. Определить H — полное количество часов и M — полное количество минут, прошедших от начала суток до того момента (в первой половине дня), когда часовая стрелка повернулась на α градусов ($0 < \alpha < 360$, α — вещественное число).

Графика в Бейсике

При изучении и программировании линейных алгоритмов довольно часто дают графические задания. Это важно с той точки зрения, что учащиеся сразу видят результат своей работы на экране, когда из каких-то довольно абстрактных команд и операторов языка программирования вдруг появляются весьма красивые картинки. Да и понимание происходящего в компьютере становится более простым. Ознакомление с системой координат тоже не помешает.

Прежде всего, необходимо сказать несколько слов о включении графических режимов. При входе в оболочку Бейсика по умолчанию включается текстовый режим, в котором можно производить вычисления и выводить результаты на экран, но если мы хотим пользоваться графическими возможностями языка, нам надо объяснить это компьютеру посредством включения графического режима командой `SCREEN 12`. Графических режимов несколько, но автор рекомендует пользоваться режимом № 12, потому что он обладает наибольшей разрешающей способностью и позволяет получать качественные графические объекты. После включения графического режима мы можем давать компьютеру команды рисования *графических примитивов*.

Замечание

Графический режим включается только один раз.

В режиме `SCREEN 12` экран представляет собой координатную сетку с началом в левом верхнем углу, вправо от которого увеличивается координата X, а вниз — координата Y. Максимальное значение X на экране 640, а Y — 480 (рис. 4.12).

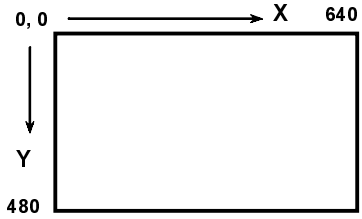


Рис. 4.12. Размеры и координаты экрана в режиме `SCREEN 12`

Графические примитивы

Ну что ж, начнем с графических примитивов, т. е. элементарных объектов, на которые можно разбить любое сложное изображение.

- Первый из них — *точка*. Для изображения точки используется оператор:

  ```
  PSET(X, Y), C
  ```

  где `X` и `Y` — координаты точки на экране, а `C` — ее цвет. Если цвет не указан, то точка будет изображена последним установленным цветом (это касается и всех прочих графических примитивов). Например:

  ```
  SCREEN 12
  PSET (320, 240), 4
  ```

  В центре черного экрана появится красная точка.

  Сразу же хочется заметить, что из эстетических соображений лучше поменять цвет фона, потому что черный экран — это мрачно. Например:

  ```
  SCREEN 12
  COLOR ,1
  PSET(320,240),4
  ```

  Теперь та же точка будет в центре синего экрана. Уже лучше.

- Следующий примитив — *отрезок прямой линии*. Оператор выглядит следующим образом:

  ```
  LINE (X1, Y1)-(X2, Y2), C
  ```

  где `X1`, `Y1` — координаты начала отрезка; `X2`, `Y2` — координаты конца отрезка, а `C`, как всегда, цвет.

  Например, мы хотим изобразить отрезок синего цвета с координатами (рис. 4.13) на желтом экране.

  Программа будет выглядеть так:

  ```
  SCREEN 12
  COLOR 1, 14
  LINE (100, 300) — (600, 50), 1
  ```

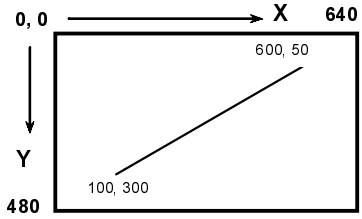


Рис. 4.13. Отрезок прямой линии

- Как известно из геометрии, по диагонали прямоугольника можно всегда его достроить. Известно это и Бейсику. Поэтому можно одним оператором построить *прямоугольник со сторонами, параллельными экрану*. Оператор выглядит так:

```
LINE (X1, Y1) – (X2, Y2), C, B
```

То есть к обычному оператору отрезка добавляется буква, именно буква, а не число `B` (от английского слова BOX — ящик, коробка). А координаты — это начало и конец одной (любой!) из диагоналей прямоугольника. Возьмем предыдущий пример и добавим там букву `B` в операторе `LINE`:

```
LINE (100, 300) – (600, 50), 1, B
```

В результате его выполнения у нас получится уже не отрезок, а прямоугольник, причем сама диагональ видна не будет (рис. 4.14).

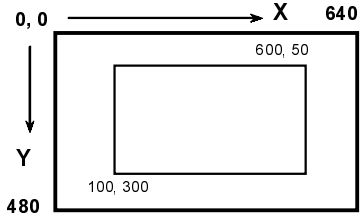


Рис. 4.14. Прямоугольник со сторонами, параллельными экрану

Предупреждение

Будьте внимательны с координатами. Довольно часто вместо координат диагонали указывают координаты стороны прямоугольника. В таком случае вы получите просто отрезок, параллельный экрану.

- И еще один примитив, основанный на операторе `LINE`. Он позволяет не только строить прямоугольник по диагонали, но и сразу *закрашивать* его. Пишется он так:

  ```
  LINE (X1, Y1) – (X2, Y2), C, BF
  ```

  Здесь добавилась еще и буква `F` (без пробела после `B`!), вместе с буквой `B` обозначающая BOX FULL — полная коробка. Это заставляет компьютер залить полученный прямоугольник указанным цветом `C`.

  Попробуйте самостоятельно использовать предыдущий пример, добавить букву `F` и посмотреть, что получится.

Замечание

Одним оператором `LINE` можно получать только прямоугольники со сторонами, параллельными экрану. Другие фигуры, состоящие из отрезков прямых, можно рисовать только последовательностью операторов `LINE` без всяких букв `B` и `F`.

Ну, что ж, для начала хватит. Надо бы поупражняться. Но перед упражнениями хотелось бы дать совет, как облегчить себе жизнь.

Чтобы быстро и четко нарисовать заданный объект, рекомендуется взять листок миллиметровки или бумаги в клеточку, нарисовать экран в масштабе 1 см = 50 экранных точек. Это получится 13 см в ширину и 10 см в высоту. Затем изобразить там то, что вы хотите получить, и тогда все координаты будут у вас, как на ладони (рис. 4.15).

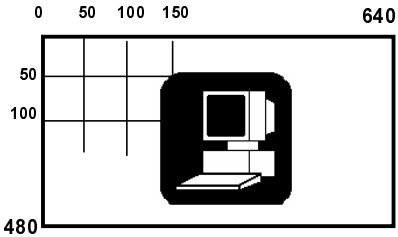


Рис. 4.15. Делать надо так!

К сожалению, опыт автора показывает, что многим этим заниматься лень — отсюда пустая трата сил и времени и очень неказистые результаты.

Напишите программу, изображающую на экране:

138. Четыре точки разного цвета в центре желтого экрана. Точки расположены квадратом. Расстояние между точками 5 точек.

139. Ромб с диагоналями (рис. 4.16).

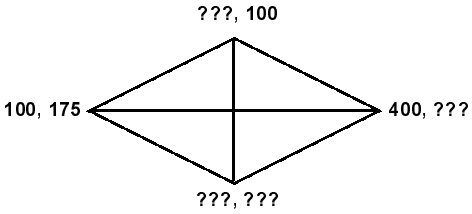


Рис. 4.16. Ромб с диагоналями

Пропущенные координаты определите сами, зная свойства ромба.

140. Российский флаг (есть, кстати, мнемоническое правило для запоминания цветов нашего флага — это магическое слово БеСиК — БЕлый, СИний, Красный — и на Бейсик похоже, и не забудешь никогда!). Флаг состоит из трех полос, поэтому рисовать их надо используя команду `LINE` с параметром `BF`, главное — четко рассчитать координаты диагоналей всех трех прямоугольников.

141. Рамку с конвертом (рис. 4.17).

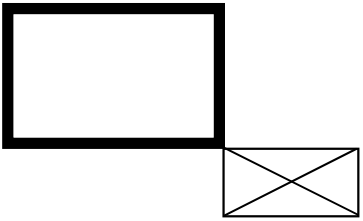


Рис. 4.17. Рамка с конвертом

Рамка рисуется двумя операторами `LINE` с параметром `BF`, конверт — подумайте сами. Координаты тоже выберите сами. Только не мельчите. Рисунок должен быть крупным и наглядным. Преподаватели это любят!

142. На одном экране куб и пирамиду (рис. 4.18).

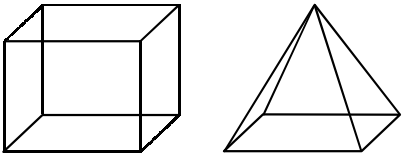


Рис. 4.18. Куб и пирамида

- Следующий этап — построение *окружности*. Все, что нужно знать об окружности для Бейсика — это координаты ее центра и радиус. Оператор выглядит так:

  ```
  CIRCLE (X, Y), R, C
  ```

  где `X`, `Y` — координаты центра; `R` — радиус (в экранных точках); `C` — цвет.

  Например, оператор:

  ```
  CIRCLE (320, 175), 50, 2
  ```

  изобразит нам зеленую окружность радиусом 50 в центре экрана.

- Далее переходим к *эллипсу*. Эллипс — это, по сути, окружность, но сжатая по вертикали или горизонтали, и вместо одного радиуса, как у окружности, у эллипса их два — по осям X и Y.

  Частное от деления R_y на R_x дает нам так называемый *коэффициент сжатия*:

  $$K = \frac{R_y}{R_x}.$$

  Таким образом, для эллипсов, сжатых по вертикали (рис. 4.19), коэффициент сжатия будет в пределах от 0 до 1.

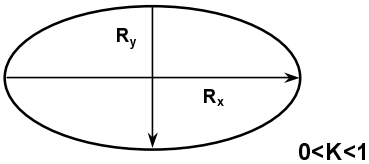


Рис. 4.19. Эллипс, сжатый по вертикали

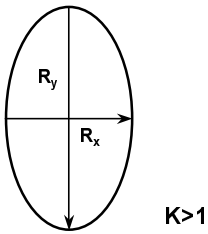


Рис. 4.20. Эллипс, сжатый по горизонтали

А для эллипсов, сжатых по горизонтали (рис. 4.20), коэффициент сжатия будет больше 1.

Очевидно, что если коэффициент сжатия равен 1, то это будет уже никакой не эллипс, а просто окружность.

Оператор рисования эллипса записывается почти точно так же, как и для окружности:

```
CIRCLE (X, Y), R, C, , , K
```

где `X`, `Y` — координаты центра эллипса; `R` — радиус той окружности, из которой этот эллипс получился; `C` — цвет; `K` — численное значение коэффициента сжатия.

Например, после выполнения оператора:

```
CIRCLE (320, 175), 50, 2, , ,.5
```

в центре экрана появится эллипс зеленого цвета, сжатый сверху и снизу, у которого R_y в два раза меньше, чем R_x.

Если же добавить еще один оператор

```
CIRCLE (320, 175), 50, 2, , , 2
```

то на наш эллипс наложится еще один, сжатый уже с боков, и R_y у него будет в два раза больше, чем R_x. Вместе они должны составить картинку, представленную на рис. 4.21.

Ну, а теперь продолжим наши упражнения. Напишите программы, которые отобразят на экране следующие изображения:

143. Квадрат и вписанную в него окружность.

Замечание

К сожалению, иногда на мониторе размеры точек по осям абсцисс и ординат не совпадают, вследствие чего может наблюдаться асимметрия в изображениях (при полной правильности всех ваших расчетов). В таких случаях приходится подгонять детали рисунка вручную.

144. Мишень — пять концентрических (то есть с одним центром) окружностей, вложенных друг в друга, и перекрестие двух отрезков прямых. Можно усложнить изображение, добавив попавшую в мишень стрелу (рис. 4.22).

145. Снеговика, состоящего из пяти окружностей разного радиуса и ведерка на голове из эллипса и двух отрезков (рис. 4.23).

146. Цилиндр, состоящий из эллипсов и двух вертикальных линий (рис. 4.24).

147. Летающую тарелку (рис. 4.25).

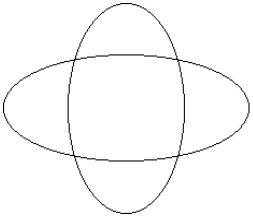


Рис. 4.21. Два пересекающихся эллипса

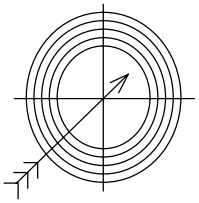


Рис. 4.22. Мишень

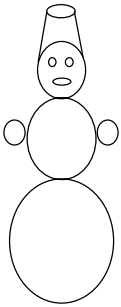


Рис. 4.23. Снеговик

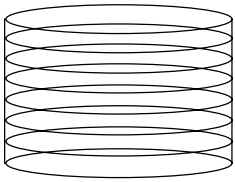


Рис. 4.24. Цилиндр

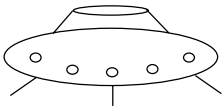


Рис. 4.25. Летающая тарелка

Ну, что же, если получается, то пора постепенно переходить к более сложным вещам, а именно — к рисованию *дуг окружностей* и *эллипсов*. Но прежде следует вспомнить, что в Бейсике применима только радианная мера углов *(см. разд. 4.2.4)*. Любая дуга имеет угол, от которого она начинается, и угол, которым она заканчивается. Чтобы разобраться с этим, сначала приведем оператор рисования *дуг окружностей*:

```
CIRCLE(X, Y), R, C, a, b
```

где появившиеся две новые величины `a` и `b` как раз и обозначают эти углы. Правило рисования дуг на Бейсике звучит так:

дуга строится от угла `a` *к углу* `b` *против часовой стрелки.*

Для того чтобы показать это наглядно, лучше всего обратиться к тригонометрической окружности (рис. 4.26).

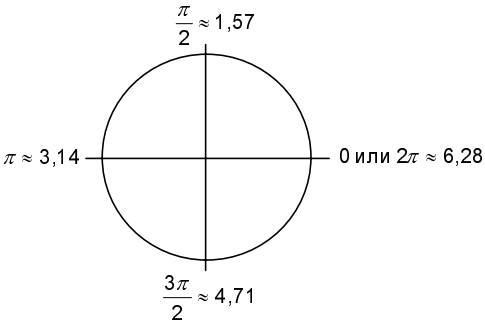


Рис. 4.26. Тригонометрическая окружность

Глядя на тригонометрическую окружность, мы всегда довольно легко можем определить начало и конец нужных нам дуг. Более того, Бейсик позволяет

использовать в качестве операндов в своих командах арифметические выражения. Поэтому, если вы знаете угол начала дуги, например 30°, но затрудняетесь определить его на тригонометрической окружности, то можете в соответствующем месте оператора дуги написать:

```
3.14*30/180
```

и пусть компьютер сам мучается! На то его люди и придумали.

Приведем несколько простых примеров:

- Построим верхнюю половину окружности синего цвета, радиусом 30, у которой центр находится в точке 150, 100 (рис. 4.27).

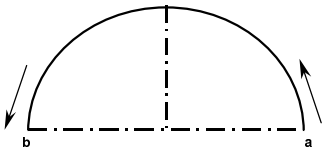


Рис. 4.27. Верхняя половина окружности

Первым делом, глядя на дугу, определяем направление против часовой стрелки. Это дает нам возможность выяснить, где начало и где конец дуги. Далее обращаемся к тригонометрической окружности. В этом простом примере сразу видно, что угол начала дуги `a = 0`, а угол конца `b = 3.14`. Таким образом, оператор построения этой дуги выглядит так:

```
CIRCLE (150, 100), 30, 1, 0, 3.14
```

- Построим правую половину той же самой окружности (мы уже не будем сопровождать это построение рисунком, т. к. процесс аналогичен).

  По тригонометрической окружности определяем, что `a = 4.71`, а `b = 1.57`. Записываем оператор построения дуги:

```
CIRCLE (150, 100), 30, 1, 4.71, 1.57
```

Предупреждение

Прежде, чем мы приведем более сложный пример дуги, хочется сделать еще одно полезное замечание. Если в операторе построения дуг поставить знак минус перед значениями углов дуги, то автоматически будут проведены радиусы, соединяющие центр окружности с концами дуг. Нельзя ставить минус перед 0. В этом случае вместо нуля надо использовать –6,28 (то есть 2π).

- Построим вот такое изображение CD-MAN (радиус "глаза" и координаты его центра определим из здравого смысла) (рис. 4.28).

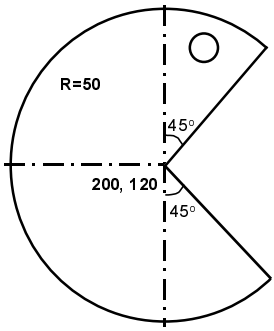


Рис. 4.28. CD-MAN

Хотя на первый взгляд кажется, что это очень сложно, на самом деле нам уже достаточно инструментов, чтобы сделать это двумя изящными легкими движениями. Сначала, действительно, давайте определим радиус "глаза". Диаметр его составляет ориентировочно пятую часть радиуса "туловища", стало быть радиус — одну десятую, т. е. 5. Координаты центра примерно `X` = 215, `Y` = 85. Теперь займемся углами дуги "туловища". Угол `a` определяется довольно легко — он равен 45°, а вот для определения угла `b` надо пройти почти всю тригонометрическую окружность — первая четверть, вторая четверть: еще +90°, третья четверть: еще +90°, и, наконец, еще +45°. Итого `b` = 315°. Ну, что ж, все данные для построения есть. Приступим

- "туловище":

```
CIRCLE (200, 120), 50, 1, -45*3.14/180, -315*3.14/180
```

- "глаз":

```
CIRCLE (215, 85), 5, 1
```

Еще раз напоминаю — в программировании главное — это предварительная подготовка всех исходных данных!

Теперь *дуги эллипсов.*

Когда мы говорили о построении самих эллипсов, то в операторе такого построения было несколько странно видеть три идущих подряд запятых. Сейчас все стало понятно — это место для углов `a` и `b`. Остальное — аналогично дугам окружности. Итак, оператор построения дуг эллипса:

```
CIRCLE (X, Y), R, C, a, b, K
```

В качестве примера нарисуем "светит месяц, светит ясный" (рис. 4.29).

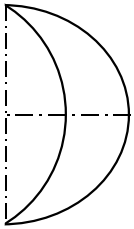


Рис. 4.29. Месяц

Внешняя граница месяца — это дуга окружности, а внутренняя — дуга эллипса. Центр у них общий (`X` = 500, `Y` = 70). Коэффициент сжатия эллипса — 2. Начала и концы дуг тоже одинаковы (`a` = 4.71, `b` = 1.57). Радиус окружности возьмем 50, а вот с радиусом эллипса придется помучиться, поскольку, как мы уже говорили, экранные точки имеют разные ширину и высоту, примерно определяющиеся соотношением 3 к 4. Поэтому в качестве радиуса эллипса возьмем значение 50×3/4. Тогда месяц построится следующим образом:

- дуга окружности:

```
CIRCLE (500, 70), 50, 14, 4.71, 1.57
```

- дуга эллипса:

```
CIRCLE (215, 85), 50*3/4, 4.71, 1.57, 2
```

Уже очень хочется попробовать свои силы? Еще немножко. Ведь мы хотим делать красиво. И нам нужно изучить еще один шаг — *закраску замкнутых контуров*.

После того как мы научимся закрашивать, у нас в руках будут практически все необходимые инструменты для рисования сложных объектов, разработки больших линейных алгоритмов.

Итак, оператор закраски:

```
PAINT (X, Y), C1, C2
```

где `X`, `Y` — координаты *любой* (!) точки внутри закрашиваемого контура; `C1` — цвет, которым закрашивается контур; `C2` — цвет самого контура. Если эти цвета совпадают, то достаточно указать `C1`.

Правила закраски:

- контур должен быть *замкнут*. Если в нем будет прокол хотя бы в одну экранную точку, то вся "краска" вытечет и зальет экран;
- контур должен быть *одноцветен*. Если составляющие даже замкнутого контура разных цветов, то для компьютера это то же, что разрыв. Будет закрашен весь экран;
- координаты точки закраски должны лежать *внутри* контура.

Рекомендуется закрашивать контур непосредственно после того, как он нарисован. Часто рисуют все изображение, потом начинают закрашивать смежные области, а линии разного цвета, пересекаясь, дают разрывы, и вот уже экран затекает самыми разными красками.

Если точка закраски попала вне контура, то закрасится весь экран, за исключением этого контура (иногда это бывает нужно, но редко).

Если точка закраски попала на контур, то ничего не закрасится.

Есть общая рекомендация из личного опыта автора, немного удлиняющая процесс отладки программы, но дающая абсолютно правильный результат закраски.

Совет

Перед закраской вместо оператора `PAINT` лучше написать оператор `PSET` с теми же координатами, но какого-нибудь контрастного цвета, чтобы ее было хорошо видно. Запустить программу и убедиться, что все вышеназванные правила соблюдены, и точка находится внутри закрашиваемого контура. После этого возвращаемся в программу, изменяем `PSET` на `PAINT`, и дело в шляпе!

Закрасим месяц из предыдущего примера белым цветом. После того как мы его нарисовали, у нас добавится всего одна команда:

```
PAINT (530, 70), 15, 14
```

Откуда взялось число 530 в качестве координаты `X`, надо думать, понятно, если посмотреть на рисунок, вспомнить координаты центра и радиус.

Ну, что ж, а теперь упражнения на дуги и закраску. Изображения уже более сложные, требующие достаточно основательной предварительной подготовки. Не забывайте о ней!

Напишите программы, выводящие на экран следующие картинки:

148. С теми же значениями, что и в примере CD-MAN, только углы, видимо изменятся, да координаты центра "глаза" (рис. 4.30).

149. Закрасьте "туловище" одним цветом, а "глаз" — другим.

150. Месяц в обратную, чем в примере, сторону. Закрасьте его красным цветом.

151. Закрашенный квадрат (одним оператором!), с вписанными в него двумя разноцветными окружностями (рис. 4.31)

152. Петлю гистерезиса, которая состоит из двух отрезков прямых линий и четырех дуг, являющихся четвертями окружности (рис. 4.32). Попробуйте сначала нарисовать изображение в масштабе на бумаге, рассчитать все исходные данные, и после этого все у вас получится!

153. Закрасьте получившееся изображение. (Будьте внимательны при состыковке дуг, зачастую там образуются разрывы!)

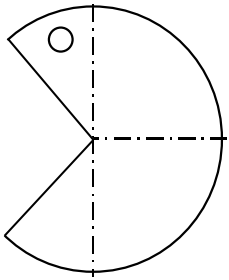


Рис. 4.30. CD-MAN, глядящий на запад

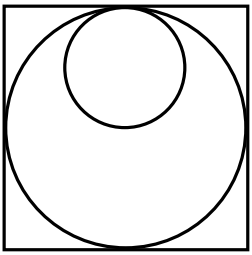


Рис. 4.31. Квадрат с окружностями

154. Пятиконечную звезду, вписанную в окружность и закрашенную как на рис. 4.33.

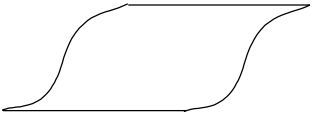


Рис. 4.32. Петля гистерезиса

Рис. 4.33. Звезда

155. Корабль в ночном море под российским флагом (рис. 4.34).

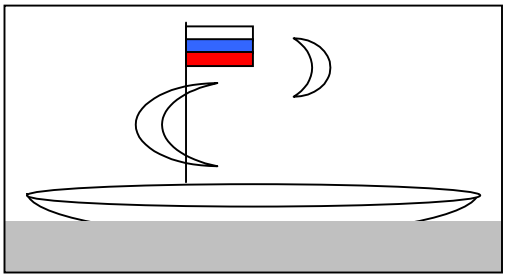


Рис. 4.34. Корабль в ночи

Это задание уже представляет собой достаточно сложное изображение. Попробуйте выполнить его так, как вы считаете нужным, а мы хотим определить, как это нужно делать.

Правила построения сложных изображений

1. Обязательно начертите изображение на бумаге в масштабе.
2. Разбейте изображение на те самые графические примитивы, каждый из которых вы могли бы нарисовать каким-либо уже изученным оператором.
3. Найдите и выделите на рисунке опорные точки — концы отрезков, начала и концы дуг, центры окружностей, эллипсов, отдельные точки (если они есть). Наметьте точки для оператора закраски `PAINT`. Если в изображении есть прямоугольники со сторонами, параллельными экрану, то необходимо выделить координаты их базовых диагоналей, по которым они будут строиться.
4. Определите максимальный размер изображения в экранных точках, исходя из размеров экрана.
5. Определите координаты всех опорных точек, значения радиусов окружностей и эллипсов, коэффициенты сжатия эллипсов, углы всех дуг.
6. Определите последовательность построения и закраски замкнутых контуров.
7. Напишите программу на компьютере, снабжая ее как можно бóльшим количеством необходимых комментариев (можно сначала на бумаге), и отладьте ее.

Предупреждение

Запускать программу необходимо не после ее полного написания, а после уже первого логически законченного фрагмента изображения, чтобы иметь возможность сразу оценивать правильность того или иного вашего программного решения.

156. Попробуйте самостоятельно построить сложное изображение по вышеизложенным правилам на примере рис. 4.35.

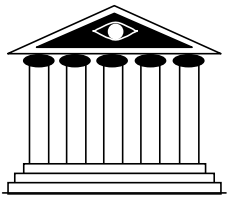


Рис. 4.35. Храм

157. А вот еще несколько вариантов сложных изображений (рис. 4.36).

158. Так как программированием чаще занимаются представители сильного пола, а среди них, как известно, бывает довольно много любителей футбола, то предлагаю изобразить эмблему своей любимой команды.

У меня это, конечно же, "Зенит"! Вот, кстати, как выглядит их эмблема (рис. 4.37). Сложно? Конечно, да! Но ведь ради любимой команды можно и постараться!.

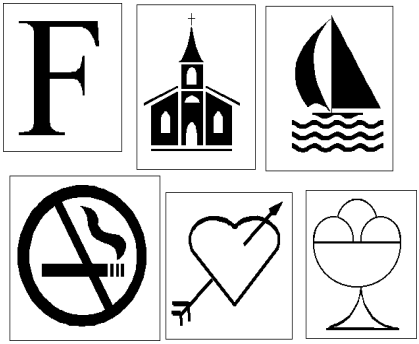


Рис. 4.36. Буква F, католический костел, яхта, знаки "Не курить" и "Любовь", мороженое

Рис. 4.37. Эмблема футбольной команды "Зенит"

159. В качестве самостоятельного задания придумайте себе сами эмблему и нарисуйте ее.

Для расширения возможностей машинной графики Бейсика был дополнительно разработан специальный макроязык GML (Graphics Macro Language). Он позволяет строить довольно сложные изображения и быстро выводить их на экран. Каждая команда языка (табл. 4.3) представляет собой латинскую букву, после которой следует один или два числовых параметра (как правило, целые числа).

Таблица 4.3. *Команды языка GML*

Команда	Действие
`Un`	Переместиться вверх на `n` точек
`Dn`	Переместиться вниз на `n` точек
`Ln`	Переместиться влево на `n` точек
`Rn`	Переместиться вправо на `n` точек
`En`	Переместиться по диагонали вверх и вправо на `n` точек
`Fn`	Переместиться по диагонали вниз и вправо на `n` точек
`Gn`	Переместиться по диагонали вниз и влево на `n` точек
`Hn`	Переместиться по диагонали вверх и влево на `n` точек
`M x, y`	Переместиться в точку с координатами `x` и `y`

Таблица 4.3 *(окончание)*

Команда	Действие
`M ±n, ±m`	Переместиться по отношению к текущей позиции на `±n` точек по оси `X` и на `±m` точек по оси `Y`. Знаки плюс и минус проставлять обязательно
`At`	Поворот изображения против часовой стрелки вокруг точки, с которой начиналось рисование, на 90°×`t` (`t` может принимать значения 0, 1, 2, 3). Действует во всех дальнейших командах до нового назначения
`Cn`	Задание нового цвета. Действует во всех дальнейших командах до нового назначения
`Sn`	Расстояние, указанное в командах перемещения, умножается на `n`/4 (0≤`n` ≤255)
`B`	Переместиться в новую позицию при помощи следующих за `B` команд, но рисование не производить. Отменяется установкой цвета `C`
`N`	Выполнить следующую команду перемещения и вернуться в исходную позицию; может предшествовать любой команде перемещения
`Pc1,c2`	Команда заполнения контура цветом: `c1` — цвет заполнения, `c2` — цвет контура. Должны соблюдаться все вышеизложенные правила закраски, и курсор должен находиться внутри закрашиваемого контура

Основные команды перемещения легко запомнить с помощью рис. 4.38.

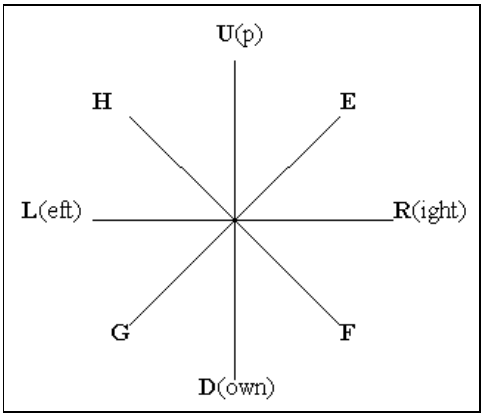


Рис. 4.38. Основные команды перемещения языка GML

Для приведения в действие последовательности команд языка GML необходимо использование оператора `DRAW`.

Например, команда:

```
DRAW "R50 D50 L50 U50"
```

изобразит квадрат, начиная от последней графической точки.

Если такой точки не было, и вы сразу начинаете строить изображение с помощью оператора DRAW, то по умолчанию исходной точкой считается центр экрана.

Если вы подготовите сначала какое-либо изображение в строковой переменной, то потом его можно будет многократно вызывать на экран. Для этого используется функция X. Ее работу рассмотрим на примере.

```
N$="R50 D50 L50 U50"
DRAW "C14 X"+VARPTR$(N$)
DRAW "BM500, 50 C2 X"+VARPTR$(N$)
```

С помощью первого оператора запоминаем последовательность команд построения квадрата в строковой переменной N$. Вторым оператором задаем цвет 14 и сообщаем компьютеру командой X, что необходимо выполнить последовательность команд N$. Это делает функция VARPTR$. Последним оператором перемещаемся без рисования в точку с координатами 500, 50, меняем цвет на 2 и снова выполняем последовательность команд N$.

В результате на экране в центре и в правом верхнем углу должны появиться одинаковые разноцветные квадраты со стороной 50.

Совместное использование графических примитивов и оператора DRAW позволит вам рисовать на вашем экране очень качественные изображения.

160. Для того чтобы опробовать эти возможности, нарисуйте елочку (рис. 4.39).

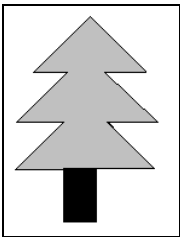


Рис. 4.39. Елочка

161. Если вам удалось предыдущее задание, то попробуйте вырастить на экране еловый бор из трех одинаковых елочек, пририсовать к ним траву, солнце, небо, облака — в общем, все, что подскажет вам фантазия.

Вывод текстовой информации в графике

Часто хочется сопроводить свои рисунки подписями. Не сдерживайте свои желания. Делайте это просто — с помощью операторов `LOCATE` и `PRINT`, как мы это уже многократно делали.

Предупреждение

Только надо помнить, что координаты в операторе `LOCATE` не графические (640×480), а текстовые (80×25).

4.3.2. Разветвляющийся алгоритм

Ну, что ж, надеемся, что с линейными программами особых трудностей у вас не возникло. Пиши команды подряд (хотя, конечно, все сначала желательно продумать), и изображения получатся просто великолепные!

Однако надо двигаться вперед. А дальше у нас — *разветвляющийся алгоритм*. Что это такое, мы уже знаем. Например, витязь на перепутье: "Если налево пойдешь, коня потеряешь, если прямо пойдешь — смерть найдешь", ну и т. д.

А вот как это реализуется на практике в Бейсике, мы узнаем сейчас.

Ветвление в алгоритме и программе осуществляется двумя способами:

- на основе *безусловного* перехода;
- на основе *условного* перехода.

Начнем с первого.

Безусловный переход

Безусловный переход оператором `GOTO` предписывает программе свернуть с линейного пути и, беспрекословно повинуясь, перейти к метке, расположенной в любом месте программы.

В качестве метки используются натуральные числа с двоеточием после них. Метка указывается только в начале строки (то есть если в строке программы несколько операторов, то нельзя ставить метку, например, перед вторым). Приведем пример программы, рисующей на экране три символа звездочки по диагонали:

```
CLS
?"*"
?"?*"
?"??*"
```

Символ `?` в данном случае означает пробел.

Добавим теперь метку 1 рядом со второй строкой и оператор GOTO в конце программы. Получаем:

```
CLS
1: ?"*"
?"?*"
?"??*"
GOTO 1
```

Казалось бы, изменения незначительные, но запустите программу и посмотрите что происходит! Вы в панике? Вы пытаетесь прекратить действие программы, но ничего не получается? Программа зациклилась! Что делать? Спокойно нажмите клавишу <Ctrl> и, не отпуская ее, клавишу <Break>. Это называется "прервать выполнение программы". Переведите дух и давайте шаг за шагом разберемся, что же произошло.

Программа очищает экран, затем рисует первую звездочку, под ней с отступом в один пробел — вторую, потом ниже третью, а вот затем получает команду GOTO 1. Это приказ, которого невозможно ослушаться. Программа ищет метку, находит ее у команды рисования первой звездочки и начинает снова последовательно выполнять команды рисования — первую, вторую, третью и — снова приказ перехода, и так — до бесконечности (вернее, до прерывания программы). А поскольку оператор PRINT, не снабженный точкой с запятой, после выполнения всегда переводит курсор на следующую строку, то и получается эффект движения.

А теперь поупражняйтесь.

162. Попробуйте спрогнозировать, что произойдет, если метку 1 в программе рисования звездочек указать в первой строке перед оператором CLS? А что, если в четвертой строке? Или (кощунственная мысль!) перед самим оператором GOTO? Проверьте свои предположения на практике.

163. Напишите программу, которая заставит бежать по экрану следующий узор:

```
 ###
#######
      #
      #
      #
#######
 ###
```

164. Если вы обратили внимание на текст программы про звездочки, то должны были заметить, что узор легко прослеживается. Воспользовавшись этим, напишите программу, которая заставит бежать по экрану слово "РОК", а потом, если не лень, то и свое имя. Буквы этих слов надо изобразить символами клавиатуры (какими вам больше нравится).

165. Посмотрите на текст следующей программы:

```
X=0
1: X=X+1
? X
GOTO 1
```

Как вы думаете, при каком значении `X` программа закончит свою работу? Если у вас достаточно быстродействующий компьютер, можете рискнуть проверить свое предположение на практике. Да, впрочем, рискните и на медленном, ведь программу всегда можно прервать. Правда, в этом случае вы так и не узнаете тайны, какое самое большое целое число компьютер может вывести на экран.

От автора

Строго говоря, серьезное программирование не любит оператор `GOTO`. Считается, что он засоряет и путает программу. Вызывает зацикливание и зависание компьютера. Рекомендуют обходиться без него. А мне он все равно нравится. Но это мое личное даже не мнение, а просто чувство. При написании серьезных программ я тоже стараюсь обходиться без `GOTO`.

Условный переход

Мало того, что условный переход подразумевает выполнение тех или иных команд в зависимости от проверяемого условия, так Бейсик предоставляет нам еще и несколько вариантов действий.

Начнем с классической конструкции:

```
IF...THEN...ELSE
```

Или по-русски:

```
ЕСЛИ...ТОГДА...ИНАЧЕ
```

В записи условия можно использовать следующие символы:

- = (равно);
- > (больше);
- < (меньше);
- <> или >< (не равно);

- <= или =< (меньше или равно);
- >= или => (больше или равно).

Уже из самой конструкции условного оператора что-то должно быть понятно. Поможем раскрыть его смысл на примере следующей программы. Допустим, в некоторой стране живут люди, говорящие только правду. Но при этом и у них есть вредные привычки — некоторые из них курят. Для продажи сигарет сделаны автоматы, которые запрашивают возраст покупателя и реагируют на это двумя различными способами в зависимости от правдиво названной величины.

```
CLS
INPUT "Как Вас зовут"; N$
INPUT "Сколько Вам полных лет"; S
IF S<18 THEN ?"Вам еще рановато курить, ";N$ ELSE ?"Минздрав России пре-
дупреждает — курение опасно для вашего здоровья!"
```

Программа запрашивает имя, потом возраст (все это уже не вызывает у нас никаких затруднений), а вот потом вступает в действие условный оператор. Если возраст `S` меньше 18, то он выводит надпись, что курить еще рановато (а чтобы не было очень обидно, обращается по имени). Иначе (то есть если возраст больше либо равен 18) автомат продает сигареты, но предупреждает об опасности курения.

Еще один маленький пример. Вычислим корень квадратный из `X`.

```
CLS
INPUT "Введите любое число"; X
? "Корень квадратный из X="; SQR(X)
```

Запускаем программу. Вводим число 4. Ответ 2. Вводим 121. Ответ 11. Вводим −9. Аварийное прерывание программы и сообщение об ошибке "Неверный вызов функции". Что такое? А все потому, что корень квадратный вычисляется только из положительного числа. Вот программа и "ругается". Но мы же умные, добавим в программу условный оператор, и будет она работать для любых чисел.

```
CLS
1:INPUT "Введите любое число";X
IF X>=0 THEN ? "Корень квадратный из X ="; SQR (X) ELSE ?"Корень квадрат-
ный из отрицательного числа вычислять
отказываюсь. Дайте положительное число!":GOTO 1
```

Теперь программа анализирует введенное значение и в случае его положительности вычисляет корень, а в случае отрицательности в легкой форме раздражается и просит у пользователя другое число.

И еще пример с графикой. Предположим, мы настолько дружественны к пользователю, что не говорим ему: "Вот, посмотри, какая картинка!", а предлагаем некий выбор. Для первого раза из двух возможностей.

```
SCREEN 12
INPUT "Введите 1, если хотите увидеть солнце и любую другую цифру, если
ничего не хотите"; X
IF X=1 THEN CIRCLE (320, 240), 50, 2: PAINT(320, 240), 14, 2
ELSE ?"Спасибо за внимание!"
```

Программа запрашивает у пользователя цифру (причем в случае ввода буквы просит осуществить ввод заново). Если это цифра 1, то рисуется желтый круг, в любом другом случае появляется надпись `"Спасибо за внимание!"`.

Поупражняемся.

166. Напишите программы, которые в зависимости от введенного числа либо вычисляют функцию, либо выдают сообщение, что функция не определена:

$$y = \frac{1}{x}.$$

$$y = \sqrt{x^2 - 1}.$$

167. Напишите программу для вычисления функции:

$$y = \begin{cases} \dfrac{\cos x}{x}, & x > 0; \\ x \sin x, & x \le 0. \end{cases}$$

168. Напишите программу, определяющую четность или нечетность введенного с клавиатуры целого числа.

169. Напишите программу, находящую меньшее из двух введенных с клавиатуры чисел.

170. По четырехзначному номеру года, запрошенному с клавиатуры, определите номер столетия (например, для 1492 г. — ответ XV век, для 1812 г. — XIX век). Учесть, что началом века считается первый, а не нулевой, год. (То есть 2000-й год — последний год XX века).

Выше мы рассмотрели полную форму условного оператора, когда в зависимости от истинности или ложности условия выполняются те или иные действия, после чего программа переходит к выполнению следующей за условным оператором команды.

Существует также *неполная форма условного оператора*. Она выглядит так

```
IF условие THEN команды,_выполняемые_в_случае_истинности_условия
```

В этом случае программа проверяет условие, и в случае его истинности выполняет команды, указанные после служебного слова THEN. Если же условие ложно, то программа переходит к выполнению следующей за условным оператором команды.

Например, программу автомата по продаже сигарет можно модифицировать при помощи сокращенного варианта условного оператора:

```
CLS
INPUT "Как Вас зовут"; N$
INPUT "А сколько Вам полных лет"; S
IF S<18 THEN ?"Вам еще рановато курить, ";N$ : END
?"Минздрав России предупреждает — курение опасно для вашего здоровья!"
```

Здесь появляется новый оператор END, выполнение которого в любом месте программы приводит к ее остановке.

Возможен еще также составной условный оператор, когда после служебных слов THEN или ELSE вновь следует условный оператор, если выполнение тех или иных действий зависит не от одного, а от нескольких условий.

Например, допустим, что в Преображенский полк гренадеров набирали только из тех новобранцев, рост которых был не менее 180 см, а вес — не менее 80 кг. Программа этого процесса выглядела бы следующим образом:

```
CLS
INPUT "Как Вас зовут?"; N$
INPUT "Каков Ваш рост в см"; R
INPUT "Каков Ваш вес в кг"; M
IF R>=180 THEN IF M>=80 THEN ?"Вы — достойный кандидат
в гренадеры, ";N$ ELSE ? N$; ", к сожалению, вы в гренадеры не годитесь"
```

Здесь условный оператор сначала проверяет соответствие кандидата по росту, а затем еще и по весу, и только в случае выполнения обоих условий выдается сообщение о пригодности кандидата.

При использовании составных условных операторов надо быть очень осторожным, чтобы не запутаться самому в THEN и ELSE. Есть правило, по которому ELSE относится к ближайшему слева от него THEN. В рассматриваемом случае первый условный оператор, который проверяет рост, относится к сокращенной форме, у него нет ELSE, а второй является полным. Отсюда получается такой казус, что если кандидат сразу не проходит по росту, то на экране вовсе не появится никакого сообщения, потому что программа должна перейти к следующей за условным оператором команде, а ее нет. Чтобы разобраться в этом, выполните упражнение. Вообще, лучше не злоупотреблять составными условными операторами, а разбивать их на несколько простых или использовать сложные условия, о которых мы поговорим чуть ниже.

171. Какое значение будет иметь переменная `X` после выполнения очередных операторов:

```
X=0
IF A>0 THEN IF B>0 THEN X=1 ELSE X=2
```

при значениях переменных `A` и `B`:

```
A=1, B=1
A=1, B=-1
A=-1, B=1
```

172. Запишите в виде одного условного оператора указанные действия:

- известно, что из трех чисел `A1`, `A2` и `A3` одно отлично от других, равных между собой. Присвоить номер этого числа переменной `N`;
- $y = \begin{cases} \cos^2 x, & 0 < x < 2; \\ 1 - \sin^2 x. \end{cases}$

173. Напишите программу, запрашивающую у пользователя три разных целых положительных числа и находящую сумму двух наименьших из них.

174. Если `N=3`, то какое значение будет иметь переменная `S` после выполнения следующего оператора:

```
S=1: K=2
1: IF K>N THEN GOTO 2
S=S*K
K=K+1
GOTO 1
2: ? "S="; S
```

Предупреждение

В сокращенной форме условного оператора при переходе к метке можно опускать оператор `GOTO`.

Таким образом, вторая строчка предыдущей программы могла бы выглядеть так:

```
1: IF K > N THEN 2
```

Сложные условия

Если для выполнения тех или иных действий нужно одновременное выполнение двух и более условий, то они объединяются логической связкой `AND`

(И). Например, программу про набор в Преображенский полк можно было бы изменить таким образом:

```
CLS
INPUT "Как Вас зовут?"; N$
INPUT "Каков Ваш рост в см"; R
INPUT "Каков Ваш вес в кг"; M
IF R>=180 AND M>=80 THEN ?"Вы — достойный кандидат
в гренадеры, "; N$ ELSE ? N$; ", к сожалению, вы в гренадеры не годитесь"
```

Если есть несколько условий и для организации тех или иных действий достаточно выполнения хотя бы одного из этих условий, то применяется логическая связка OR (ИЛИ). Например, программа для вычисления следующей функции:

$$y = \begin{cases} x^3, & x < -1; \\ x^2, & -1 \le x \le 1; \\ x^3, & x > 1 \end{cases}$$

выглядит так

```
CLS
INPUT "Введите значение X"; X
IF X<-1 OR X>=1 THEN Y=X ^ 3 ELSE Y=X ^ 2
? "Y (X)="; Y
```

Для различения применения AND и OR можно привести такой жизненный пример. Каждый человек хоть раз задумывался о спутнике жизни и, естественно, выдвигал какие-либо условия к будущему избраннику. Так вот, более требовательные могли бы выдвинуть два таких условия: пусть он будет красивым И (AND) богатым. А менее требовательные, предпочитающие синицу в руках, могли бы сказать так: пусть он будет красивым ИЛИ (OR) богатым. Чувствуете разницу? В первом случае избраннику не будет отказано только в одном случае из четырех возможных, а во втором, наоборот, три исхода благосклонны. Он может быть только красив, или только богат, ну а если и красив, и богат, то тем более.

Еще один пример. В стене существует квадратное отверстие N×N см. И имеется кирпич с измерениями A, B и C. Определить, пройдет он в отверстие или нет, если подавать его можно только параллельно стенкам отверстия.

Понятно, что кирпич пройдет в отверстие только в случае, если хотя бы два его измерения меньше N. Отсюда программа будет выглядеть так:

```
CLS
INPUT "Введите сторону отверстия N"; N
```

```
1: INPUT "Введите стороны кирпича A, B и C"; A, B, C
IF A<N AND B<N OR A<N AND C<N OR B<N AND C<N THEN ?"Кирпич проходит в
отверстие" ELSE ?"Кирпич не проходит в отверстие"
INPUT "Рассмотрим еще один кирпич? 1 - да, 0 — нет"; X
IF X=1 THEN GOTO 1 ELSE ?"Спасибо за работу!"
```

По-русски это условие можно объяснить следующим образом. Если `A` и `B` меньше `N`, или `A` и `C` меньше `N`, или `B` и `C` меньше `N`, тогда печатать `"Кирпич проходит в отверстие"`, иначе печатать `"Кирпич не проходит в отверстие"`. Кроме того, в этой программе применен достаточно простой, но эффективный ход, управляющий совместной работой пользователя и программы. Чтобы каждый раз не запускать программу (если пользователь хочет рассмотреть несколько вариантов исходных данных), применен запрос с клавиатуры вариантов продолжения: либо рассматриваем еще один кирпич, либо — конец программы. Там же вы можете видеть применение безусловного перехода в условном операторе. Разберитесь и пользуйтесь этим для грамотного оформления своих программных продуктов.

Напишите программы для решения следующих задач, используя условный оператор и сложные условия.

175. По введенным с клавиатуры коэффициентам квадратного уравнения `A`, `B` и `C` найдите его корни. Рассмотрите шесть возможных вариантов:

- `A=B=C=0`, корней бесчисленное множество (`X` — любое);
- `A=B=0`, `C≠0`, уравнение не имеет корней;
- `A=0`, `B≠0`, `C≠0`, вырожденное квадратное уравнение, имеется один корень (формулу вычисления корня найдите сами);
- `D<0`, где `D` — дискриминант, который предварительно надо вычислить; уравнение не имеет вещественных корней;
- `D=0`, уравнение имеет два одинаковых корня (вывести их значения);
- `D>0`, уравнение имеет два различных вещественных корня (вычислить и вывести их значения).

Для проверки правильности работы программы предлагается шесть тестовых вариантов исходных данных:

- `A=B=C=0`;
- `A=B=0`, `C=1`;
- `A=0`, `B=3`, `C=6` (должно получиться `X=-2`);
- `A=5`, `B=3`, `C=2`;
- `A=1`, `B=2`, `C=1` (должно получиться `X1=X2=-1`);
- `A=2`, `B=5`, `C=2` (должно получиться `X1=-2`, `X2=-8`).

176. Осуществите запрос трех целых различных чисел с клавиатуры. Выведите на экран наибольшее и наименьшее.

177. Запросите с клавиатуры у тренера олимпийской сборной России по марафону результаты в часах, минутах и секундах трех победителей чемпионата России. Если какие-то два результата различаются менее чем на 5 с, выведите сообщение "Вот так шла борьба за _____ медаль". В сообщении должно быть указано достоинство медали (золотая, серебряная), за которую шла борьба. В противном случае вычислите среднюю скорость победителя в км/час, если принять длину марафонской дистанции 42 км 195 м.

178. А сейчас мы попробуем сделать пока не очень красивый, но очень простой вариант телевизионной игры "О, счастливчик!". Придумайте пять любых вопросов и к каждому из них четыре варианта ответов. Теперь я попробую словесно описать алгоритм, а вы — перевести его на Бейсик. Итак, запрашиваем у игрока имя и узнаём, желает ли он играть. Если не желает, прощаемся, если желает — приветствуем и предлагаем первый вопрос с вариантами ответов. Запрашиваем у игрока с клавиатуры, какой вариант он выбирает. В случае правильного ответа начисляем ему сто очков и переходим ко второму вопросу. Если ответ неверен, то отправляемся на самое начало — снова регистрация и т. д. Первый вопрос — 100 очков, второй — 200, третий — 300, четвертый — 500, пятый — 1000. Если игрок правильно отвечает на все пять вопросов, то поздравляем его и заканчиваем программу.

179. Каждую пятницу члены Клуба толстяков выстраиваются в определенном порядке и взвешиваются. Напишите программу, которая хранит данные взвешивания 10-ти членов Клуба за прошлую неделю. Затем программа запрашивает новые данные взвешивания и для каждого члена Клуба либо выводит поздравление в случае похудения, либо величину прибавки веса с сожалением.

В рамках изучения условного оператора мы можем проанализировать основные принципы движения геометрических объектов по экрану.

Но прежде необходимо рассмотреть программу-сумматор, которая по мере ввода чисел вычисляла бы их сумму, пока мы не захотим прекратить этот процесс. Она выглядит так:

```
CLS
S=0       Rem S — переменная, в которую
          Rem будет заноситься накапливаемая сумма.
          Rem Естественно, пока мы еще не ввели
          Rem ни одного числа, она должна быть обнулена
1: INPUT "Введите любое число"; N
S=S+N
?"Сумма стала равна"; S
```

```
INPUT "Еще число? (1 — да, 0 — нет)"; R
IF R=1 THEN GOTO 1 ELSE ?"Спасибо за работу!"
```

Выполните упражнение.

180. Попробуйте сами написать программу, вычисляющую произведение последовательно вводимых с клавиатуры отличных от нуля чисел. Что изменится по сравнению с предыдущей программой?

Движение геометрических объектов по экрану

Так как интерпретатор Бейсика достаточно медлителен, то очень красивых программ с движением нам не сделать, но, зная принципы, легко потом добиться большего на других языках.

Итак, движение. На самом деле, конечно, ничего там не движется, а создается иллюзия движения за счет того, что сначала рисуется изображение, потом оно стирается (то есть рисуется на том же месте цветом фона), а затем вновь рисуется в новом месте экрана. Компьютер достаточно быстро проделывает это, и нам кажется, что шарик, например, двигается. Начнем мы с точки, которая движется по горизонтали, отражаясь от вертикальных сторон экрана.

```
SCREEN 12
X = 300: Y = 240:    rem(начальные координаты точки)
DX = 1:              rem(приращение координаты X. Так как мы его
                     rem сделали положительным, координата X будет
                     rem увеличиваться, а следовательно, точка
                     rem сначала будет двигаться вправо)
1: PSET (X, Y), 14: rem(вывод точки на экран)
PSET (X, Y), 0:      rem (стирание точки цветом фона)
IF X = 0 OR X = 640 THEN DX = -DX:
                     rem(в случае достижения
                     rem точкой вертикальных сторон экрана приращение
                     rem по X меняется на противоположное, а значит
                     rem точка начнет двигаться в обратную сторону)
X = X + DX:          rem (изменение координаты точки)
GOTO 1:              rem (переход на метку 1,
                     rem где происходит рисование точки).
```

Программа, как вы видите, получилась зацикленной, т. е. точка будет двигаться бесконечно, пока мы не нажмем <Ctrl>+<Break>.

Замечание

Если у вас уже очень хороший компьютер, то точка может двигаться так быстро, что вы не сможете насладиться процессом ее движения. В таком случае мы

искусственно замедлим программу. Для этого между двумя операторами `PSET` надо вставить оператор:

```
FOR I = 1 to 100: NEXT I
```

Это оператор цикла, о котором мы поговорим позже; пока просто поймите, что в этом месте компьютер посчитает до 100 в своем уме, и на это время в программе возникнет пауза. Подставляя вместо числа 100 другие числа, вы сможете управлять величиной паузы.

181. Напишите программу, которая заставит точку двигаться по вертикали. Потом замените точку на закрашенную окружность. Потом — на закрашенный прямоугольник.

Теперь рассмотрим организацию движения точки, которая движется по экрану и отражается от всех его четырех сторон под углом 45°, как это показано на рис. 4.40.

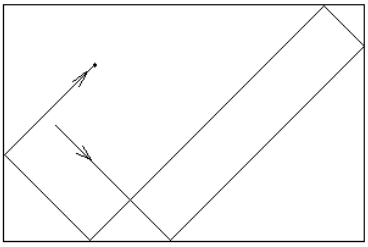


Рис. 4.40. Движение точки по экрану под углом 45°

```
SCREEN 12
X = 300: Y = 240: rem (начальные координаты точки)
DX = 1: DY = 1
                rem (приращение координат X и Y. Так как мы их
                rem сделали положительными, координаты X и Y
                rem будут увеличиваться, а следовательно, точка
                rem сначала будет двигаться вправо и вниз под углом 45°)
1: PSET (X, Y), 14:       rem (вывод точки на экран)
PSET (X, Y), 0            rem(стирание точки цветом фона)
IF X = 0 OR X = 640 THEN DX = -DX
           rem (в случае достижения точкой вертикальных сторон экрана
           rem приращение по X меняется на противоположное, а значит
           rem точка начнет двигаться в обратную по X сторону)
IF Y = 0 OR Y = 480 THEN DY = -DY
           rem (в случае достижения точкой горизонтальных сторон экрана
```

```
            rem приращение по Y меняется на противоположное, а значит
            rem точка начнет двигаться в обратную по Y сторону)
X = X + DX: Y = Y + DY:  rem (изменение координаты точки)
GOTO 1
            rem (переход на метку 1, где происходит рисование точки).
```

Не забывайте про паузу!

182. Замените в предыдущей программе точку на закрашенную окружность. Получится подобие бильярдного стола. А еще можно сопровождать удар шарика о границу экрана звуковым сигналом `BEEP`, который нужно добавить после изменения приращения на противоположное.

183. Усложним задачу. На экране появляется прямоугольник, от которого точка тоже должна отражаться (рис 4.41).

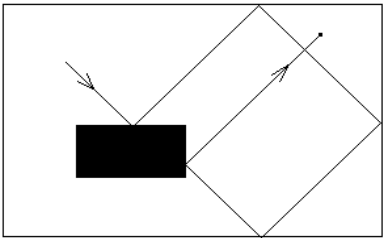


Рис. 4.41. Движение точки по экрану с препятствием

Допустим, координаты углов прямоугольника следующие:

- ❒ левый верхний 100, 180
- ❒ правый верхний 300, 180
- ❒ левый нижний 100, 230
- ❒ правый нижний 300, 230

Тогда, по сравнению с предыдущим примером, к условиям отражения от сторон экрана добавятся еще четыре условия отражения от сторон прямоугольника. Мы приведем два — для верхней и левой сторон, а вы напишите еще два — для нижней и правой.

Для верхней:

```
IF Y = 180 AND X>100 AND X<300 THEN DY = -DY
```

То есть если точка достигла горизонтали 180, а по `X` попала в интервал 100, 300, то изменяем направление движения по `Y` на противоположное.

Для левой:

```
IF X = 100 AND Y>180 AND Y<230 THEN DY = -DY
```

Рассуждения аналогичны.

184. Теперь вы готовы к более объемному и сложному заданию. Называться оно будет "Муха в графине". Сначала на экране вы рисуете прямоугольный графин с горлышком (рис. 4.42).

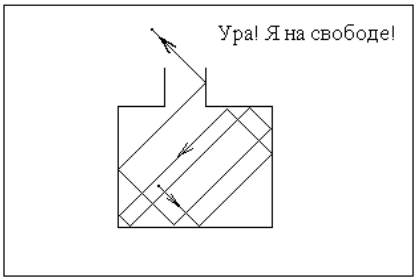


Рис. 4.42. Муха в графине

Затем располагаете там точку (муху) и заставляете ее двигаться внутри графина, отражаясь от его стенок (станет интересней, если вы не будете ее стирать, тогда мы сможем видеть траекторию ее полета). Меняя начальное расположение точки, можно добиться того, что через какое-то время она вылетит из графина. Пусть в этот момент на экране появится надпись "`Ура! Я на свободе!`", а муха продолжит полет, отражаясь от сторон экрана. Графин можно сделать закрашенным.

185. Если вы обратили внимание, муха при полете слегка пробивает стенки графина. Что нужно изменить в условиях, чтобы этого избежать? Сделайте из мухи шарик, и снова добейтесь того, чтобы он не портил стенки графина.

186. Давайте создадим простейший графический редактор. У некоторых в детстве была, наверное, игрушка "Волшебный экран", где можно было, крутя колесики, рисовать отрезками прямых линий. Почему бы то же самое не сделать на компьютере? Для этого нам понадобится использовать оператор `INKEY$`, который ожидает нажатия любой символьной клавиши и запоминает ее значение. Например:

```
CLS
1:K$ = INKEY$
PRINT K$
GOTO 1
```

На экране будет воспроизводиться символ той клавиши, которую вы нажмете (попробуйте поставить после `PRINT K$` точку с запятой и посмотрите, что изменилось.

Теперь, совместно используя INKEY$ и условный оператор, мы можем сделать полезную прикладную программу-развлечение для младших друзей. Мы приведем ее фрагмент.

```
SCREEN 12
X = 300: Y = 200: C = 15
    rem (начальные координаты и цвет точки рисования. Вы можете
    rem задать им свои, произвольные значения)
1: PSET (X ,Y), C
K$ = INKEY$
IF K$ = "w" THEN Y = Y -1
    rem эта строка заставляет программу после нажатия клавиши
    rem "w" уменьшить координату Y на 1, т. е. осуществить
    rem передвижение точки на одну экранную единичку вверх)
```

Здесь вы программируете другие клавиши для движения в иные стороны, а также, возможно, изменение цвета и стирание. Все в ваших руках!

```
GOTO 1
rem (зацикливание программы на постоянный опрос клавиатуры)
```

Теперь, когда программа готова, нарисуйте сами с ее помощью какую-нибудь осмысленную картинку — домик в деревне, или звездные войны, или что хотите.

Замечание

Хорошо бы предусмотреть рисование цветом фона — для перемещения точки без следа. А чтобы потом определить, где находится точка, достаточно изменить ее цвет!

Оператор выбора

Для организации меню, т. е. выбора из нескольких возможностей, применяется *оператор выбора*, который позволяет, в зависимости от значения переменной, делать те или иные действия.

Конструкция оператора несколько тяжеловесна, но ничего страшного в ней нет — в зависимости от значения выражения выполняется один из нескольких блоков операторов.

```
SELECT CASE тест_выражение
  CASE список_выражений1
    [блок_операторов-1]
  [CASE список_выражений2
    [блок_операторов-2]]...
```

```
  [CASE ELSE
    [блок_операторов-n]]
END SELECT
```

Здесь:

- тест_выражение — любое числовое или строковое выражение;
- список_выражений1, список_выражений2 — одно или несколько выражений для сравнения с тест_выражением. В выражении ключевое слово IS должно стоять перед любым знаком отношения;
- блок_операторов-1, блок_операторов-2, блок_операторов-n — один или несколько операторов в одной или нескольких строках.

Аргументы списка выражений могут принимать любую из следующих форм или их комбинацию и должны разделяться запятыми:

- выражение[,выражение]... ;
- выражение TO выражение;
- IS знак_отношения выражение.

Выражение — любое числовое или строковое выражение, совместимое с тест_выражением; знак_отношения — один из следующих знаков отношения: <, <=, >, >=, <> или =.

Пример:

```
CLS
INPUT "Введите уровень риска (1-5): "; RISK
SELECT CASE RISK
CASE IS >= 5
PRINT "Максимальный риск, шансов на возвращение практически нет."
PRINT "Пишите завещание"
CASE 2 TO 4
PRINT "Высокий риск, шансов на возвращение немного."
PRINT "Проверьте снаряжение, оружие и припасы."
CASE 1
PRINT "Риск отсутствует, возврат гарантирован."
PRINT "В Вас отсутствует дух авантюризма, это слишком скучно"
END SELECT
```

Еще пример. По заданному номеру дня недели напечатать его название:

```
CLS
INPUT "Введите номер дня недели"; N
SELECT CASE N
  CASE 1
```

```
    ? "Понедельник"
  CASE 2
  ? "Вторник"
END SELECT
```

Ну, что ж, теория без практики суха. Задания.

187. Написать программу, которая выводит на экран меню, содержащее список трех поэтов под номерами, затем запрашивает у пользователя номер поэта и выводит на экран какое-нибудь его четверостишие, после чего возвращается в меню (последним пунктом меню предусмотреть выход из программы).

188. В восточных календарях принят 60-летний цикл, состоящий, в свою очередь, из пяти 12-летних подциклов. Подциклы обозначались названиями цветов: зеленый, красный, желтый, белый и черный. Внутри каждого подцикла годы носили названия животных: крысы, быка, тигра, кролика (зайца или кота), дракона, змеи, лошади, овцы (барана или козы), обезьяны, петуха, собаки и свиньи. Написать программу с использованием операторов выбора, запрашивающую номер года нашей эры и печатающую его название по восточному календарю. Для проверки: 1966 — год красной лошади, 1984 — зеленой крысы.

189. Усложненный вариант предыдущего задания. Теперь требуется определить не только название введенного с клавиатуры года, но и год наступления "светлого будущего", до которого вы, безусловно, доживете! Делаем это поэтапно. Вот примерный словесный алгоритм. После введения исходного четырехзначного года с клавиатуры вам необходимо выделить составляющие его цифры и записать их в переменные `A1`, `A2`, `A3` и `A4` (Если в номере года есть нули, то в соответствующие переменные записать 4). Затем взять от них синусы по модулю. Из полученных значений взять по две цифры после запятой и сложить их. Если сумма больше либо равна 10, то сложить еще раз. Должно получиться 4 цифры, из которых надо составить четырехзначное число. Это и будет номер года наступления светлого будущего. Осталось определить его название по восточному календарю и вывести результаты на экран. Для проверки — при исходном 1986 годе светлое будущее настанет в 3589 году.

4.3.3. Циклический алгоритм

В тех случаях, когда нам необходимо выполнить одинаковые действия, в которых изменяется только какая-либо величина (так называемый *параметр*), то мы применяем операторы цикла. Это дает мощный толчок программированию, позволяя реализовывать очень сложные алгоритмы.

Оператор с заранее известным числом повторений

Когда мы работали с графикой, то в одном из заданий вам предлагалось построить мишень — пять концентрических окружностей. Мы это легко сделали, пять раз написав оператор построения окружности `CIRCLE`, изменяя в нем каждый раз радиус.

Ну, а если бы таких окружностей надо было 100? Неужели 100 раз писать практически одно и то же? Очень нелегко, даже если пользоваться копированием строк. Неужели нельзя как-нибудь полегче? Конечно можно! Нам на помощь приходит оператор цикла `FOR...NEXT`.

Итак, правила пользования оператором цикла `FOR...NEXT`.

- ❒ Рассмотреть повторяющиеся действия и выделить в них равномерно изменяющуюся величину (параметр).
- ❒ Дать параметру имя.
- ❒ Определить для параметра начальное значение, конечное значение и шаг изменения, т. е. насколько за один раз увеличивается (или уменьшается) параметр.
- ❒ Написать оператор цикла, состоящий из трех частей:
  - заголовок цикла

    FOR <параметр>=<нач.знач.> **TO** <кон.знач.> **STEP** <шаг>

  - тело цикла

    В теле цикла указываются один или несколько операторов, описывающих повторяющиеся действия, причем вместо конкретных значений изменяющейся величины ставят имя параметра.

  - **NEXT** <параметр>

При описании оператора цикла жирным шрифтом были выделены обязательные служебные слова и символы, которые нельзя опускать (кроме специально оговоренных случаев).

Например, построим мишень. Действуем по правилам.

При рисовании мишени изменяется радиус, это и будет параметр.

Обозначим его буквой `R`.

Пусть начальное значение `R=20`, конечное значение `R=60`, шаг изменения Δ`R=10`.

Записываем оператор цикла (предварительно, конечно, включив графический режим):

```
FOR R = 20 TO 60 STEP 10
CIRCLE (320, 175), R, 14
NEXT R
```

Если мы также четко будем и в дальнейшем следовать этим правилам, то нам не составит труда работать с оператором цикла.

Необходимо отметить, что можно записывать оператор цикла в одну строку, это не ошибка:

```
FOR R = 20 TO 60 STEP 10: CIRCLE (320, 175), R, 14: NEXT R
```

Но если в теле цикла много операторов, то такая запись будет плохо читаема.

Рассмотрим работу оператора цикла на примере вышеприведенной программы. Встретив заголовок цикла `FOR`, программа присваивает параметру начальное значение и переходит в тело цикла. Там находит команду `CIRCLE` и исполняет ее, подставляя вместо параметра его начальное значение. Встретив команду `NEXT`, программа добавляет к предыдущему значению параметра величину шага, затем проверяет условие, не стало ли значение параметра больше конечного его значения, и, если нет, продолжает выполнение цикла, а если да, переходит к выполнению следующей за оператором цикла команды или, если таковой не имеется, заканчивает выполнение программы.

Таким образом, последнее значение параметра в нашем примере будет 70, после чего цикл закончится. (Попробуйте к программе примера добавить строку `? "R="; R` и посмотрите, что появилось на экране).

Теперь о составной части оператора цикла — шаге изменения параметра `STEP`. Довольно часто преподаватели провокационно спрашивают, а какие значения может принимать `STEP`? Может ли он быть отрицательным? А дробным? Вот ответы на эти вопросы. Шаг может быть *любым вещественным числом, за исключением нуля*, т. е. и целым, и дробным, и отрицательным. Но отрицательным он может быть только в случае, если начальное значение параметра больше конечного, т. е. когда мы идем от большего к меньшему. Например, тот же оператор цикла про мишень с тем же результатом можно было бы записать и так:

```
FOR R = 60 TO 20 STEP -10: CIRCLE (320, 240), R, 14: NEXT R
```

Более того, если шаг изменения параметра равен единице, то при написании оператора цикла его можно опускать, например:

`FOR W=12 TO 24 STEP 1` можно было бы записать так: `FOR W=12 TO 24`

Но это не касается минус единицы!

Исполнение оператора цикла очень наглядно в графике (хотя графика, как раз, не главное применение этого очень полезного инструмента). Поэтому, давайте освоим правила пользования оператором цикла, выполнив несколько упражнений. Напишите программы, рисующие на экране:

190. Горизонтальную линию, состоящую из точек, расстояние между которыми 8 (рис. 4.43).

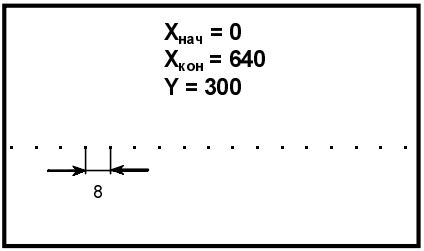


Рис. 4.43. Горизонтальная линия из точек

191. Заполните экран горизонтальными линиями (через 10), а затем с помощью еще одного оператора цикла вертикальными линиями другого цвета (тоже через 10). Должна получиться решетка, как на рис. 4.44.

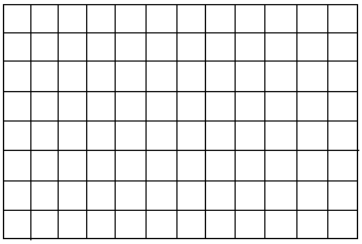


Рис. 4.44. Решетка

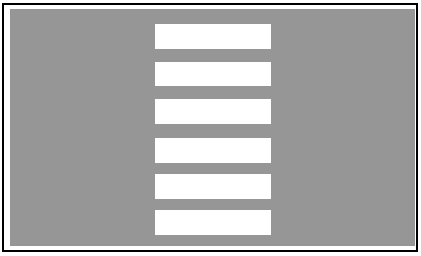


Рис. 4.45. Пешеходный переход "зебра"

192. Постройте пешеходный переход — "зебру", как на рис. 4.45.

193. Нарисуйте пятиконечную звезду, вписанную в окружность, как на рис. 4.33. Используйте для рисования и закраски операторы цикла.

194. Нарисуйте фантастический глаз, как на рис. 4.46.

195. Взрыв сверхновой звезды (если это задание делать по-простому, то понадобится 4 оператора цикла, можно сложнее — но и короче). Сделайте лучи во всех четырех случаях разного цвета. Это придаст картине живости (рис. 4.47).

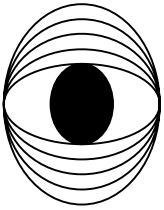


Рис. 4.46. Фантастический глаз

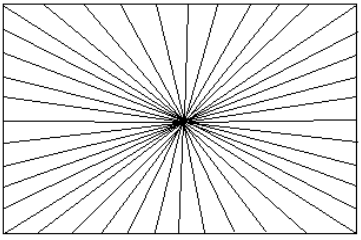


Рис. 4.47. Взрыв сверхновой звезды

196. Заставьте окружность надуваться как воздушный шарик.

197. Мы уже двигали по экрану геометрические фигуры с использованием условного оператора и оператора безусловного перехода. Теперь наши фигуры будут двигаться равномерно по вертикали или горизонтали. Подвигайте:

- закрашенный прямоугольник;
- закрашенную окружность.

Все это замечательно. Но давайте вспомним, что Бейсик все-таки язык в большей степени не графический, а вычислительный, и попробуем применить оператор цикла к вычислениям.

Например, напечатать значения $y = \sin x$ на интервале $[-30°;30°]$ с шагом 5°. Действуя по правилам, несложно понять, что параметр — это `X`, так мы его и обозначим. Начальное, конечное значения параметра и шаг указаны непосредственно в самом задании. Пишем оператор цикла:

```
FOR X=-30 TO 30 STEP 5
Y = SIN (X*3.14/180)
? "SIN("; X; ")="; Y
NEXT X
```

На что, кроме оператора цикла, здесь хочется обратить ваше внимание, так это на оператор `?` (`PRINT`). Разберитесь, пожалуйста, как он работает, и что дает такая, казалось бы, запутанная запись.

Еще пример. Вывести на экран четные числа от 2 до 20. Меняется само число, обозначаем его `N`, начальное значение 2, конечное 20, шаг 2. Пишем:

```
FOR N = 2 TO 20 STEP 2: ? N: NEXT N
```

А если от 20 до 2, то будет, соответственно:

```
FOR N = 20 TO 2 STEP -2: ? N: NEXT N
```

Проверим ваше понимание, как всегда, при помощи ряда заданий.

198. Вывести на экран в строку все числа первой сотни, оканчивающиеся на пять.

199. Определить значение переменной `F` после выполнения следующих операторов:

```
F = 1: N = 1
FOR I = 2 TO N: F = F + 1/I: NEXT I
```

200. Напишите программу, запрашивающую возраст пользователя, а затем печатающую текст "Да ты крут!" по числу прожитых лет. (Обратите внимание, что здесь в теле цикла не будет использоваться параметр. Такое тоже возможно!)

201. С клавиатуры запрашивается любая цифра от 2 до 9, а затем компьютер печатает таблицу умножения на эту цифру.

202. Напишите программу, выводящую на экран степени числа 2 от 2 до 10 включительно.

203. Распечатайте в табличном виде (с аргументами) значение функции квадратного корня на интервале [2; 4] с шагом 0,1.

204. Напишите программу, которая в центре чистого экрана на одном и том же месте выведет последовательно цифры от 1 до 3600 (аналог электронного секундомера).

Когда мы двигали геометрические объекты, то нам приходилось иногда искусственно замедлять движение, чтобы оно было более плавным. Тогда без объяснений был введен так называемый *пустой цикл* для организации пауз. Пустой цикл — такой, в котором нет тела. Например,

```
FOR I = 1 TO 100: NEXT I
```

Просто компьютер считает в уме до 100. А мы его долго и нудно ждем (ну, правда, если до 100 — это быстро, а вот до 10 000 уже придется и подождать!).

Следующая группа упражнений поможет вам закрепить работу с оператором цикла в графике на примере орнаментов. Сначала рассмотрим пример.

Орнамент — это повторяющийся геометрический узор, например, как на рис. 4.48.

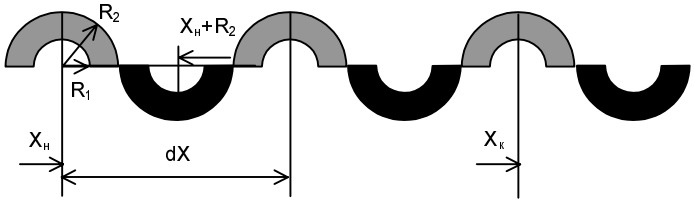


Рис. 4.48. Орнамент

Правила построения орнаментов.

- Выделить в орнаменте повторяющийся фрагмент.
- Выделить опорную изменяющуюся величину (параметр), ее начальное, конечное значения и шаг.
- Дать параметру имя.
- Определить смещение других изменяющихся величин относительно опорной.
- Написать оператор цикла.

205. Напишите программу для рисования такого орнамента (см. рис. 4.48).

206. Закрепите успех, написав программу для построения греческого орнамента (рис. 4.49).

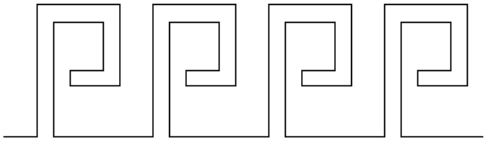


Рис. 4.49. Греческий орнамент

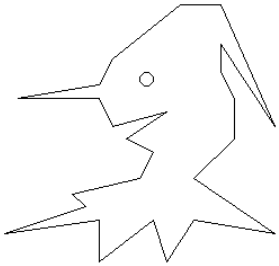


Рис. 4.50. Буратино

207. Когда мы изучали операторы `DATA` и `READ`, то говорили, что используем их при изучении оператора цикла. Этот час пробил. Задав в операторе `DATA` координаты всех точек концов отрезков, из которых состоит рис. 4.50, напишите программу, рисующую Буратино при помощи оператора цикла.

Совет

При рисовании непрерывной ломаной линии применяется сокращенная форма оператора `LINE`, которая выглядит так:

```
LINE — (X, Y), C
```

То есть линия будет рисоваться от последней графической точки на экране до точки с координатами `X`, `Y`. Поэтому, при выполнении такого задания рекомендуется поставить первую точку оператором `PSET`, а затем в цикле считывать поочередно координаты следующих точек и рисовать отрезки при помощи сокращенной формы оператора `LINE`.

Оператор цикла *WHILE...WEND*

Этот оператор позволяет сочетать свойства условного оператора и оператора цикла. С его помощью можно выполнять повторяющиеся действия с заранее неизвестным количеством повторений.

Работает он следующим образом. Сначала идет оператор `WHILE`, в котором описываются условия, при которых цикл выполняется. Ключевое слово `WEND` аналогично по своему назначению слову `NEXT`, т. е. является последней строкой цикла. Всякий раз, доходя до `WEND`, компьютер проверяет, выполняются ли условия, описанные оператором `WHILE`. Если они не выполняются, то программа переходит к исполнению операторов, следующих за `WEND`. Если же условия выполняются, то цикл повторяется снова. Следующая програм-

ма, прежде чем допустить пользователя до продолжения работы, запрашивает у него пароль.

```
WHILE WORD$<> "TERMINATOR"
INPUT "ВВЕДИТЕ ПАРОЛЬ"; WORD$
WEND
? "ПРАВИЛЬНО! ВЫ ДОПУЩЕНЫ К ПРОДОЛЖЕНИЮ РАБОТЫ"
```

Вот как эта программа будет работать после запуска:

```
ВВЕДИТЕ ПАРОЛЬ? MOON
ВВЕДИТЕ ПАРОЛЬ? ARNOLD
ВВЕДИТЕ ПАРОЛЬ? TERMINATOR
ПРАВИЛЬНО! ВЫ ДОПУЩЕНЫ К ПРОДОЛЖЕНИЮ РАБОТЫ
```

208. При помощи оператора `WHILE...WEND` вычислите `C` — наибольший общий делитель введенных с клавиатуры натуральных чисел `X` и `Y`.

209. Определите, какие из нижеследующих операторов Бейсика написаны правильно, а какие нет. В случаях неправильной записи объясните допущенные ошибки. Все операторы рассматривайте по отдельности, потому что они не являются частями одной программы.

а) `FOR D=R TO S STEP H`

б) `IF W=13 THEN ?"ЧЕРТОВА ДЮЖИНА"`

в) `IF W=13 THEN X=X+1`

г) `FOR S=2 — 6`

д) `FOR S=4 TO 12 STEP 2`

е) `FOR S=12 TO 4 STEP 2`

ж) `WEND S=45`

Что будет выведено на экран в результате выполнения следующих программ при указанных исходных данных.

210. Программа:

```
INPUT W
FOR R = 1 TO W
? "WAR"
?
NEXT R
IF R = W THEN GOTO 1
? "SUNDAY"
1: "PEACE"
```

при `W = 3`

211. Программа:

```
W = 27.6
1: IF INT(W) = W THEN GOTO 2
? "ЧЕРТ ПОБЕРИ!!!"
W = W * 2 — 5.2
GOTO 1
2: ? "МИХАИЛ СВЕТЛОВ"
A = W -1
IF A <> 50 THEN 3
GOTO 1
3: END
```

212. Программа:

```
FOR H = 5 TO 21 STEP 4
? H — 3
? H + 3
NEXT H
```

213. Программа:

```
1: ? "НА СТАРТ! ВНИМАНИЕ!"
FOR V = 5 TO 1 STEP -1
? V; "СЕКУНД"
GOTO 1
? "МАРШ!"
```

214. Программа:

```
DATA 10, 100, 1000
READ A, B
? B — A
READ C
RESTORE
READ X, Y, Z
IF C = Y THEN 1
IF B = Y THEN 2
GOTO 3
1: ? "ПЛЮС"
GOTO 3
2: ? "МИНУС"
3: END
```

215. Программа:

```
X = 13
Y = 17
Z = 2
2: IF Z >=5 THEN 1
? X, Y, Z
X = X — 1
Y = X + Y
Z = Z + 1
GOTO 2
1: END
```

216. Программа:

```
FOR R = 1 TO 49 STEP 6
W = W + R*2
NEXT R
? R
```

217. Программа:

```
FOR K = 7 TO 28 STEP 7
? "УРА!!!"
NEXT K
? "ВПЕРЕД!!!"
```

218. Программа:

```
X = 1
1: ? X
X = X + 2
IF X <= 13 THEN 1
END
```

219. Программа:

```
FOR B = 1 TO 4
D = B * 2
? D, B
NEXT B
```

220. Программа:

```
FOR C = 2 TO 11 STEP 2
? C, C^2
NEXT C
? "ВЗРЫВ!"
```

221. Программа:

```
X = 13
Y = 52
Z = 99
FOR U = 100 TO 1 STEP -2
IF U = X THEN ? U
IF U = Y THEN ? U
IF U = Z THEN ? U
NEXT U
```

222. Программа:

```
WHILE S < 10
? S
WEND
```

223. Программа:

```
WHILE D$ <> "ИНФОРМАТИКА"
READ D$, B
? "ПРЕДМЕТ"; D$
? B
WEND
DATA МАТЕМАТИКА, 4, ФИЗИКА, 3, ИНФОРМАТИКА, 5, ХИМИЯ, 3
```

Случайные числа

В процессе изучения циклических алгоритмов уместно рассказать о получении случайных чисел программным путем. При кажущейся простоте получить случайные числа не так легко. Попробуйте надиктовать своему другу 100 цифр. В идеале, по теории вероятности, каждая из них должна повториться около 10 раз, но у людей встречаются какие-то пристрастия к тем или иным цифрам, и в ряду из 100 цифр не будет абсолютной случайности. Получить на компьютере случайные числа тоже было нелегко, но в конце концов это удалось. В качестве основы для получения какой-либо последовательности случайных чисел служат показания встроенного в компьютер таймера. А т. к. они в каждый момент времени различны, то мы и получаем великое множество практически неповторяющихся случайных чисел в заданном диапазоне.

Случайные числа нужны прежде всего в компьютерных играх, чтобы обеспечить непредсказуемость игры для человека. Кроме того, они используются в учебных задачах и моделировании различных математических, физических и других процессов на компьютере.

Итак, новый оператор `RND`.

Чтобы получать случайные числа, прежде всего следует инициировать процесс генерации последовательности случайных чисел компьютером при помощи оператора `RANDOMIZE TIMER`. Он пишется в программе только один раз и указывает компьютеру, что надо получать случайные числа, беря за основу показания таймера.

Затем, чтобы получить из этой последовательности какое-либо число, мы должны использовать оператор `RND(N)`, который выдает случайное число в диапазоне от 0 до 1. То есть минимальное полученное число будет 0,0000001, а максимальное 0,9999999. В качестве `N` можно использовать любое целое или действительное число. Этот параметр также влияет на выбор компьютером случайных чисел. Например, программа

```
RANDOMIZE TIMER
X = RND(1)
? X
```

может вывести на экран число 0,367423, а может 0,0034289, а может 0,8912314. Каждый раз при запуске программы мы будем видеть новое число.

Но часто встает проблема получения случайных чисел в заданном диапазоне. Как это сделать, мы рассмотри на двух примерах.

Пример 1. Моделирование бросания монеты

Нам необходимо при каждом запуске программы случайным образом получать либо число 0 ("решка"), либо 1 ("орел").

Если просто брать целую часть от выданного оператором `RND(N)` случайного числа, то всегда будем получать 0.

Попробуем умножить `RND(N)` на 2. Тогда начнут выдаваться числа от 0,0000002 до 1,9999998. Если мы теперь будем брать целую часть, то как раз и получим 0 или 1. Программа будет выглядеть так:

```
CLS
RANDOMIZE TIMER
X = INT(RND(1)*2)
IF X = 0 THEN ? "У вас выпала решка" ELSE ? "У вас выпал орел"
```

Пример 2. Моделирование бросания игрального кубика

Казалось бы надо пойти по уже проторенному пути первого примера и, раз нам надо получить цифры от 1 до 6, просто умножить `RND(N)` на 6. Но, если разобраться повнимательней, то обнаружится, что при этом мы будем получать числа в диапазоне от 0,0000006 до 5,9999994. Взятая целая часть даст нам числа от 0 до 5. На 6 никак не выходим. Хорошо, умножим `RND(N)` на 7. Тогда получим числа от 0,0000007 до 6,9999993. Теперь получается 6, но нам

ведь совсем не нужен 0. Что же делать? Делаем просто — умножаем на 6 и прибавляем 1!

```
CLS
RANDOMIZE TIMER
X = INT ( RND (1) * 6) + 1
? X
```

Так как в дальнейшем нам придется часто сталкиваться с заданиями с использованием случайных чисел, то необходимо потренироваться.

Напишите операторы для получения случайных целых чисел в интервалах:

224. От 1 до 10.

225. От −5 до +5.

226. От 10 до 20.

227. От 50 до 100.

228. От −35 до 65.

Применим теперь оператор случайных чисел на практике, и, как всегда, для наглядности, начнем с графики.

Программа "Звездное небо"

Изобразим звездное небо пятьюстами белыми точками на черном фоне. Координаты каждой из точек должны быть в пределах по `X` от 0 до 640 и по `Y` от 0 до 480.

```
SCREEN 12
FOR N =1 TO 500
X = INT (RND(1) * 641)
Y = INT (RND(1) * 481)
PSET (X, Y), 15
NEXT N
```

Здесь число 500 в заголовке цикла указывает на количество "звезд". Можете поэкспериментировать, изменяя это число.

А теперь мы хотим получить разноцветные "звезды"!

Добавим для этого перед `PSET` оператор случайных чисел для цвета

```
C = INT (RND(1) * 16)
```

а в самом операторе `PSET` вместо цвета 15 укажем цвет `C`, и можем наслаждаться разноцветными "звездами".

Но давайте уж доведем дело до конца. Ведь при использовании оператора `C = INT (RND(1) * 16)` у нас будет получаться и 0 — черный цвет, т. е. свое-

образные "черные дыры" на черном фоне. Давайте избежим и этого, заставив компьютер при получении 0 не брать его в расчет, пока не получится другой цвет:

```
SCREEN 12
FOR N =1 TO 500
X = INT (RND(1) * 641)
Y = INT (RND(1) * 481)
1: C = INT (RND(1) * 16)
IF C = 0 THEN 1
PSET (X, Y), C
NEXT N
```

Со звездным небом разобрались. Теперь задания. Напишите программы, изображающие следующее.

229. Когда-то на петербургском 5-ом канале была скандально известная информационная программа "600 секунд" с телеведущим Александром Невзоровым, который всячески эксплуатировал криминальные сюжеты для роста рейтинга своей передачи, причем делал это очень профессионально. Так вот, начиналась она с заставки, где экран телевизора покрывался сеткой разноцветных отрезков прямых линий со случайными координатами концов (рис. 4.51). Попробуйте воспроизвести эту заставку и покажите ее своим старшим родственникам или друзьям. Это навеет на них соответствующие воспоминания.

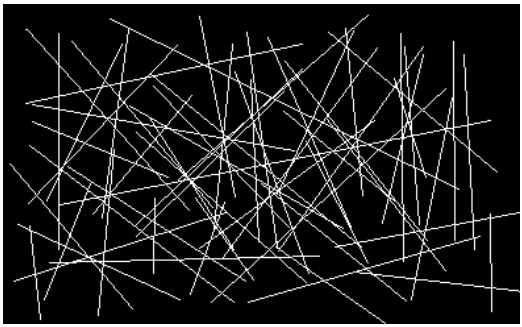


Рис. 4.51. "600 секунд"

230. Видоизменим (весьма, надо сказать незначительно, легким движением руки) предыдущую программу, чтобы она заполняла экран сотней цветных прямоугольников случайных размеров (рис. 4.52). Постарайтесь, чтобы на черном фоне не получались черные прямоугольники.

Рис. 4.52. Разноцветные прямоугольники

231. А теперь настал черед заполнить экран полусотней закрашенных окружностей. Закраска пусть будет разноцветная, а вот сами окружности только белые (рис. 4.53).

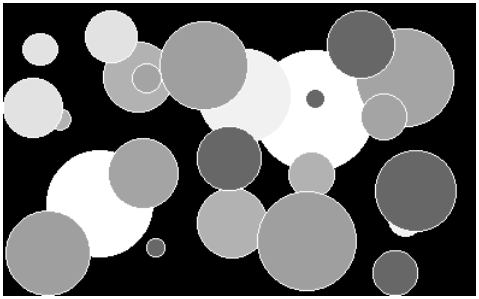


Рис. 4.53. Разноцветные окружности

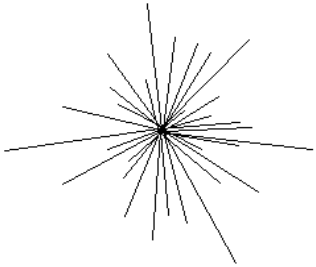


Рис. 4.54. Взрыв

232. А теперь давайте устроим в центре экрана маленький взрыв. Он будет изображен сотней отрезков разноцветных прямых, сходящихся в точке $X = 320$, $Y = 175$. А длина их пусть лежит случайным образом в пределах от 50 до 100 экранных точек (рис. 4.54).

Построение графиков алгебраических функций

Строить графики в Бейсике мы будем точками. А задавать компьютеру функцию для построения оператором `DEF FN`.

Кроме того, координатная сетка экрана компьютера, как мы должны помнить, является перевернутой по отношению к привычной геометрической. Отсюда и последуют преобразования по приведению координатной сетки к нормальному виду. Мы не хотим утомлять читателя длинными математическими выкладками, а призываем поверить на слово. Если же в ком-то проснется любознательность, то разобраться во всем можно и самому.

Итак, оператор `DEF FN` определяет компьютеру ту функцию, график которой мы хотим построить. Работает он следующим образом:

```
DEF FN [имя функции] (параметр функции) = [функция]
```

Например, определим функцию $y = \sin x$

```
DEF FNY(X) = SIN(X)
```

Здесь `Y` — имя функции, `X` — ее параметр, а `SIN (X)` — непосредственно сама функция.

Определим функцию $y = \cos 2x + \mathrm{tg}\, x/2$

```
DEF FNY(X) = COS (2*X) + TAN (X/2)
```

Предупреждение

В одной и той же программе нельзя использовать для разных функций одно и то же имя. Необходимо давать разные имена, например `Y(X)`, `Z(X)`, `R(X)` и т. д.

Теперь непосредственно о построении графиков на примере функции синуса.

Сначала нарисуем оси координат — две взаимно перпендикулярные линии, пересекающиеся в центре экрана.

```
SCREEN 12
LINE (0, 240) – (640, 240), 15:  rem ось абсцисс
LINE (320, 0) – (320, 480), 15:  rem ось ординат
```

Теперь определим функцию, как мы это уже делали.

```
DEF FNY(X) = SIN(X)
```

А теперь самое интересное — загадочный цикл непосредственного построения графика.

```
FOR X = 0 TO 640
Y = 240 – 30*FNY*((X – 320)/30)
PSET (X, Y), 15
NEXT X
```

После запуска программы должна получиться следующая картинка (рис. 4.55).

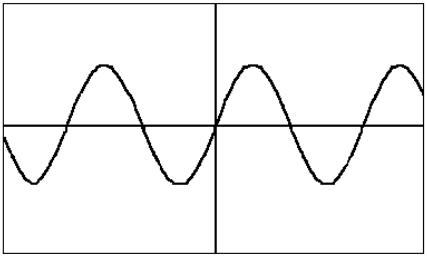


Рис. 4.55. График функции синуса

Значение `X` изменяется от 0 до 640, давая возможность строить график на всю ширину экрана. Достаточно сложно выражение для `Y`. Это связано с переносом начала координат в центр экрана (отсюда числа 240 и 320). А вот число 30 в данном случае несколько произвольно — это коэффициент растяжения. Попробуйте его поменять сначала на 10, а потом на 50 (только менять всегда надо в двух местах) и посмотрите на возникающие отличия. На наш взгляд, коэффициент 30 наиболее оптимален.

Построение любого другого графика сводится к замене определения функции на другую.

Давайте поупражняемся. Напишите программы, строящие графики следующих функций.

233. `Y=COS(X)`

234. `Y=ABS(X)`

235. `Y=INT(X)`

236. `Y=1/(1+X^2)`

237. `Y=X^2`

238. `Y=2*SIN(X/2)+0.5*COS(2*X)`

239. `Y=TAN(X)`

Графики представляют собой семейства точек, между которыми могут быть значительные разрывы (как, например, при построении графика тангенса). Чтобы избежать этого и уплотнить график, можно в заголовке цикла установить шаг изменения по `X` меньше 1. Попробуйте.

Теперь попытаемся построить график функции $y = 1/x$ (кажется, это называется гиперболой). Если строить его как описано выше, ничем не дополняя, то возникнет ошибка, о которой мы уже говорили — "Деление на ноль". Придется в очередной раз вспомнить математику и область допустимых значений. В данном случае, `X` не может быть равен 0, но т. к. начало координат у нас искусственно перенесено в центр экрана, то для графика `X` будет обращаться в 0 при координате экрана, равной 320. Поэтому программа будет выглядеть так:

```
SCREEN 12
LINE (0, 240) - (640, 240), 15: rem ось абсцисс
LINE (320, 0) - (320, 480), 15: rem ось ординат
DEF FNY(X) = 1/X
FOR X = 0 TO 640
IF X = 320 THEN X = X + 1
Y = 240 - 30*FNY*((X - 320)/30)
PSET (X, Y), 15
NEXT X
```

А график должен выглядеть так (рис. 4.56).

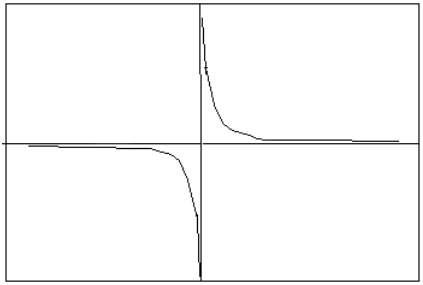


Рис. 4.56. График гиперболы

Постройте теперь графики следующих функций.

240. `Y=1/(X+1)`

241. `Y=SQR(X)`

242. `Y=X/(X^2-1)`

А теперь задания посложнее, но и покрасивее. Изобразите на экране поверхность, образованную вращением вокруг оси `X` графиков следующих функций:

243. `Y=1/(1+X^2)`

244. `Y=1/X`

245. А теперь поверхность, образованную вращением вокруг оси `Y` графика функции $y = x^2$.

Совет

Попробуйте строить график не точками, а эллипсами, радиус которых... Вот именно!

Циклы с несколькими зависимыми параметрами

Бывает, что при попытке выполнить задание с оператором цикла выясняется, что изменяется не одна, а две или три величины. В том случае, если вы можете установить между ними зависимости, следует применить следующие правила:

- определить, какие же величины изменяются, и обозначить их буквами;
- назначить одну из этих величин независимым аргументом;
- определить закономерности, связывающие изменяющиеся величины, и выразить эти величины через независимый аргумент;

- написать программу с оператором цикла, основным параметром которого будет независимый аргумент, а в командах тела цикла, на месте других изменяющихся величин, необходимо подставить их выражения через независимый аргумент.

Приведем пример. Пусть мы хотим построить семейство окружностей, как на рис. 4.57. Правда, напоминает рупор?

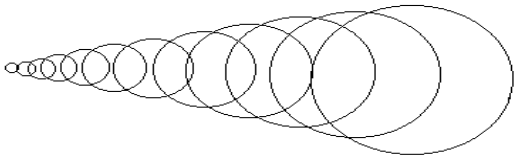


Рис. 4.57. "Рупор"

Действуем по правилам:

- Все окружности лежат на одной горизонтали, т. е. у их центров меняются не координаты `Y`, а координаты `X`. Кроме этого у них меняется радиус. Причем не спонтанно, а плавно увеличиваясь. Итак, обозначим изменяющиеся величины: `X` — координата центров окружностей по оси абсцисс, и `R` — радиус окружностей.
- Назначим независимым аргументом координату `X`.
- Попробуем установить закономерность изменения радиуса `R` от аргумента `X`. `X` изменяется от 0 до 640. Допустим, шаг изменения 10. Обычно устанавливают такое взаимоотношение `X` и `R`:

  ```
  R = X
  ```

  Но в этом случае получится, что радиус последней окружности будет равен 640, диаметр, соответственно, 1280. А ведь экран высотой всего 480! Значит, если мы ходим видеть на экране все наши окружности, то последний радиус не должен превышать 240, а лучше был бы меньше. Попробуем избрать такое соотношение:

  ```
  R = X/4
  ```

- Максимальное значение радиуса в таком случае равно 160.

  Напишем программу:

  ```
  SCREEN 12
  FOR X = 0 TO 640 STEP 10
  CIRCLE (X, 240), X/4, 15
  NEXT X
  ```

Посмотрите, что получается на экране. Похоже? Не совсем, скажете вы и будете правы, потому что последние окружности выходят за пределы экрана справа. Но с этой проблемой мы предлагаем разобраться вам самостоятельно.

Потренируемся. Напишите программы, изображающие следующие картинки:

246. "Орбиты", соприкасающиеся все в крайней правой точке и имеющие разные радиусы.

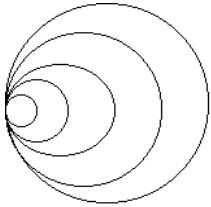


Рис. 4.58. "Орбиты"

247. "Труба" — окружности одного радиуса, выходящие из левого верхнего угла экрана и идущие под углом 45° вправо вниз. Окружности нарисованы одним цветом, а закрашены другим (рис. 4.59).

248. Такая же "труба", только окружности идут из левого верхнего угла в правый нижний по диагонали экрана (рис. 4.60). Здесь надо использовать соотношения сторон экрана.

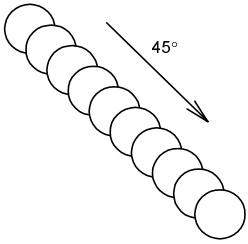


Рис. 4.59. "Труба" под углом 45°

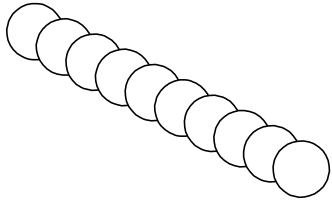


Рис. 4.60. "Труба" по диагонали

249. А теперь две лестницы (рис. 4.61 и 4.62).

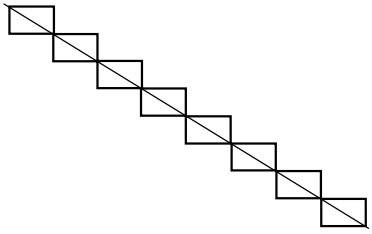


Рис. 4.61. Лестница 1

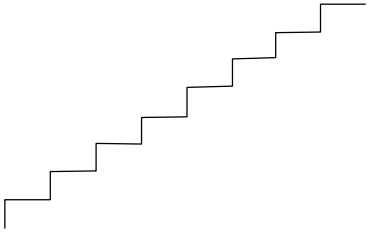


Рис. 4.62. Лестница 2

250. Далее две пирамиды индейцев майя — вид спереди (рис. 4.63) и вид сверху (рис. 4.64).

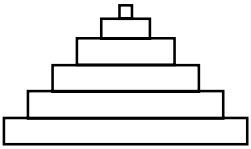


Рис. 4.63. Пирамида: вид спереди

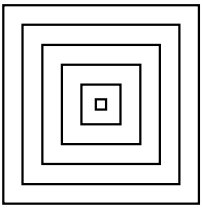


Рис. 4.64. Пирамида: вид сверху

251. Вернитесь к примеру с "рупором" из увеличивающихся окружностей и попробуйте сделать картинку, на которой подобные рупоры выходят из всех четырех углов экрана и встречаются в центре.

252. Напишите программу построения "эллипсирующих окружностей" (рис. 4.65).

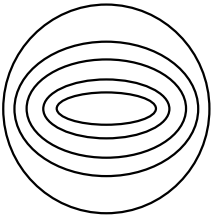


Рис. 4.65. "Эллипсирующие окружности"

Вложенные циклы

В предыдущей теме мы рассмотрели случаи, когда в цикле изменяются несколько величин, но их можно выразить друг через друга.

Сейчас нам надо рассмотреть случаи, когда изменяются несколько независимых величин.

Приведем пример. Напишем программу заполнения экрана точками с расстоянием между ними по горизонтали и вертикали, равным 10.

```
SCREEN 12
FOR X = 0 TO 640 STEP 10
FOR Y = 0 TO 480 STEP 10
PSET (X, Y), 4
NEXT Y
NEXT X
```

Работает программа следующим образом. В первом заголовке цикла переменной `X` присваивается значение 0. Потом программа переходит к выполнению следующего оператора, т. е., в нашем случае, второго цикла, который и исполняется, пока `Y` не превысит значение 350. После этого берется следующее значение `X`, и история с `Y` продолжается. И так до тех пор, пока `X` не станет больше 640. То есть программа строит вертикальные цепочки точек, пока не заполнится весь экран.

Все это не так сложно, если понять и запомнить следующее. Циклы должны быть вложены друг в друга как матрешки. Параметры, упоминаемые в заголовках циклов, в их окончаниях должны быть перечислены наоборот.

Правильно:

```
FOR X = ....
  FOR Y = ....
    Тело цикла...
  NEXT Y
NEXT X
```

Неправильно:

```
FOR X = ....
  FOR Y = ....
    Тело цикла...
  NEXT X
NEXT Y
```

Кроме того, в окончании вложенных циклов можно указывать только одно общее ключевое слово `NEXT` с перечислением параметров, например

```
NEXT Y, X
```

Упражнения. Напишите программы, заполняющие экран следующими изображениями.

253. Лоскутный ковер. Экран заполняется плотно прилегающими друг к другу разноцветными квадратиками со стороной 10 (рис. 4.66).

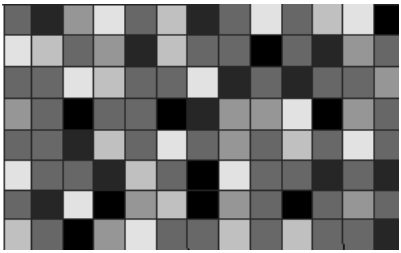


Рис. 4.66. Лоскутный ковер

254. Светофорные круги (рис. 4.67). Самые большие круги красные, средние — желтые, маленькие — зеленые.

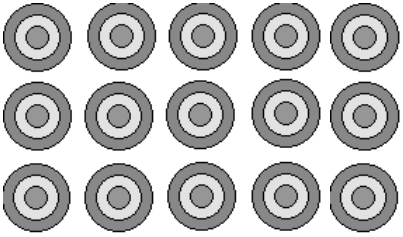


Рис. 4.67. Светофорные круги

255. А теперь в предыдущем задании замените окружности на квадраты — получатся "светофорные" квадраты.

256. Изобразите шахматную доску.

Мы все работаем с графикой. Это наглядно. Но не будем забывать о том слове, от которого произошло само название компьютера — вычислять.

257. Напишите программу, выводящую на экран таблицу умножения от 2 до 10 в следующем виде (рис. 4.68).

	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

Рис. 4.68. Таблица умножения

Прежде чем задать следующее задание, хочется ознакомить вас с весьма полезным оператором `SLEEP`. Он приостанавливает выполнение программы на указанное после него количество секунд. Например:

```
? "Поспите 10 сек..."
SLEEP 10
? "Подъем! Учитель пришел!"
```

Если значение секунд указано 0 или не указано вовсе, то программа будет ожидать нажатия любой клавиши, например:

```
? "Прочитайте внимательно нижеследующие слова и нажмите любую клавишу,
когда надоест..."
? "Вы — просто гений!"
SLEEP
? "Но от скромности вы не умрете..."
```

258. А вот теперь, используя полученные знания про вложенные циклы и оператор `SLEEP`, напишите программу, которая в центре чистого экрана будет выводить показания хронометра — часы, минуты и секунды, разделенные двоеточием, начиная с 0 час. 0 мин. 0 с. Отрегулируйте свой хронометр, чтобы он шел правильно.

259. Усложним предыдущее задание. Нарисуем будильник, сделаем в нем два окошка. В одно выведем сегодняшнюю дату, а во второе — наш уже получившийся хронометр. Если очень захочется, то можно сделать из него будильник, добавив в нужный момент оператор `BEEP`, который производит отрывистый и не совсем приятный звук (рис. 4.69).

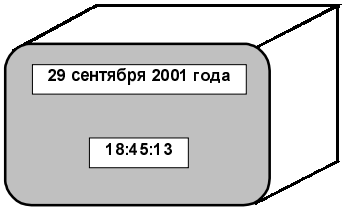


Рис. 4.69. Электронный будильник

А теперь два интегрированных задания, где придется применить многие знания, приобретенные вами ранее.

260. Программа "Уничтожение астероида". Экран заполняется пятьюстами звездами. Астероид изображается красным кругом с белым контуром, с центром в точке (470, 160) и радиусом 10. По траектории функции $y = 0{,}5\sin 2x + 2\cos x/2$ от правой стороны экрана движется боевой звездолет, изображаемый зеленым кругом с желтым контуром и радиусом 3. Каждый момент продвижения сопровождается проверкой условия, не находится ли звездолет в опасной близости от астероида (опасным считается расстояние менее 50 экранных точек). Если это расстояние в какой-то момент времени станет меньше критического, то звездолет уничтожает астероид, т. е. он взрывается (см. задание 232). На экране появляется торжествующая надпись. Все радуются (в том числе и учитель).

Подсказка для тех, кто не знает

Расстояние между двумя точками на плоскости вычисляется по формуле: корень квадратный из суммы квадратов разностей координат $X1$, $X2$ и $Y1$, $Y2$. Мы специально не пишем эту формулу. Проверяем понимание сложных инструкций.

261. Это задание носит прикладной характер и позволяет опытным путем вычислить число π. Да, безусловно, практически все из вас знают это число с точностью по крайней мере до двух знаков. Но уж больно метод хорош. Называется он метод Монте-Карло. Монте-Карло — европейская столица игорного бизнеса, а значит там владычествует Его Величество Случай. Вот мы и попробуем поставить его себе на службу.

Сначала забудьте, чему равно π и послушайте теорию вопроса. Представим себе окружность радиусом $R = 1$, вписанную в квадрат. Из этого следует, что сторона квадрата будет $2R$, а его площадь $SK = (2R)^2 = 4R^2$. А площадь круга $SO = \pi R^2$. Далее равномерно посыпаем квадрат песком. Затем нанимаем бригаду рабочих, которые считают, сколько песчинок всего ($N1$) и сколько из них попало в круг ($N2$). Затем составляется простая пропорция — площадь квадрата так относится к площади круга, как общее количество песчинок к количеству песчинок, попавших в круг. ×

$$\frac{SK}{SO} = \frac{N1}{N2} \Rightarrow \frac{4 \times R^2}{\pi \times R^2} = \frac{N1}{N2} \Rightarrow \pi = \frac{4 \times N2}{N1}.$$

Отсюда сенсационный вывод — радиус окружности не имеет никакого значения, она должна быть лишь вписана в квадрат (рис. 4.70).

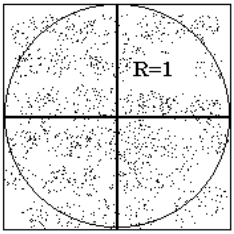


Рис. 4.70. Вычисление числа π методом Монте-Карло

Но где ж мы найдем песок, а главное тех, кто все это будет считать? Поэтому ставим компьютерный эксперимент. Рисуем квадрат со вписанной в него окружностью. Координаты опорных точек (если сами рисовали) знаем. Уравнение окружности $X^2 + Y^2 = R^2$ тоже знаем. Задаем цикл до 1000, в котором случайным образом определяем координаты "песчинок" так, чтобы они лежали внутри квадрата. Тут же проверяем условие, а не попала ли "песчинка" в круг (используя уравнение окружности), и если попала, подсчитываем их количество, а кроме того, рисуем их на экране разными цветами (попавшие и не попавшие). По окончании цикла подсчитываем и вы-

водим на экран число π. Понятно, что чем больше количество "песчинок", тем более точным будет результат. Для того чтобы знать, когда закончится эксперимент, рекомендуется выводить на экран счетчик "песчинок" (как мы делали с хронометром). Но, все-таки, экспериментировать с миллионом "песчинок" не надо — замучаетесь ждать сами, да и компьютер, хотя и железный, но все же живой.

Наращивание переменной

О том, что такое переменная и об операторе присваивания мы уже говорили. И обращали внимание на абсурдное с точки зрения математики выражение

```
X = X + 1
```

Применяется оно обычно в Бейсике для работы всевозможных счетчиков, а также для подсчета сумм или произведений рядов чисел.

Например, вычислить сумму всех четных чисел от 1 до 100 включительно. Программа будет выглядеть так:

```
CLS
S = 0   'Обнуление переменной, где будет накапливаться сумма
FOR I = 2 TO 100 STEP 2
S = S + I
NEXT I
? "Сумма четных чисел от 1 до 100 равна"; S
```

Программа работает эффективно и просто. Параметром цикла являются сами четные числа, которые нам остается накапливать в переменной `S`. Разберем алгоритм по шагам:

1. `S=0`;
2. `I=2;`
3. `S=0+2=2;`
4. `I=4;`
5. `S=2+4=6;`
6. `I=6`
7. `S=6+6` и так далее до 100.

На экране в результате видим надпись:

```
Сумма четных чисел от 1 до 100 равна 2550.
```

Объяснять здесь вроде больше нечего. Иногда лучше решать, чем говорить. Напишите программы для нахождения сумм:

262. Всех чисел, делящихся на 13 в интервале [1;1000]

263. $1^2 + 2^2 + 3^2 + 4^2 + 10^2$.

264. $\frac{1}{\sqrt{1}}+\frac{2}{\sqrt{2}}+\frac{3}{\sqrt{3}}+...+\frac{10}{\sqrt{10}}$.

265. Написать программу для нахождения суммы пяти произвольных чисел, вводимых с клавиатуры (сумматор).

266. Написать программу для нахождения суммы пяти заданных чисел, хранящихся в `DATA`.

Написать программы для вычисления произведений.

267. $\cos 5° \times \cos 10° \times \cos 15° \times \ \times \cos 85°$.

268. $\frac{1}{1+a^2}+\frac{1}{1+(a+1)^2}+...+\frac{1}{1+b^2}$.

269. Написать программу, рисующую цепочку соприкасающихся окружностей с заданными в `DATA` радиусами (рис. 4.71).

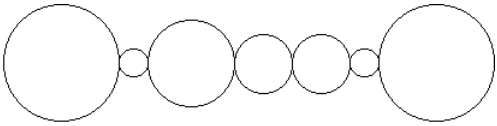


Рис. 4.71. Цепочка соприкасающихся окружностей

270. Написать программу вычисления среднего роста десяти человек, данные о которых занесены в `DATA`.

271. Написать программу вычисления среднего балла при поступлении в институт по результатам четырех экзаменов, которые вводятся с клавиатуры.

272. Написать программу вычисления среднего геометрического трех произвольных чисел. (Среднее геометрическое есть корень степени N из произведения N чисел).

Оператор *DO...LOOP*

Оператор `DO...LOOP` чем-то похож на оператор `WHILE...WEND`, но возможности его применения несколько шире — проверяемое условие может быть использовано не только в начале оператора, но и в конце.

Поэтому возможны четыре различных формы написания оператора:

- первая форма

```
DO UNTIL условие
    тело_цикла
LOOP
```

- вторая форма

```
DO WHILE условие
    тело_цикла
LOOP
```

- третья форма

```
DO
    тело_цикла
LOOP UNTIL условие
```

- четвертая форма

```
DO
    тело_цикла
LOOP WHILE условие
```

Если используется ключевое слово `UNTIL`, то цикл будет выполняться только, когда условие имеет значение "ложь".

Если используется ключевое слово `WHILE`, то цикл будет выполняться до тех пор, пока условие будет иметь значение "истина".

А размещение условий в начале или конце цикла чревато следующим: при проверке условия в конце оператора цикла операторы, составляющие его тело, будут исполняться хотя бы один раз, в то время как при проверке условия в начале возможен вариант, что цикл исполняться вообще не будет.

Пример:

```
K = 0
? "Значение K в начале цикла равно"; K
DO WHILE K < 10
K = K + 1
LOOP
? "Значение K в конце цикла равно"; K
```

А кстати, чему же будет равно значение `K` в конце цикла?

273. Напишите программы — одну с оператором `DO...LOOP WHILE`, а другую с `DO...LOOP UNTIL`, которые бы выводили на экран расположенные по диагонали экрана круги, закрашенные цветами с 1 по 6. (Параметр цикла — номер цвета, рис. 4.72).

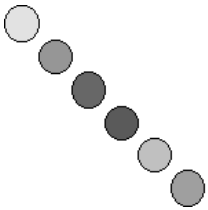


Рис. 4.72. Разноцветные круги

4.3.4. Символы и строки

Когда мы говорили о дружественном интерфейсе, то упоминали о так называемых строковых переменных, имена которых отличаются от обычных добавлением знака `$`. В таких переменных могут содержаться как отдельные символы, так и их последовательности длиной до 255 символов. К ним в Бейсике применимы специальные операции, на использование которых и направлены следующие задания.

Используя функции `ASC`, `CHR$`, `INPUT$`, `LEN`, `LEFT$`, `RIGHT$`, `MID$` и `INSTR$`, выполните следующие задания.

274. Напишите программу, определяющую, что за символы скрываются под кодами 33, 66, 99, 100, 128.

275. Примените функции `ASC` и `CHR$` к примеру простейшей шифровки информации, когда символы вводятся побуквенно, а программа определяет их код, добавляет к ним 1 и выводит на экран вместо введенного символа символ с новым получившимся кодом. Слово для тестовой проверки такой программы — `CAT`, после его побуквенного введения должно получиться `DBU`.

276. Напишите программу-дешифратор для предыдущего задания. Тестовая проверка: из слова `DBU` должно получиться слово `CAT`.

Эти программы грамотно работают для первых стандартизированных 128 кодов. Чтобы правильно работать, например, с русским текстом, надо знать коды строчных и прописных букв кириллицы, которые скрываются в интервале от 129 до 255. Поэтому еще одно задание.

277. Напишите программу, выводящую на экран символы, скрывающиеся за кодами 129—255. Распечатайте или выпишите коды строчных и прописных букв кириллицы.

278. Определите с помощью функции `LEN`, сколько символов будет в следующем адресе:

197110, Россия, Санкт-Петербург, Чкаловский пр., 78—33

279. Определите, какие слова получатся в результате выполнения приведенной ниже программы?

```
CLS
F$="ГАЗОНОКОСИЛЬЩИК"
W1$=MID$(F$, 4, 2)+RIGHT$(F$, 7)
W2$=MID$(F$, 4, 2)+LEFT$(F$, 2)
W3$=MID$(F$, 9, 1)+MID$(F$, 7, 2)+MID$(F$, 11, 2)+MID$(F$,7,2)
? W1$
? W2$
? W3$
```

280. Напишите программу, которая выдаст на экран пять слов максимальной длины из слова "ЭЛЕКТРИЧЕСТВО". Побеждает тот, у кого сумма букв во всех словах наибольшая.

В качестве следующего примера приведем задачу подсчета слов во введенном тексте. Как известно, для компьютера словом является последовательность символов, заключенная в пробелы с двух сторон. Подчеркиваю, это не обязательно слово в привычном для нас понимании, а любой набор символов, например, 45po9) или ВАР56+УР47. Поэтому, в простейшем случае, подсчет количества слов во введенном тексте сводится к подсчету количества пробелов и добавлению к полученному значению единицы. (Почему так? Очень просто: слов два, а пробел между ними один; слов три — а пробелов два и т. д.) Получаем программу.

```
CLS
INPUT "Введите текст телеграммы"; W$
N=LEN(W$) : K=0
FOR I=1 TO N
  P$=MID$(W$, I, 1)
  IF P$="  " THEN K=K+1
NEXT I
? "В вашей телеграмме — "; K+1; "слов"
```

Программа работает очень просто. Она определяет длину текста в символах и заносит это число в переменную `N`. Затем устанавливает счетчику пробелов `K` нулевое значение. После чего в цикле вырезает из текста последовательно по одному символу и проверяет, не является ли он пробелом. Если это так, то увеличивает счетчик пробелов `K` на единицу, а если нет — берет следующий символ. По завершении цикла в переменной `K` хранится количество пробелов в тексте, и мы выводим ответ о количестве слов на экран, добавляя к `K` еще единичку.

281. Используя пример с подсчетом слов в телеграмме, напишите программу, имитирующую отделение связи с очень хорошим обслуживанием. Программа должна выяснять имя клиента и в дальнейшем обращаться к нему только по имени. Запрашивается также регион, куда посылается телеграмма. Их три — Россия (коэффициент 1), страны СНГ (стоимость одного слова умножается на 2) и дальнее зарубежье (стоимость одного слова умножается на 5). По России стоимость одного слова составляет 1 руб. 50 коп. (причем неважно, какой длины слово). Затем у клиента запрашивается текст телеграммы и денежная сумма, определяется количество слов, стоимость телеграммы. Если денег ровно столько, сколько надо, его благодарят и прощаются. Если больше, чем надо, то ему предлагают сдачу и прощаются. Если — меньше, то просят добавить необходимую сумму, а затем, после расчета, с клиентом прощаются. А для пущей красоты рекомендуется нарисовать окошко те-

леграфа, в прорезях которого и происходит диалог компьютера с пользователем (рис. 4.73).

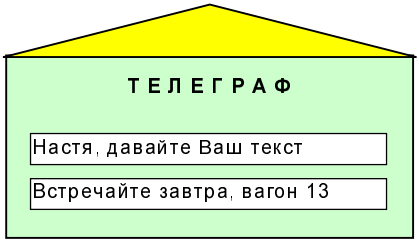


Рис. 4.73. Телеграф

282. Напишите программу, проверяющую, является ли введенное слово или фраза палиндромом, т. е. читающимся слева направо и справа налево одинаково (например, шалаш, казак, А роза упала на лапу Азора). Программа сообщает "Да, это палиндром" или "Нет, это не палиндром" и выводит на экран введенный текст в варианте слева направо и справа налево. Здесь необходим цикл посимвольного чтения от N (длины текста) до 1.

283. Напишите программу, подсчитывающую количество слогов во введенном слове.

284. В операторе `DATA` перечислены 10 слов. Напишите программу, которая:

- напечатает все слова из списка, отличные от слова SUN;
- напечатает слово, ближайшее к началу алфавита в списке (считаем, что все слова различны; выполнять аналогично поиску минимального из ряда чисел);
- слово, составленное из последних символов всех слов списка;
- все слова из списка, содержащие три буквы.

285. В тексте, содержащем между словами от 1 до 3 пробелов, оставьте только по одному.

286. Подсчитайте, сколько раз входит каждый символ в данную строку.

287. Напишите программу шифратор-дешифратор, ставящую в соответствие русским символам соответствующие латинские и наоборот (аналог так называемого конвертора).

288. Напишите программу, в которой вводится целое десятичное число, которое затем должно быть представлено в римской записи (например, 24 — XXIV).

289. Замените в исходном тексте "photo, graph, philophon, cophe" все сочетания "ph" на символ "f".

4.3.5. Массивы

Мы подошли к одной из самых сложных, на наш взгляд, для начинающих тем в программировании. И раскрыть ее постараемся как можно более доходчиво, пусть не совсем научными терминами, но понятно, поскольку без понимания массивов дорога в программирование закрыта.

Но и без умных определений не обойтись.

Итак, *массив* — это набор однородных данных (чисел, символов, слов), имеющий имя и последовательную нумерацию его элементов. Например, список фамилий учеников вашего класса — массив, численные данные о среднесуточной температуре за месяц — массив, буквы русского алфавита — массив.

Как физически массив представляется в компьютере, мы рассматривать не будем, а вот как формально — это надо себе представлять.

Если мы знаем, что программе предстоит работать с большим объемом каких-то данных (то есть с массивом), то мы должны этот массив в программе описать с помощью специального оператора `DIM` (от английского слова *dimension*), после которого идет имя массива, а потом в скобках следует так называемая размерность массива, т. е. количество элементов, его составляющих. Например, в группе 4 человека, массив — это фамилии учеников. Мы тогда должны сделать так:

```
DIM FAM$ (4)
```

В этом случае компьютер в памяти отводит некую область из 4 ячеек, которую всю и называет `FAM$` (знак `$` добавляется, т. к. в массиве будут строковые переменные). Кроме того, эти ячейки нумеруются натуральными числами, начиная с 1. Подобную процедуру, да и сам массив, можно сравнить с одноэтажной улицей в деревне или маленьком городке. Построили на улице 4 дома, назвали улицу `FAM$` (имя массива дается по тем же правилам, что и имя переменной), пронумеровали дома и заселили туда жильцов (рис. 4.74).

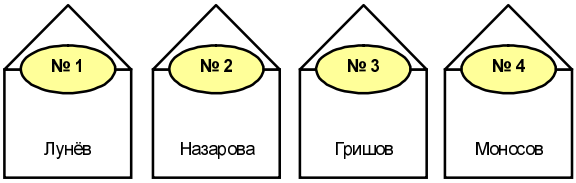


Рис. 4.74. Массив `FAM$` из четырех элементов

Из этого следует, что:

- у массива есть *имя*, которое дает программист;
- у массива есть *тип*, который определяется именем — числовой (имя без `$`) и текстовый (символьный, строковый) — имя с `$`;

- у массива есть *размерность*, т. е. количество составляющих его элементов;
- у массива есть *сквозная последовательная индексация* составляющих его элементов;
- у каждого элемента массива есть *значение* (в нашем случае это фамилия).

290. Заполните массив десятью случайными целыми числами, каждое из которых лежит в пределах от 50 до 200, и выведите на экран их численные значения, а также их графическое представление в виде вертикальных закрашенных прямоугольников шириной 30 и высотой, соответствующей их значению. Нижние стороны прямоугольников лежат на линии с координатой $Y = 300$, левой стороне первого прямоугольника соответствует координата $X = 100$. (рис. 4.75)

291. Заполните массив десятью случайными целыми числами, каждое их которых лежит в пределах от 5 до 30, и выведите на экран их численные значения, а также их графическое представление в виде закрашенных соприкасающихся кругов, радиусы которых равны значениям элементов массива (рис. 4.76).

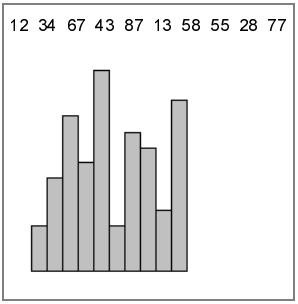


Рис. 4.75. Столбиковая интерпретация одномерного массива

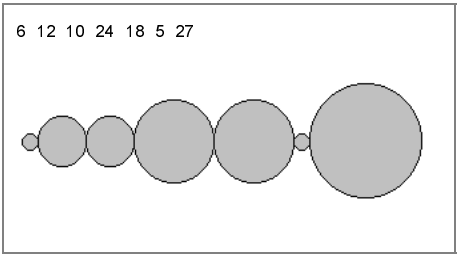


Рис. 4.76. Круговая интерпретация одномерного массива

292. Напишите программу вычисления среднего арифметического содержащихся в операторе `DATA` следующих 10 чисел: 31, 19, 52, 65, 6, 8, 13, 16, 97, 33.

293. Напишите программу вычисления следующего выражения:

$$\frac{(x_1 - S)^2 + (x_2 - S)^2 + \ldots + (x_n - S)^2}{N},$$

где x_i — числа из оператора `DATA` из предыдущего задания; N — их количество; S — среднее арифметическое элементов массива x.

294. Найдите сумму 1-го, 4-го, 9-го, 16-го, ..., 81-го элементов массива, состоящего из 100 целых случайных чисел, каждое из которых лежит в пределах от 2 до 22.

295. Замените в массиве из 10 случайных целых чисел, каждое из которых лежит в пределах от 1 до 10, все четные элементы нулями и выведите полученный массив на экран.

296. Массив состоит из 60 случайных двузначных целых чисел. Выведите их на экран в обратном порядке по 6 чисел в строке.

297. В массиве содержатся 10 букв — С, Ф, О, И, К, Л, О, И, Л, Н. Выведите на экран слово, образованное буквами с четными индексами, и слово, образованное буквами с нечетными индексами.

298. Массив состоит из 20 целых положительных и отрицательных чисел, каждое из которых находится в пределах от 2 до 12. Вывести на экран сначала отрицательные, а затем положительные числа. Определите, модуль суммы каких чисел больше — положительных или отрицательных.

299. Найдите максимальный, минимальный элементы массива из 10 случайных целых двузначных чисел и разность между ними. Представьте графическую столбиковую интерпретацию этого массива, выделив максимальный элемент красным, а минимальный — зеленым цветом. Остальные прямоугольники должны быть желтого цвета.

Если в программе используется несколько массивов, то они могут быть описаны через запятую, например:

```
DIM X(10), Y(10), W$(20)
```

Задания с несколькими массивами.

300. Даны два массива, заполненные каждый десятью случайными целыми числами, каждое из которых от 1 до 9 включительно. Сложить массивы поэлементно, результаты записать в третий массив. На экран вывести все три массива.

301. Даны три массива одинаковой размерности 5, в которых содержатся стороны треугольников A_i, B_i и C_i. Определите периметр P_i и площадь S_i каждого треугольника по формуле Герона. (Так как для того, чтобы треугольник существовал, необходимо определенное соотношение его сторон, то лучше элементы массивов задать в операторе `DATA`).

302. Даны два массив A и B одинаковой размерности 5, которые содержат случайные числа от 2 до 9 включительно. Найдите скалярное произведение этих массивов, вычисляемое по формуле:

$$P = A_1 \times B_1 + A_2 \times B_2 + \ldots + A_N \times B_N,$$

где N — размерность массива.

303. Найдите соотношение SX/SY, где SX и SY — средние арифметические значения массивов X и Y соответственно. (Массивы размерностью 10 содержат случайные двузначные целые числа).

304. Найдите объем каждого из 10 цилиндров, для которых заданы радиусы оснований R_i (случайные целые числа от 5 до 25 см) и высоты H_i (случайные целые числа от 10 до 30 см).

305. Заданы 10 пар координат X_i, Y_i одних точек на плоскости и 10 пар координат A_i, B_i других точек на плоскости. Вычислите попарно расстояния между точками по формуле:

$$S_i = \sqrt{(X_i - A_i)^2 + (Y_i - B_i)^2}.$$

306. Занесите эти расстояния в массив S. Проиллюстрируйте задачу графически (соответствующими отрезками на экране). Выделите разными цветами наибольшую и наименьшую длину.

307. Дан одномерный массив W из 10 случайных целых чисел, каждое из которых лежит в пределах от 1 до 100. Получите новый массив R, где каждый элемент получается из массива W делением соответствующего элемента на его индекс.

308. Вычислите и представьте в виде массива последовательность первых 20 чисел Фибоначчи, если $X_1 = 1$, $X_2 = 1$, а каждый последующий элемент получается суммой двух предыдущих.

309. В операторе `DATA` содержатся данные о среднесуточной температуре в течение февраля 2000 года по Санкт-Петербургу. Вычислите среднюю температуру февраля, наибольшую и наименьшую температуры. Постройте линейный график изменения среднесуточной температуры.

310. Школьники неохотно носят одежду, отличающуюся по цвету от одежды одноклассников. Напишите программу, выбирающую 2 цвета (для мальчиков и для девочек) из 12 возможных цветов, которые сегодня будут носить все. Выбор производится случайным образом и сопровождается выводом на экран прямоугольников соответствующих цветов, внутри одного из которых написано "Сегодня этот цвет для мальчиков", а внутри другого "Сегодня этот цвет для девочек".

311. В фирме "Green Beavis Ltd" работают семь сборщиков компьютеров. Для того чтобы повысить производительность их труда, в компьютер в конце недели заносят количество компьютеров, собранных каждым из них ежедневно. Напишите программу, которая выдает на экран следующие данные:

- наибольшее количество компьютеров, собранных одним сборщиком за неделю;
- среднее количество собранных за день компьютеров;
- лучший результат за один день;
- номер сборщика, показавшего этот результат, и день, в который он был достигнут.

312. Дан массив X, состоящий из 100 целых случайных чисел, каждое из которых лежит в пределах от 3 до 13. С клавиатуры вводится целое число N, также лежащее в этих пределах. Определите количество элементов массива, равных числу N.

313. Даны два числовых массива X и Y размерностью 10 и 20 соответственно. Получите массив Z размерностью 30, составленный добавлением массива X в конец массива Y.

314. Дан массив из 20 случайных целых чисел, каждое из которых лежит в пределах от 10 до 50. Найдите максимальное число и его номер, а если таких чисел несколько, то подсчитайте, сколько их.

315. Дан массив из 20 случайных целых чисел, каждое из которых лежит в пределах от 10 до 50. Определите среднее арифметическое элементов массива, а также значение элемента, ближайшего к среднему, и его номер.

316. Дан массив среднемесячных температур за год. Определите, в каком месяце была самая высокая температура, а в каком самая низкая, а также среднесезонные температуры.

317. У автора этой книги была копилка, в которой было 100 советских монеток достоинством в 1, 2, 3, 5, 10, 15, 20 и 50 копеек. Задайте массив $M(100)$ случайным образом из этого набора, а затем подсчитайте, сколько в копилке было пятачков и полтинников, и какова общая сумма накопленного.

318. Дан массив из десяти целых двузначных случайных чисел. Найдите сумму трех максимальных из них.

319. Заполните массив $R(25)$ случайными целыми двузначными числами так, чтобы числа не повторялись.

320. С клавиатуры вводится дата в формате День/Месяц/Год (День и Месяц представлены двумя цифрами, а Год четырьмя), а программа представляет в том же формате дату следующего дня, например:

```
Введите дату 31/12/2001
Результат 01/01/2002
```

Результат работы программы надо проверять при помощи тестов на окончание месяцев, високосный/невисокосный годы, программа должна указывать на недопустимые даты и т. д.

321. С клавиатуры вводится дата в формате День/Месяц (каждая часть даты представлена двумя цифрами), а программа переводит дату в словесное представление, например:

```
Введите дату 13/10
Результат "Тринадцатое октября"
```

322. С клавиатуры вводится значение некоей денежной суммы в рублях и копейках. Необходимо вывести на экран ее словесное представление, например:

```
Введите количество рублей — 18
Введите количество копеек — 55
Результат "Восемнадцать рублей пятьдесят пять копеек"
```

323. Программа "Пожиратель звезд". Когда мы говорили о циклах и построении графиков функций, то писали программу о движении "звездолета", который, пролетая по траектории заданной тригонометрической функции, уничтожал астероид, находящийся от него в опасной близости. Теперь, со знанием массивов, мы можем написать более красивую и более сложную программу. Сначала заполним экран тысячью разноцветных звезд, изображенных либо точками, либо окружностями радиусом 1. Координаты всех звезд запоминаются в массивах $X(1000)$ и $Y(1000)$. Затем по траектории $y = 2\sin x/2 + 1/2\cos 2x$ начинает двигаться кавалькада: впереди сторожевой звездолет (закрашенный круг радиусом 2), проверяющий, не находится ли какая-нибудь звезда в опасной близости от армады ($S \leq 30$), и уничтожающий ее в этом случае (звезда заменяется точно такой же, но цветом фона). Сзади движутся три крейсерских звездолета (закрашенные круги радиусом 4). В результате выполнения программы на экране должен быть проложен коридор по траектории функции y шириной 60 экранных точек (рис. 4.77).

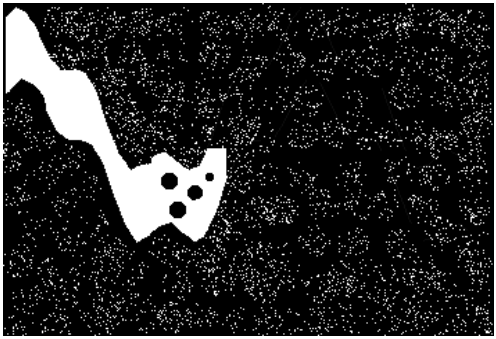


Рис. 4.77. "Пожиратель звезд"

324. Напишите программу "Сторож", которая заставила бы змейку (рис. 4.78) оббегать стороны экрана по часовой стрелке (это можно написать и без массива). Усложните программу, взяв на службу еще одну "змейку". Теперь они ползают друг за другом (рис. 4.79). Еще сложнее — программа для движения змейки, управляемой стрелками или буквенными клавишами и совершающей повороты под прямым углом.

А если и это вам по плечу — заставьте змейку поворачивать еще и под углом 45°.

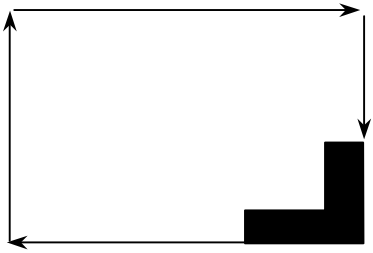


Рис. 4.78. Сторож-змейка

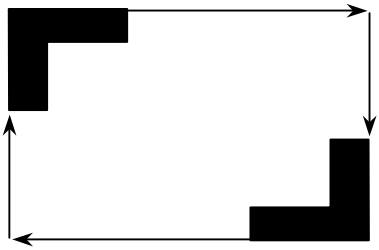


Рис. 4.79. Две змейки

Простейшие сортировки

Одной из основных операций, производимых над массивами, являются операции *сортировки* или упорядочивания элементов массива по какому-либо признаку — чаще по возрастанию или убыванию для чисел, и по алфавиту — для символов и строк.

Сортировок придумано множество, и, говорят, тому, кто придумает новый эффективный метод сортировки, сразу будет вручена Нобелевская премия.

С древних времен, однако, до нас дошли два самых простых (но, конечно, не самых эффективных) способа сортировки, которые мы здесь и рассмотрим.

Сортировка выбором

Допустим, дан числовой массив из N элементов:

Надо отсортировать его по возрастанию.

Суть способа в следующем. Находим наибольший элемент в массиве и меняем его местами с последним. Уменьшаем количество рассматриваемых элементов на 1 (так как последний элемент уже на своем месте). Повторяем операцию для уменьшенного на единицу массива. И так $N - 1$ раз.

Пусть дан массив из пяти элементов:

```
8    4    9    6    7
```

Рассмотрим процесс упорядочивания по шагам.

```
  8    4    7    6    9
❒

  6    4    7    8   (9)
❒

❒ 6    4    7   (8)  (9)
```

На этом шаге третий элемент поменялся сам с собой.

❒ **4** **6** (7) (8) (9)

325. Напишите программу для упорядочивания массива по возрастанию методом выбора. Измените программу так, чтобы она упорядочивала массив по убыванию.

326. Дан массив из 13 чисел. Расположите числа по возрастанию. Введите с клавиатуры число *M* так, чтобы оно вошло в массив, и получившийся массив также был бы упорядочен по возрастанию.

Метод обмена или "пузырька"

Название этого метода часто вызывает нездоровый смех у молодежи, хотя никакого тайного смысла здесь нет. Просто сортировка выполняется таким образом, что максимальное число после каждого шага сортировки как бы всплывает в конце массива, на своем заслуженном месте. Заключается метод в следующем. Программа, начиная с первых элементов массивов, сравнивает эти элементы попарно, и, в случае, если они расположены не по возрастанию, меняет их местами. В результате $(N-1)^2$ перестановок (в случае самого плохого расположения элементов массива — все элементы по убыванию) массив окажется упорядочен по возрастанию. Например, есть массив из 4 элементов:

4 3 2 1

Рассмотрим по шагам метод "пузырька" для этого массива.

❒ 3 **4** 2 1

❒ 3 2 **4** 1

❒ **3** 2 1 **4**

❒ 2 **3** 1 4

❒ **2** 1 **3** 4

❒ 1 **2** 3 4

327. Напишите программу, реализующую метод "пузырька". Для уменьшения количества ненужных сравнений введите счетчик, подсчитывающий количество обменов за один полный пробег вдоль всего массива. Как только его значение станет равно нулю (то есть ни одного нового обмена не произведено), это будет означать, что массив упорядочен. Для наглядности оформите исходный и получившийся массив в виде столбиковых интерпретаций друг под другом.

328. Дан массив букв, составляющих английский алфавит, но стоящих не по порядку. Напишите программу, преобразующую этот массив в алфавит английского языка.

329. Дан массив из 13 четырехбуквенных русских слов, существительных, нарицательных, в единственном числе, в именительном падеже. Упорядочите их по алфавиту.

Задания на обработку двумерных массивов.

330. Найдите в двумерном числовом массиве 4×4 максимальный и минимальный элемент, и, при выводе массива на экран, выделите их красным цветом.

331. Дан двумерный массив 5×5. Определите сумму элементов каждой строки и ту строку, в которой сумма элементов максимальна.

332. Дан массив *A*(2, 10). В первом столбце содержатся координаты *X* точек плоскости экрана, а во втором столбце — координаты *Y* тех же точек. Определите количество точек, попадающих в нижнюю правую четверть экрана, выведите их на экран, а искомые точки выделите другим цветом.

333. Определите наименьший элемент в массиве *X*(10, 10). Выделите его другим цветом.

334. Дан массив *W$*(5, 4), в котором каждая строка состоит из четырех символов, составляющих английское слово. Отсортируйте массив таким образом, чтобы слова были расположены по алфавиту.

335. В массиве *R*(5×5) поменяйте местами первую и последнюю строки.

336. В массиве *R*(5×5) замените элементы, стоящие ниже главной диагонали, нулями.

337. В массиве *R*(5×5) замените элементы главной диагонали нулями.

338. В массиве *R*(5×5) вычислите сумму элементов главной диагонали.

339. В массиве *R*(5×5) упорядочьте строки по возрастанию элементов главной диагонали.

340. Определите, является ли заданный массив 3×3 магическим квадратом, т. е. таким, суммы элементов которого в строках, столбцах и главных диагоналях равны между собой.

341. Выведите на экран номера строк массива 5×5, сумма элементов которых четна.

342. Выведите на экран изображение Андреевского флага (рис. 4.80), если у данного массива 5×5 суммы элементов диагоналей равны, и флаг Японии — в ином случае (рис. 4.81).

343. Одномерный массив из *N* элементов сверните по спирали в квадратную матрицу размерностью корень квадратный из *N* по образцу, приведенному в задании 344.

344. Исходный массив *S*1(16) состоит из следующих элементов:

3, 5, 9, 7, 12, 34, 21, 13. 6, 89, 54, 66, 2, 10, 99, 55.

Создайте массив *S*2(4, 4), вид которого представлен на рис. 4.82.

4	6	8	3	**2**
6	**9**	5	**3**	4
6	0	**1**	8	7
5	**8**	2	**3**	5
7	6	0	3	**4**

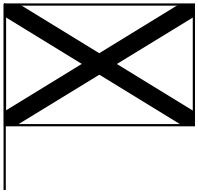


Рис. 4.80. Андреевский флаг (суммы элементов диагоналей равны)

4	6	8	3	**2**
6	**8**	5	**3**	4
6	0	**1**	8	7
5	**8**	2	**3**	5
7	6	0	3	**4**

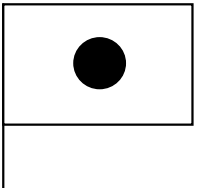


Рис. 4.81. Флаг Японии (суммы элементов диагоналей не равны)

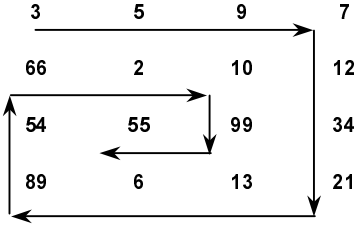


Рис. 4.82. Преобразованный в спираль одномерный массив

4.3.6. Подпрограммы

Иногда в разных местах программы приходится выполнять практически одни и те же последовательности действий с разными исходными данными. Такие последовательности действий можно оформить в виде так называемых *подпрограмм* (от англ. *subroutine*). Подпрограммы сокращают текст программы, существенно уменьшают время их исполнения, облегчают жизнь программистам, которые могут создавать программы модульно, т. е. собирая сложную программу из законченных кусочков более простых составляющих. Это позволяет группе программистов создавать большие программы, а группе школьников разрабатывать и реализовывать какие-либо глобальные проекты.

Для примера использования процедуры приведем программу вычисления следующего выражения:

$$Z = \frac{N!}{M!(N-M)!}.$$

Переменные N и M, а также выражение $(N - M)$ снабжены восклицательным знаком, который в математике означает не что иное как *факториал* (например, N-факториал). Факториал — это произведение натуральных чисел от 1 до N включительно. Например, $3! = 1\times2\times3 = 6$, а, например, $6!$ — это уже $1\times2\times\times3\times4\times5\times6 = 720$.

В выражении придется три раза вычислять факториал, производя при этом одни и те же действия с разными числами. Нам облегчит работу подпрограмма, которую мы разместим в конце основной программы и назовем, например, FACT. В подпрограмме мы опишем, как вычислить факториал вообще, а затем из основной программы обратимся к подпрограмме три раза с различными параметрами. Вот как будет выглядеть программа:

```
CLS
1 : INPUT "Введите значения N и M"; N, M
K = N
GOSUB FACT
X1 = P
K = M
GOSUB FACT
X2 = P
K = (N — M)
GOSUB FACT
Z = X1 / (X2 * P)
PRINT "Искомое выражение равно"; Z
PRINT "Еще одно выражение?(Y/N)": INPUT a$
IF UCASE$(a$) = "Y" THEN 1 ELSE PRINT "До новых встреч у монитора..."
END

' Описание подпрограммы FACT для вычисления факториала
FACT:
P = 1
FOR i = 1 TO K
P = P * i
NEXT i
RETURN
```

Рассмотрим работу программы. Начало стандартное — очистка экрана, запрос с клавиатуры исходных данных. Самое интересное начинается потом. Подпрограмма FACT описана после слова END основной программы, что обусловлено простым желанием, чтобы программа все же не зациклилась, а когда-нибудь-таки кончилась. Параметром подпрограммы является переменная K, для которой и вычисляется факториал.

В основной программе мы присваиваем переменной K значение переменной N, для которой требуется вычислить первый факториал из трех. Затем необходимо обратиться к подпрограмме. Для этого существует специальное слово GOSUB, после которого указывается имя подпрограммы (в нашем случае FACT). Управление переходит на первый оператор, содержащийся в подпрограмме, после чего выполняются все операторы подпрограммы, вплоть до ключевого слова RETURN. Это означает, что подпрограмма закончилась, и управление передается в основную программу.

Предупреждение

Ключевое слово RETURN передает управление в основную программу на оператор, следующий за обращением к подпрограмме GOSUB.

Результатом выполнения подпрограммы в нашем случае является переменная P, в которой сохраняется значение факториала. Запомним первое полученное значение (фактически N-факториал) в переменной X1. Это необходимо сделать, т. к. подпрограмма будет выполняться несколько раз, и значение переменной P будет меняться.

Затем мы повторяем действия для переменных M и (N – M) соответственно. Причем, в последнем случае уже не надо запоминать полученное значение факториала в новой переменной, т. к. больше обращений к подпрограмме не будет. Осталось вывести на экран полученное значение и запросить пользователя, будет ли он работать с программой еще.

Предупреждение

Привыкайте оформлять самостоятельные куски программ в виде подпрограмм, это способствует четкой структуризации, осознанию внутренних логических связей и пониманию со стороны коллег-программистов, что не маловажно. Программирование с использованием подпрограмм — это хороший стиль!

Выполним теперь ряд упражнений с использованием подпрограмм.

345. Для каждого двумерного массива: $X(3,4)$, $Y(5,3)$, $Z(4,6)$ определите номер строки с максимальной суммой положительных элементов.

346. Напишите программу, предлагающую пользователю меню из 10 функций и строящую по ним графики этих функций в зависимости от выбора пользователя.

347. Напишите программу для младших школьников, проверяющую знание ими таблицы умножения от 2 до 12. Учащемуся задаются 5 примеров перемножения случайных чисел в заданном интервале. Оценкой является количество правильных ответов. Используйте подпрограмму для печати замечаний в ответ на каждый результат, вводимый пользователем. За правильный ответ — замечание должно быть поощрительным, за неправильный — сожалеющим. Чтобы сделать опрос более интересным, необходимо заготовить по десять замечаний для правильных и неправильных ответов и выбирать их случайным образом, обращаясь при этом к пользователю по имени, запрошенному в начале программы. Сделайте красочную заставку, легкое музыкальное сопровождение тоже не будет лишним.

348. Напишите программу, рисующую цветное кольцо. Используя ее в качестве подпрограммы, нарисуйте олимпийский флаг.

349. Напишите программы, находящие минимальное и максимальное числа из трех чисел X, Y и Z, введенных с клавиатуры. Используя их в качестве подпрограмм, напишите программу, вычисляющую значение следующей функции:

$$S = \frac{\max^2(x, y, z) - 2^x \min(x, y, z)}{\sin 2 + \max(x, y, z) / \min(x, y, z)}.$$

350. Даны два одномерных массива из 20 элементов каждый. Элементом является случайное целое двузначное число. Напишите программу с использованием подпрограммы, которая изменяет исходный массив путем деления четных чисел на их индексы. Применяя эту подпрограмму, определите, в каком из массивов было произведено больше замен.

4.3.7. Файлы последовательного и прямого доступа

В языках программирования высокого уровня обычно существуют два метода доступа к информации, хранящейся в файлах:

- последовательный доступ;
- прямой доступ.

Файлы последовательного доступа наиболее просты как в организации, так и в работе с ними. Записи обрабатываются последовательно одна за другой. Информация в таких файлах хранится в текстовом формате в кодах ASCII. Подобные файлы легко просматривать на экране, используя любой простейший редактор. Но, как всегда, у каждой медали две стороны. Простота — хорошо, а последовательность в данном случае — плохо. Если информация об интересующих нас объектах упорядочена в файле по алфавиту, то нам всякий раз придется перебирать практически весь файл, чтобы доб-

раться до нужной записи. Отсюда при большом информационном объеме файла обработка его резко замедляется.

Файлы прямого доступа хранят информацию в специальном формате, в котором каждая запись занимает строго фиксированную, одинаковую с остальными длину. То, что такие файлы могут занимать на диске больше места, чем файлы последовательного доступа, с лихвой компенсируется скоростью работы с ними.

Предупреждение

Если при каждом обращении к файлу вы собираетесь использовать почти все данные, а менять их содержимое часто не предполагается, то выбирайте метод последовательного доступа. Его применение будет и более оптимальным, и облегчит вам программирование. Прямой доступ к файлу целесообразен в том случае, когда требуется часто менять содержимое записей и просматривать их в произвольном порядке. Так, адреса для рассылки корреспонденции уместнее хранить в последовательном файле, а меняющиеся от спектакля к спектаклю бронированные места в зале театра — в файле прямого доступа.

Задания.

351. Школе необходим последовательный файл для учета выпускников.

- Создайте для канцелярии последовательный файл по учету выпускников. Храните в нем фамилию, имя, год выпуска, любимый вид спорта и нынешний род занятий выпускника. Для образца составьте файл на десять человек.
- Воспользуйтесь этим файлом и напечатайте приглашения на очередной домашний матч "Зенита" тем выпускникам, которые назвали футбол своим любимым видом спорта.

352. Компьютерная фирма ведет файл со сведениями о двадцати своих сотрудниках.

- Создайте последовательный файл, содержащий имя и адрес каждого сотрудника (с указанием улицы, дома, квартиры и почтового индекса).
- По содержимому файла напечатайте почтовые адреса для рассылки чеков еженедельной заработной платы.

353. Гидрометцентр ведет статистику выпадения снега по регионам, для каждого из которых заведен последовательный файл. Во всех файлах присутствуют три элемента данных: имя метеоролога, название региона, количество выпавшего за зиму снега в мм.

- Напишите программу ввода данных, заполните файлы для трех регионов.
- Просмотрите все три файла и подсчитайте средний уровень снежных осадков по трем областям. Результат выведите на экран.

354. Налоговая инспекция поощряет налогоплательщиков, вносящих подоходный налог до истечения апрельского контрольного срока, делая им скидку.

- Создайте файл, в котором содержались бы имена, сведения о сроках уплаты и размере налога для каждого налогоплательщика (ограничьтесь группой из шести человек).
- Пусть ваша программа читает файл и делает скидку в 10% для тех, кто уплатил налог досрочно, а также выводит на экран их имена и размер скидки в рублях.

355. Фабрика игрушек ведет учет фирм розничной торговли, сбывающих ее продукцию. Файл контрагентов содержит названия этих фирм, сведения об их местоположении и индекс кредитоспособности: низкая или высокая.

- Напишите программу, которая создала бы последовательный файл контрагентов.
- Напишите программу, которая создала бы два последовательных файла с именами Good.dat и Bad.dat соответственно для фирм с высокой и низкой кредитоспособностью.
- Пусть ваша программа спрашивает у бухгалтера, какой из двух списков ему представить, а затем выдает названия фирм и их местоположение из соответствующего файла.

356. Предположим, адвокат Михаил Бурщевский с помощью компьютера ведет учет своих клиентов и их дел (табл. 4.4).

- Напишите программу, которая позволяла бы ему вводить в последовательный файл следующие сведения: имя клиента, обвинение, исход дела.
- Клиент Мицкевич "из огня попадает в полымя". Напишите программу, которая заменяла бы неопределенное решение суда на "Проиграно".
- Напечатайте обновленный файл.

Таблица 4.4. *Исходные данные задачи*

Имя клиента	Обвинение	Исход дела
Сердюков	Клевета	Выиграно
Прохоров	Оскорбление	Проиграно
Мицкевич	Поджог	?????
Максимова	Взлом	Выиграно
Лерман	Взятка	Проиграно

357. Хоккейные команды "Черные ястребы" и "Красные крылья" хранят в последовательных файлах имена всех своих двенадцати нападающих, число заброшенных ими шайб, сделанных голевых передач и заработанное штрафное время.

- Создайте файлы Black.dat и Red.dat, содержащие информацию о каждой из двух команд.
- Ваша программа по данным, извлеченным из этих файлов, должна создавать новый файл Allstars.dat, в котором содержались бы имя, команда и сумма очков (голы и передачи) для шести лучших игроков обеих команд. Пусть имена и показатели результативности хоккеистов выводятся на экран.

358. Имена и адреса всех, кто обращается за информацией в фирму, попадают в список рекламной рассылки.

- Создайте основной файл Master.dat из десяти записей в качестве списка рассылки и меньший файл Family.dat из пяти записей для вновь обратившихся с запросами в фирму. Добавьте данные из второго файла в конец первого.
- Напишите программу, которая случайным образом выбирала бы из основного файла одну запись и посылала бы адресату письмо с уведомлением о выигрыше приза.

359. Инспектор колледжа ведет файл академических занятий студентов.

- Создайте последовательный файл и заполните его фамилиями, названиями академических курсов и оценочным коэффициентом студентов. Воспользуйтесь данными, перечисленными в табл. 4.5.

Таблица 4.5. *Исходные данные*

Фамилия студента	Курс	Оценочный коэффициент
Югов	Программирование	78
Северов	Японский язык	91
Западов	Психология	56
Востоков	Психология	45
Зюйдов	Корейский язык	89
Вестов	Программирование	66
Полюсов	Психология	90

- Выберите "умных" студентов, т. е. тех, кто имеет оценку выше 88, и запишите сведения о них в файл Best.dat. Пусть программа помогает

инспектору формировать на основе этого файла группы углубленного обучения. По названию курса она должна выдавать список "умных" студентов, зачисленных в такую группу.

4.4. Задания повышенной трудности, интегрированные, азартные

Если вы дошли до сих слов, дорогой читатель, то дальше, мы полагаем, вы сможете идти один, гордо неся над миром знамя российского программирования (ведь только в США по некоторым данным 80% разработчиков программного обеспечения и математиков — наши соотечественники! Надеемся, правда, что вы останетесь в России, чтобы помочь ей занять в мире надлежащее место).

Напоследок мы хотим предложить вам ряд разнообразных задач, поломав голову над которыми, вы приобретете неоценимую закалку и "опыт, сын ошибок трудных". Кроме того, подобные задания всегда хочется видеть не просто работающими, но и оформленными, и снабженными Fool proof, т. е. такими, чтобы с ними было приятно работать. Задачи будут без решений. И если что-то не получается или хочется чем-то поделиться, то обращайтесь в редакцию, меня найдут.

Удачи!

4.4.1. Угадайка (математика и программирование)

Компьютер случайным образом загадывает число от 1 до 100. Задача пользователя — за минимальное количество попыток, но не более шести, угадать это число, предлагая компьютеру свои варианты, в ответ на которые программа указывает номер попытки и сообщение о том, больше загаданное число или меньше введенного с клавиатуры. При правильной стратегии (делении интервала чисел пополам) наверняка угадать число можно за 7 попыток (2^7=128). Именно поэтому человеку дается только шесть попыток, чтобы у компьютера тоже был шанс. Игра состоит из пяти партий, и, как показывает практика, человек не всегда побеждает!

4.4.2. Анаграммы (русский язык и программирование)

Игра со словами. Анаграмма — это слово, в котором перепутаны буквы, например, "ШАДОЛЬ" — это "ЛОШАДЬ", а "ТИВОНКР" — это "ВТОРНИК". Программа рассчитана на двух игроков, соревнующихся друг с другом. В качестве слов используются только имена существительные, нарицательные, в единственном числе, в именительном падеже. Первый игрок вводит с клавиатуры слово длиной не менее пяти, но не более восьми букв (постарайтесь сделать так, чтобы во время ввода на экране вместо букв отображались символы *). Затем компьютер определяет длину введенного слова,

разбирает его на отдельные символы, заносит их в массив, откуда случайным образом выводит на экран. Соперник в течение 2-х минут (время отсчитывает компьютер) должен определить это слово. В случае правильного ответа (вводит его с клавиатуры) он получает 1 очко и сам загадывает слово первому игроку. Игра идет до 5 очков. За неправильный ответ очки не начисляются. Безусловно, приветствуется дружественный интерфейс: запрос и обращение к игрокам по именам, вывод на экран правил игры, графическое и звуковое оформление.

4.4.3. Стрельба из пушки (физика, математика и программирование)

Вариант 1

В замке в заточении томится прекрасная принцесса. В стене замка имеется небольшое окошко-бойница. У нас есть пушка. Необходимо передать принцессе план побега, выстрелив им из пушки (и, естественно, попав в окошко). Дается всего три попытки. Исходными данными являются угол наклона ствола пушки по отношению к горизонту и начальная скорость полета ядра. Они запрашиваются с клавиатуры. Предусмотрите примитивное изображение пушки (рис. 4.83), поворот ствола в зависимости от введенного угла, вычерчивание траектории полета ядра, реакцию замка на попадание ядра в стену, вывод на экран текста плана побега в случае правильного попадания.

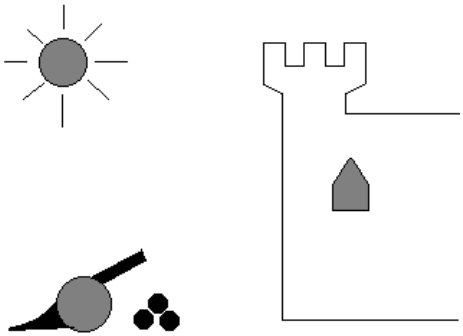


Рис. 4.83. Стрельба из пушки: вариант 1

Вариант 2

Дуэль на пушках через каменную стену — кто кого. Пушки теперь есть у двух игроков. Они небольшого размера. Стреляют по очереди, также задавая угол и скорость полета ядра в пределах: угол от 10° до 90°, начальная скорость от 30 до 100 м/с. Попадание засчитывается, если расстояние от центра

ядра до центра колеса пушки не более 20 экранных точек. В таком случае противник проигрывает. Высота стены каждый раз выбирается случайным образом в пределах от 50 до 200 экранных точек (рис. 4.84). Игра идет до трех побед.

Рис. 4.84. Стрельба из пушки: вариант 2

4.4.4. Царь-пушка (математика, физика, экономика, история, русский язык и программирование)

Выполните расчет рентабельности изготовления ядер для "Царь-пушки".

Царь выделил средства на изготовление наиболее рентабельных ядер для Царь-пушки. Необходимо написать программу, рассчитывающую рентабельность изготовления ядер для 17 различных материалов, сэкономив таким образом для государственной казны значительные средства, которые могли бы быть затрачены на проверку рентабельности в полевых условиях с использованием натуральных материалов.

Исходные данные таковы. Отпущено денег 10 млн. руб. Калибр Царь-пушки, фамилию мастера и год изготовления выясните дома. Взяв из табл. 4.6 исходные данные, создайте файл Pushka.dat.

Таблица 4.6. *Исходные данные для задания "Царь-пушка"*

№	Материал	Плотность, г/см^3	Себестоимость, руб./кг	Коэффициент убойной силы
1	Платина	21,5	3636,36	1000,0
2	Золото	19,3	3090,91	950,0
3	Свинец	11,3	178,18	900,0
4	Серебро	10,5	454,55	800,0
5	Медь	9	81,82	300,0

Таблица 4.6 *(окончание)*

№	Материал	Плотность, г/см^3	Себестоимость, руб./кг	Коэффициент убойной силы
6	Никель	8,9	92,73	340,0
7	Железо	7,9	29,09	500,0
8	Олово	7,3	67,27	550,0
9	Титан	4,5	203,64	450,0
10	Алюминий	2,7	21,82	450,0
11	Глина	2,2	3,64	0,1
12	Каучук	0,9	25,45	130,0
13	Цинк	7,1	60,00	610,0
14	Графит	2,5	9,09	2,5
15	Латунь	8,24	73,09	230,0
16	Мельхиор	5,42	84,00	200,0
17	Чугун	7,63	28,09	440,0

Здесь необходимо перевести плотности в систему СИ. Далее вычислить (все вычисления производить в системе СИ):

- объем ядра (ядро идеально круглое);
- массу ядра (M);
- начальную скорость (по формуле $V_n = \frac{3000}{\sqrt{M}}$);
- цену одного ядра (C);
- количество ядер (N);
- дальность полета (S), если выстрел производится под углом 45°;
- убойную силу ($US = \frac{M \times K \times V_n}{S}$, где K — коэффициент убойной силы);
- рентабельность ($R = \frac{N \times US}{C}$).

Напишите программу, которая поможет:

- найти пять наиболее рентабельных материалов, их рентабельности занести в массив и упорядочить по возрастанию;
- из самой большой рентабельности взять вторую цифру;

- из следующей по убыванию — число, составленное из первых двух цифр, поделить на два;
- в очередной по убыванию — сложить цифры целой части и прибавить число, которое получится, если число, составленное из двух первых цифр пятой рентабельности, поделить пополам;
- из четвертой рентабельности взять разность второй и третьей цифр.

Полученные четыре значения соответствуют порядковым номерам букв русского алфавита, в котором отсутствует буква "е". Необходимо определить эти буквы и составить из них четырехбуквенное слово, которое является первым результатом выполнения задания. Второе слово получаем следующим образом: из фамилии мастера берем три последние буквы, добавляем букву, чей порядковый номер — вторая цифра года создания Царь-пушки, и букву, чей порядковый номер получается делением на 2 числа, образуемого второй и четвертой цифрами года создания Царь-пушки. Составляем пятибуквенное слово.

4.4.5. Кинотеатр "Кристалл-Палас" (математика, экономика и программирование)

И еще одно очень объемное задание. Представьте себе, что вас приглашают на должность директора в кинотеатр "Кристалл-Палас", расположенный на Невском проспекте. Когда вы пытаетесь узнать, сколько же вам будут платить, то говорят, что сейчас заодно и проверят ваши деловые способности, и предлагают написать программу, используя следующие исходные данные. В кинотеатре "Кристалл-Палас" три зала: "Красный", "Зеленый" и "Синий". Залы одинаковы и представляют собой квадраты 11×11 кресел. В результате маркетинговых исследований были установлены показатели по заполнению зала в среднем (рис. 4.85).

8	9	7	9	9	0	8	9	7	2	5
1	5	8	9	3	9	1	1	0	7	0
5	8	8	4	0	4	6	0	0	4	0
4	1	2	3	7	3	1	7	9	7	3
4	8	9	8	3	6	5	0	3	7	3
6	8	1	6	8	1	4	3	5	5	7
3	1	0	7	5	2	6	3	9	6	5
1	1	6	7	9	6	3	1	4	3	3
8	6	3	6	9	3	1	2	5	3	6
9	6	5	9	5	2	2	0	1	2	1
7	3	7	1	2	6	9	2	5	7	3

Рис. 4.85. Рассадка зала

Первым делом, чтобы облегчить себе жизнь, создайте файл исходных данных, содержащий массив 11×11, элементами которого будут цифры из таблицы. Здесь цифры 1 и 2 обозначают мужчин, 3 и 4 — женщин, 5 и 6 — детей, 7 и 8 — тех, кто имеет право на льготные билеты, 0 — пустые места и 9 — знакомые и родственники служащих кинотеатра, которые ходят в кино бесплатно. К исходным данным также относится курс доллара на сегодняшний день. Цены на билеты в зависимости от времени сеанса и принадлежности к той или иной категории зрителей приведены в табл. 4.7, а затраты от общей прибыли — в табл. 4.8.

Таблица 4.7. *Цены билетов в кинотеатре "Кристалл-Палас"*

	Цена билетов (руб.)		
Категория зрителей	**Утро (1 сеанс)**	**День (3 сеанса)**	**Вечер (2 сеанса)**
Мужчины	70	150	300
Женщины	50	100	200
Дети	30	50	100
Льготники	10	20	50

Таблица 4.8. *Затраты из общей прибыли за месяц*

Затраты	**%**	**Затраты**	**%**
Амортизация	12	Охранные структуры	25
Налоги	22	Новые фильмы	22
Зарплата сотрудникам	16	Директор	3

Необходимо вычислить прибыль кинотеатра за день, за месяц (считая, что в месяце 30 дней), расходы по статьям в рублях и долларах. Вывести эти данные на экран, выделив зарплату директора цветом.

4.4.6. Тараканьи бега (математика, дизайн и программирование)

Конечно, сейчас уже далеко не все из вас смотрели фильм "Бег", поставленный по роману М. Булгакова. В частности, в нем есть сюжет о том, как наши эмигранты первой волны зарабатывали себе в Париже на жизнь. Одним

из источников их доходов были тараканьи бега. Попробуем реализовать их компьютерную версию.

- Часть 1. Красочная заставка.
- Часть 2. Представление кличек участвующих тараканов, запрос количества делающих ставки и их фамилий, выдача всем кредита в 200 единиц, прием ставок (не более 100, но не менее 30).
- Часть 3. Оформление игрового поля примерно, как на рис. 4.86.
- Часть 4. Непосредственно забег. Скорость каждого таракана за один такт — случайное целое число от 1 до 5. Забег заканчивается, как только какой-либо таракан коснулся "головой" финишной черты. Если никто из сделавших ставки не выиграл, деньги переходят в доход казино. В случае выигрыша игрока, ему начисляется двойная ставка. Игра идет до тех пор, пока кто-либо не разорится или кто-либо не наберет сумму в размере двойного кредита.
- Часть 5. После каждого забега выводится информационное сообщение о финансовом положении участников и статистике побед тараканов.

Рекомендуется оформить забег музыкально.

Рис. 4.86. Игровое поле для "тараканьих бегов"

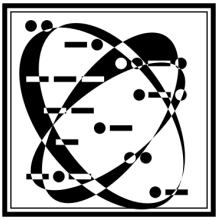
4.4.7. Тесты (психология, русский язык и программирование)

Для этого задания вы можете взять любой тест из многочисленных выпускаемых сейчас журналов. Создайте файл исходных данных — вопросов и вариантов ответов. Напишите программу, знакомящуюся с пользователем, сообщающую ему цель теста и инструкцию по работе с программой, проводящую тест, анализирующую результаты и выводящую итоги, взятые из другого файла данных.

Такая программа может использоваться как для психологических тестов, так и для проверки знаний в какой-либо области.
Если с одним тестом у вас получилось, то можно написать целую гибкую систему тестирования, где пользователю предлагается выбор из нескольких тестов, а также непосредственное создание тестов самому.

На этой же базе можно реализовать популярную телевизионную игру "Кто хочет стать миллионером".

Глава 5



Информационные технологии

Непрестанно совершенствующийся и обновляющийся пакет программ Microsoft Office позволяет нам творить чудеса. Из всех составных частей Office для обычного человека, на наш взгляд, крайне важен *текстовый процессор* Word, которому мы уделим повышенное внимание. Кроме того, современному пользователю обязательно надо знать основы работы с *электронными таблицами Excel*. Желательно уметь делать хотя бы простые *презентации* в PowerPoint. Надо ознакомиться с конструированием несложных *баз данных* в Access. Ну, а возможности *Интернета* нам подарят Internet Explorer и Outlook Express.

5.1. Обработка текстовой информации. Первое знакомство с MS Word

При всем многообразии прикладных программ для обработки текстовой информации в данный момент стандартом является Word из Microsoft Office. При постановке своих заданий и упражнений мы и будем ориентироваться на него. Word позволяет готовить документы практически любого уровня сложности, а также дает возможность вести верстку школьных изданий, если нет времени на изучение, например PageMaker.

Основные возможности Word, с которыми мы будем работать, следующие:

- набор текста;
- редактирование текста (вставка и удаление символов, рассечение, соединение, вставка и удаление строк и т. д.);
- задание параметров страницы (поля, ориентация и т. д.);
- сохранение текста в виде файла и его печать;
- шрифтовое форматирование (изменение размера и начертания шрифта, выбор гарнитуры и т. д.);
- абзацное форматирование (выравнивание абзацев, задание красной строки, границ абзаца, межстрочного интервала и т. д.);

- копирование и перенос фрагментов текста, настройка по образцу, одновременная работа с несколькими документами;
- проверка орфографии, расстановка переносов;
- оформление абзацев и списков;
- автозамена;
- форматирование текста в колонки;
- вставка таблиц;
- использование колонтитулов и автоматическая нумерация страниц;
- поиск и замена элементов текста;
- написание математических формул;
- оформление текста графическими объектами;
- использование фигурного текста;
- связь документов гиперссылками и установка закладок;
- создание простейших Web-страниц;
- создание документов по шаблону

и многое другое.

Прежде чем мы перейдем непосредственно к работе по набору, редактированию и форматированию документа, хотелось бы ознакомить вас с тем, как открыть уже существующий на диске файл, как создать новый и как сохранить созданный документ.

5.1.1. Создание, сохранение документов. Открытие уже существующих документов. Переход от одного документа к другому через меню *Окно*

Практикум 1. Создание и сохранение документов

1. Загрузите Word. Наберите текст русской народной загадки `Сто одежек и все без застежек`. Сохраните набранный текст в файле в папке Zagadki под именем Zagadka1.doc следующим образом:

   Выбрать команду меню **Файл | Сохранить** или нажать кнопку на панели инструментов. В появившемся диалоговом окне выбрать требуемый диск, открыть нужную папку (необходимо добиться, чтобы в поле ввода **Папка** было видно название требуемой папки). Затем нажать кнопку создания новой папки, дать ей требуемое имя, открыть ее, дать имя файлу, после чего нажать <Enter> или кнопку **Сохранить**.

2. Создайте новый документ, не закрывая предыдущего:

   Выбрать команду меню **Файл | Создать | Новый документ** или нажать кнопку на панели инструментов.

3. Наберите текст русской народной загадки `Зимой и летом одним цветом`. Сохраните набранный текст в файле в своей папке под именем Zagadka2.doc.
4. Аналогично создайте два файла с отгадками под названиями Otgadka1.doc и Otgadka2.doc.
5. Через меню **Окно** перейдите к документу Zagadka1.doc и наберите на следующей строке вопрос `Отгадай, что это?`.
6. Аналогично для документа Zagadka2.doc. Сохраните и закройте все документы.

Практикум 2. Сохранение документа под новым именем (*Сохранить как*)

1. Откройте уже готовый документ Tabel.doc (рис. 5.1).

Табель за _____ четверть
ученика ____ класса
фамилия, имя
Алгебра — ____
Геометрия — ____
Рус. яз. — ____
Ин.яз. — ____
Химия — ____
Физика — ____
Информатика — ____
Классный руководитель
<Фамилия, имя, отчество>

Рис. 5.1. Табель

2. Впишите туда номер четверти, свои имя, фамилию, класс, оценки, фамилию, имя, отчество классного руководителя. Сохраните документ в своей папке под именем Tabel_my.doc:

   Выбрать команду меню **Файл | Сохранить как**. Найти и открыть свою папку. Дать имя сохраняемому документу и нажать <Enter>.

5.1.2. Набор и редактирование текста

Правильно организовывать работу с документом надо следующим образом. Первая часть работы — это скучный однообразный труд по набору текста (облегчить его могут, если они вам доступны, сканер или голосовой ввод), вторая часть — форматирование и оформление набранного — творческая радостная работа. Тем не менее, кто-то должен заниматься и первой частью. А чтобы облегчить дальнейшую, вторую часть, надо знать основные правила набора текста на компьютере. Вот они:

- *Слово* для компьютера — это набор любых символов между пробелами.
- *Абзац* для компьютера — это набор любых символов между символами конца абзаца. Новый абзац получается, как только вы нажимаете клавишу <Enter>.
- *Перенос* текста по строкам компьютер производит автоматически, поэтому клавишу <Enter> надо нажимать только в конце абзаца.
- Между словами допустим только один *пробел*.
- После таких знаков препинания, как *точка*, *запятая*, *двоеточие*, *точка с запятой* обязателен пробел, перед ними — нет.

Замечание

Это связано не только с благоприятным визуальным восприятием текста, но и с проверкой орфографии: т. к. для компьютера слово — это набор любых символов между пробелами, а ошибки он ищет, сравнивая написанные слова с имеющимся словарем, то если в предложении: "Скоро пойдет дождь, потому что на небе собираются тучи", не поставить пробел после запятой, то компьютер подчеркнет "дождь,потому" как несуществующее слово, хотя вроде бы никаких видимых грамматических ошибок здесь нет. Из этого правила есть и исключения — например, в датах и после инициалов (13.10.79 или А.А.Назарова) пробелы не ставятся, чтобы не возникло недопустимых переносов типа 13.10.

79

или А.А.

Назарова

- *Дефис* не берется в пробелы, а вот *тире* — обязательно. Например, "кое-где", "как-нибудь", но "Зенит — чемпион!" или "Санкт-Петербург — культурная столица России".
- *Текст, взятый в кавычки или скобки*, не отделяется от них пробелами (см. предыдущий пункт — там как раз и с кавычками пример есть).
- *Римские цифры* набираются латинскими заглавными буквами и обозначаются так: 1 — I, 5 — V, 10 — X, 50 — L, 100 — C, 500 — D, 1000 — M. Год 1979, таким образом, запишется как MCMLXXIX. Красиво, но применяется редко — в основном в написании веков.

- Еще есть правило *"неразрывного дефиса"*. Он вводится при помощи одновременного нажатия клавиш <Ctrl>+<Shift>+<Дефис>. Неразрывный дефис применяется для улучшения внешнего вида документа, т. к. препятствует нежелательному переносу слов, его содержащих, например, при написании двойных фамилий (Римский-Корсаков).
- Знак *"неразрывный пробел"* препятствует символам, между которыми он поставлен, располагаться на разных строчках. Этот знак очень удобно применять при вводе дат, фамилий с инициалами и т. п., например: О.Григорьев, 2002 г., 14 февраля, 1682—1696 гг. Ставится он при одновременном нажатии клавиш <Ctrl>+<Shift>+<Пробел>. Тогда у вас никогда не получится так: 1682—1696
  гг.
  Кроме того, если Word автоматически не превращает в тире дефис, взятый с двух сторон в пробелы, то это можно сделать принудительно при помощи одновременного нажатия следующих клавиш:
  - просто тире — <Ctrl>+<Shift>+<Дефис>;
  - длинное тире — <Ctrl>+<Alt>+<Серый минус>.

Обычно занятия по Word мы начинаем с упражнения по выделению фрагментов текста и обучения основным приемам редактирования вводимого или уже введенного текста. Правила выделения фрагментов текста в Word сведены в табл. 5.1.

Таблица 5.1. *Правила выделения фрагментов текста в Word*

Выделяемый элемент	Методика выделения
Произвольный фрагмент	Протаскиванием мыши от начала фрагмента до его конца (или наоборот). Позволяет выделить от одного символа (хотя бы даже и точки или запятой) до всего текста. Универсальный метод, к которому, к сожалению, быстро привыкают, и другие, более быстрые способы (приведены ниже) не применяют
Слово	Двойной щелчок левой кнопкой мыши внутри выделяемого слова
Строка	Одинарный щелчок левой кнопкой мыши в *области выделения* (пустая зона слева от строки, указатель мыши принимает в ней вид белой стрелки, направленной вправо)
Несколько строк	Протаскиванием мыши в зоне выделения
Предложение	Одинарный щелчок левой кнопкой мыши при удерживаемой клавише <Ctrl>
Абзац	Двойной щелчок левой кнопкой мыши в области выделения или тройной щелчок левой кнопкой мыши внутри абзаца

Таблица 5.1 *(окончание)*

Выделяемый элемент	Методика выделения
Весь текст	Тройной щелчок левой кнопкой мыши в области выделения или одинарный щелчок левой кнопкой мыши в области выделения при удерживаемой клавише <Ctrl>
Снять выделение	Одинарный щелчок левой кнопкой мыши в любом месте рабочего поля

И пока вы не начали набирать и оформлять документы в Word, покажем две очень полезные для начинающих пользователей кнопки. Они выглядят так . Левая из них называется **Отменить**, правая — **Вернуть**. Word запоминает ваши последние действия по работе над текстом, и их можно отменить или вернуть ранее отмененные. Поэтому — если что-то сделали не так, не надо паниковать, воспользуйтесь спасительными кнопками. Но — это касается только действий над текстом.

Теперь, после ознакомления с основными правилами набора, коротко расскажем о приемах элементарного редактирования текста.

Удаление символов

Для *удаления символов* в Word используются две основные клавиши: <Delete> и <Backspace>. Клавиша <Delete> удаляет символы, стоящие справа от курсора (вертикальной мигающей черточки), а клавиша <Backspace> слева от него.

Например, если у вас набрано слово **СОБ|ЛАКА** и вы хотите удалить лишнюю букву Л, то надо воспользоваться клавишей <Delete>. А если бы курсор стоял вот так: **СОБЛ|АКА**, то клавишей <Backspace>. К сожалению, как показывает опыт, многие пользователи привыкают и для всех случаев жизни используют только клавишу <Backspace>. Помните о <Delete>! Она тоже хочет удалять.

Если же у вас выделен фрагмент, то он удаляется как <Delete>, так и <Backspace> — тут уж чем хотите.

Практикум 3. Удаление лишних символов и фрагментов

1. Откройте файл Lishniy.doc (рис. 5.2).
2. Последовательно удалите из текста (все, подлежащее удалению, выделено в нем курсивом):
   - часть второй строки первого абзаца;
   - третий абзац;

Рождественские гадания

Гадание на чистом листе

Можно гадать на чистом листе бумаги. Участвуют 2-3 человека. *два-три-четыре, потом пять-шесть-семь, тут же восемь девять*. Каждый сам комкает белый лист, кладет его на чистое блюдо или на большую плоскую тарелку и поджигает. Когда лист сгорит или почти сгорит, на блюде образуется горрка пепла и остаток сгоревшей бумаги.

Теперь следует создать проекцию теней на стене при помощи свечи. Внимательно рассматривая тени, можно усмотреть на стене различные каартины, образы, символы. Остается истолковать их каждым участником гадания или всеми сообща.

Раз в крещенский вечерок девушки гадали — дружно испекли пирог и в окно бросали.

Современный способ гадания

Узнать будущее в наше время можно... по радиоооо или по телевизоооору. Мысленно загадаешь вопрос или загадываешь желание, после этого наобум включаешь радио на любой волне или телевизор. Первые услышанные слова и увиденная картинка — ответ на ваш вопрос. Это может быть песня или речь диджея, монолог юмориста или кинокомедия. *Трактовать сказанное нужно применительно лично к себе.*

Рис. 5.2. Рождественские гадания

- последнее предложение;
- лишние буквы в словах: горка, картина, современный, радио, телевизору.

3. Сохраните документ в своей папке под именем Udalenie.doc.

Для перемещения курсора можно использовать как щелчок в нужном месте левой кнопкой мыши, так и клавиши управления курсором (со стрелками).

Вставка символов

При быстром наборе неизбежны опечатки и пропуски. Так как в Word при загрузке автоматически включается режим вставки, то пропущенные символы вставить очень легко: просто установите курсор в место вставки и наберите нужные символы.

Практикум 4. Вставка символов

1. Откройте файл Vstavka.doc и вставьте все пропущенные символы и слова, предварительно удалив ненужное. В качестве причины следует привести коротенькую душещипательную историю (строчек пять), например на тему "Помогал старушке улицу перейти" (рис. 5.3).

Дирктору <номер> школы

<фамилия, имя, отчество директора>

от ученка <класса>

<свои фамилия, имя>

Обяснителная запика

Я, <имя, отчество>, опоздал сегоня на урок инфортики, потому что <причина опоздания>. Болше не буду.

<Дта>

Рис. 5.3. Объяснительная записка-1

2. Сохраните документ в своей папке под именем Zapiska.doc.

Кроме удаления и вставки нужно знать еще следующие приемы:

- *Рассечение строки* (перенос части ее на строчку ниже) — курсор в место рассечения и <Enter>.
- *Соединение (склеивание) строк* — курсор в конец верхней строки и <Delete> или курсор в начало нижней строки и <Backspace>.
- *Вставка пустых строк* — курсор в начало строки, выше которой должна быть пустая, и <Enter> или курсор в конец строки, под которой должна быть пустая, и <Enter>.

Практикум 5. Приемы элементарного редактирования

1. Загрузите файл Elochka.doc (рис. 5.4). Он специально набран без пробелов для отработки приемов элементарного редактирования.

Влесурадиласьелочкавлесуонарослазимойилетомстройнаязеленаябыламетельейпелалесенкуспнелочкабайбайморосснешкомукутывалсмотринезамерзай

Рис. 5.4. Абракадабра

2. А теперь — преобразуйте этот текст в следующий (рис. 5.5): сначала разбейте на слова, потом на строки, исправьте ошибки, расставьте знаки

препинания, напишите название. Готовый текст сохраните в своей папке под именем Newyear.doc.

Новогодняя песня

В лесу родилась елочка,
В лесу она росла,
Зимой и летом стройная,
Зеленая была.

Метель ей пела песенку:
"Спи, елочка, бай-бай!",
Мороз снежком укутывал:
"Смотри, не замерзай!".

Рис. 5.5. Новогодняя песня-1

В начале каждого учебного года мы проводим во всех классах (с 5-го по 11-й) зачет на скорость набора текста. По нашему мнению, скорость набора обычного человека для нормального общения с компьютером должна быть не ниже 50 символов в минуту. Для этого предлагается следующее задание. Всем ученикам раздается печатный текст, содержащий ровно 1000 символов, включая знаки препинания и пробелы. На выполнение задания выделяется 20 минут. По истечении этого времени работа по набору прекращается и подводятся результаты по следующему алгоритму:

Скорость набора = Кол-во набранных символов /(время, мин. + кол-во ошибок).

Учитываются как грамматические и синтаксические ошибки, так и нарушения приведенных выше правил набора на компьютере. Если учащийся не укладывается в скорость набора 50 символов в минуту, то он должен пересдать зачет на другом тексте в течение года (попыток может быть несколько). Задачи научить десятипальцевому слепому методу печати не ставится, хотя дело это хорошее, только из-за недостатка учебных часов.

Практикум 6. Набор текста на скорость и соблюдение правил набора

Примеры текстов (зачет сдается как по русскоязычной клавиатуре, так и по латинской).

Русские тексты

1 вариант. Документ сохранить под именем Zemletr.doc.

При нахождении во время землетрясения вне дома или места работы, например, в магазине, театре или просто на улице, не следует спешить домой, надо спокойно выслушать указания соответствующих должностных лиц по действиям в создавшейся ситуации и поступать в соответствии с такими указаниями. В случае нахождения в общественном транспорте нельзя покидать его на ходу, нужно дождаться полной остановки транспорта и выходить из него спокойно, пропуская вперед детей, женщин, инвалидов, престарелых. Учащиеся старших классов должны помочь дирекции и учителям в поддержании порядка среди школьников младших классов.

Землетрясение может длиться от нескольких мгновений до нескольких суток (периодически повторяющимися подземными толчками). Примерная периодичность толчков и время их возникновения возможно будут сообщаться по радио и другими доступными способами. Следует свои действия сообразовывать с этими сообщениями.

Главное — спокойствие. Не поддавайтесь панике — это лучший помощник стихийного бедствия!

2 вариант. Документ сохранить под именем Telephon.doc.

Развлечение "Испорченный телефон"

Наверняка все знают эту игру. А теперь немного усложним ее. Для участия потребуется большая компания гостей. Один ведущий, человек 5—6 играющих (чем больше, тем веселее), остальные зрители. Игроки выходят за дверь. Ведущий вызывает первого из них и зачитывает ему в присутствии зрителей заранее приготовленный небольшой текст (10—12 предложений, не слишком сложных). Текст читается один раз внятно и не слишком быстро. Задача игрока — пересказать услышанный текст следующему игроку, который до этого находился за дверью. Второй игрок пересказывает третьему то, что сумел запомнить в свою очередь. И так далее. В итоге последний игрок пересказывает зрителям те крохи первоначального текста, которые до него дошли. В общем, рассказ так изменяется к концу, что зрители просто падают со смеху! Проверено на собственном опыте.

Очень хорошо проводится в дружеских компаниях. Еще лучше будет, если записывать все на магнитофон, а потом давать слушать самим рассказчикам.

3 вариант. Документ сохранить под именем Rolling.doc.

История "Роллинг Стоунз"

Состав был сформирован окончательно: Мик Джеггер — вокал, Кейт Ричард и Брайен Джонс — гитары, Билл Уайман — бас, Чарли Уотс — ударные. Следующим шагом была запись демонстрационных лент. Такие ленты музыканты обычно рассылают в фирмы грамзаписи, стараясь их заинтересовать. Ни одна из фирм, однако, этими записями не заинтересовалась. Но ребята не унывали — их выступления проходили с успехом. Вскоре им удалось добиться постоянной работы в Ричмонде, где ими заинтересовался Джордж Гомельски, или, правильнее, Георгий Гомельский, поскольку отец его был русским бе-

логвардейским офицером, когда-то эмигрировавшим за границу. Гомельский, кроме того, снимал экспериментальные киноленты, и в этот период в становлении группы он сыграл важную роль. Гомельский фактически стал первым серьезным менеджером группы. На этом небольшом примере мы с вами видим, что русское влияние простирается далеко за пределы классического балета, хоккея с шайбой и стрельбы по пересеченной местности.

Тексты на английском языке

1 вариант. Документ сохранить под именем Cars.doc.

I guess, you have difficulty, my little friend. Listen carefully. Carry on along the road towards town Volkhov. The town is well served by main roads. Then take the turning to the left. Go through the small town. Out of the town the road forks, take the right hand fork.

As you come out of the wood at the cross roads you have to turn left and go up to the bridge across the river to the corner of streets Lenina and 2nd Krasnoarmeyskaya. Go under the railway. By the railway bridge turn right and go up to the town Velikie Luki. After about ten kilometres you'll see a country, it is the second turning on the left. At the end of E95 road you'll see a hotel "Ubileynaya". The road winds there through the hills. Behind the hotel you'll see the house of Mr.Sidorov.

If your car doesn't start, you should check three things first: the battery, the fuel level and the sparks plugs. If your car still doesn't start, the petrol pump may be broken or the fuel pipe may be blocked. Finding a fault in a car!

2 вариант. Документ сохранить под именем Football.doc.

Table-tennis or football?

TEACHER: Our sketch today is about sport. We have talked of sports a little and all connected with sport. Nick, come here. We may begin.

NICK: My name is Nick. I'm stranger here. I've come to this town to take part in table-tennis competitions. They say I'm a promising player! Yes, I often win. I often score 20 or more to love. yes, to love this is a sport phrase, for example 2:0 — two to love, love in sport means "0". A good word love, isn't it? For example we say: "All love game" — 0:0. So I play table-tennis.

TEACHER: If you stick to table-tennis now, why are you holding that football?

NICK: This is not my football. I don't play football. I don't understand football. This football was given to me by my coach. My coach has a cousin in this town, Michael. So my coach asked me to give it to his cousin. This is a famous football. It's a lucky one too. There are signatures of our best footballers on it. Just look!

TEACHER: Yes, I see. This football is very unusual. Ok!

3 вариант. Документ сохранить под именем Stones.doc.

"Rolling Stones"

I think the reason why I like the Rolling Stones so much, even more than the Beatles, is that their music had a more raw and edgy feel. Maybe as a man, Mick Jaggers's crowing style of vocals and Keith's gritty street smart guitar riffs touch my soul just the right way. Their instrumental prowess is slightly better than the Beatles

but not overly outstanding except maybe when Mick Taylor was a member of the band. I wish Taylor would have told the Glimmer Twins, "Look dudes, I have some cool songs and want to contribute them to OUR band!" Who knows what the Stones would have been. They did have some spotty releases from the mid seventies to the end of the eighties. However, when you keep going to the well as often as the Stones have, you are bound to come out with a lot of quality material. And over the course of their long lives, the Stones are fully loaded with great songs, riffs, arrangements, and more. I guess that fully explains my feelings about this great band.

5.1.3. Настройка параметров страницы

Word предоставляет пользователю возможность подготовки документов на страницах разного формата, с разными полями и расположением. Все эти настройки устанавливаются командой меню **Файл | Параметры страницы**. На вкладке **Поля** можно установить величину верхнего, нижнего, правого и левого полей. На вкладке **Размер бумаги** — формат страницы (по умолчанию А4, 210×297 мм). Там же выставляется и ориентация страницы — книжная (если страница располагается вертикально) или альбомная (для расположения страницы горизонтально). Все это позволяет наглядно разместить текст и рационально использовать площадь страницы.

Поля можно изменить и вручную. Надо навести указатель мыши на горизонтальную линейку, поставить его на границу между рабочим пространством и полем, уловить момент, когда указатель превратится в двунаправленную стрелку ↔, нажать левую кнопку и, не отпуская ее, изменить поле на столько, на сколько нужно.

Практикум 7. Настройка параметров страницы

1. Откройте файл Parametr.doc (или любой файл, содержащий две и более страниц текста).
2. Настройте поля следующим образом:
   - верхнее — 1,5 см;
   - нижнее — 1,5 см;
   - левое — 2,5 см;
   - правое — 1 см.
3. Сделайте ориентацию второй страницы альбомной (для этого надо ее выделить).
4. Выполните команду меню **Файл | Предварительный просмотр** или нажмите соответствующую кнопку на стандартной панели инструментов. Посмотрите, что получилось.
5. Сделайте ориентацию второй страницы снова книжной.
6. Сохраните документ с новыми полями под тем же именем.

5.1.4. Шрифтовое форматирование

Все возможности по шрифтовому оформлению содержатся в команде меню **Формат | Шрифт**. Причем, напоминаем, компьютеру непременно нужно указать, в каком именно фрагменте текста вы собираетесь производить изменения, а, значит, первым делом этот фрагмент надо выделить. Что же мы можем сделать с текстом, не входя в меню, пользуясь только панелями инструментов? Достаточно много.

Размер шрифта

По умолчанию (если вы ничего не меняли) Word предлагает размер 10 пунктов (1 пункт = 0,376 мм — наименьшая единица размера типографских шрифтов). Вообще же считается, что для основного текста документов стандартом является размер 12, и это можно настроить командой меню **Формат | Шрифт**. Можно также после выделения фрагмента текста щелкнуть левой кнопкой мыши по маленькой черной стрелке справа от цифры 10 на панели инструментов [10]. Откроется список доступных размеров — от 8 до 72. Выбираете любой и чудесным образом размер шрифта в выделенном вами фрагменте изменится.

Возникает вопрос — а можно ли использовать другие размеры, не учтенные приведенным списком? Можно! Word позволяет применять пропорционально изменяющиеся размеры, начиная от 1 пункта и заканчивая размером 1638 пунктов. Для этого, как всегда, сначала надо выделить фрагмент, затем щелкнуть левой кнопкой мыши непосредственно на окне размера, набрать там свой размер и нажать <Enter>.

Изменение начертания шрифта

Для этого используются кнопки [Ж К Ч]. Кнопка с буквой **Ж** сделает выделенный текст полужирным, кнопка с буквой ***К*** — курсивом (наклонным), а кнопка с буквой **Ч** — подчеркнутым одинарной линией. Эти три кнопки работают как выключатели — Вкл/Выкл — и вы можете делать с их помощью комбинации, например полужирный курсив, или подчеркнутый курсив, или полужирный подчеркнутый, или полужирный подчеркнутый курсив!

Изменение самого шрифта (гарнитуры)

В Windows имеется набор стандартных шрифтов, которые расположены в папке Fonts. В Word они выбираются инструментом [Times New Roman]. Шрифт Times New Roman назначен по умолчанию и считается стандартным для деловых документов. Выбор шрифта осуществляется аналогично выбору размера, из открывающегося списка. К сожалению, некоторые стандартные

шрифты Windows не поддерживают кириллицу, например Modern, а некоторые являются символьными, например Wingdings, Webdings или Symbol. При выборе этих шрифтов текст непредсказуемо преобразуется в набор непонятных символов, неопытный пользователь хватается за голову — все пропало! Да нет, все нормально, выберите другой шрифт, например тот же Times New Roman, и текст вернется к прежнему виду.

Изменение цвета шрифта

Достигается выбором цвета кнопкой .

Выполнить команду меню **Формат | Шрифт** можно самостоятельно ознакомиться с *дополнительными возможностями* шрифтового оформления, из которых обязательно отметим верхний и нижний индексы — для написания выражений типа x^2 (верхний индекс) или H_2O (нижний индекс). Есть там также и разнообразные подчеркивания, и увеличение/уменьшение интервала между символами и даже анимация — это скорее уже для электронной почты.

В меню **Формат** можно найти и такую интересную опцию, как **Буквица**. Это такая первая буква текста, выделяющаяся своими размерами и необычайной красотой, прямо как в сказке. Вспомним детство золотое (рис. 5.6).

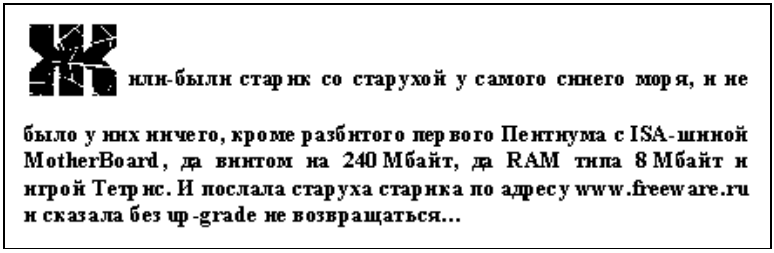


Рис. 5.6. Буквица

Ну а теперь попрактикуемся!

Практикум 8. Шрифтовое оформление

Задание 1. Загрузите файл Newyear.doc. Заголовок "Новогодняя песенка" сделайте полужирным шрифтом размером 20 пунктов. Гарнитуру выберите Comic Sans MS. Основной текст песенки — размер 14, курсив, гарнитура Arial. Слово "елочка" в основном тексте сделайте полужирным, подчеркните двойной волнистой линией, выделите зеленым цветом. Должно получиться так, как показано на рис. 5.7.

Сохраните документ под тем же именем.

Новогодняя песня

В лесу родилась елочка,
В лесу она росла,
Зимой и летом стройная,
Зеленая была.

Метель ей пела песенку:
"Спи, елочка, бай-бай!",
Мороз снежком укутывал:
"Смотри, не замерзай!".

Рис. 5.7. Новогодняя песня-2

Задание 2. Наберите и оформите следующий текст (рис. 5.8). Размер основного текста 12, заголовок и формулы — 15, полужирный. Заголовок еще и подчеркнутый. Подпись исполнителя — размер 13, курсив.

Решение квадратного уравнения.

Чтобы решить квадратное уравнение вида

$$ax^2 + bx + c = 0$$

необходимо сначала вычислить *дискриминант* по формуле:

$$D = b^2 - 4ac$$

Если $D < 0$, то уравнение не имеет вещественных корней.

Выполнил ученик 9а класса Пифагоров Иван.

Рис. 5.8. Решение квадратного уравнения

Сохраните документ под именем Kvadur.doc.

Задание 3. Наберите и аналогично предыдущему оформите следующий текст (рис. 5.9). Отступ от названия кислоты до ее формулы производите ни в коем случае не пробелами, а клавишей <Tab>.

Сохраните документ под именем Kislota.doc.

Задание 4. Придумайте и напишите начало компьютерной пародии на любую русскую народную сказку (например, "Колобок" или "Репку"), используя буквицу и шрифтовое оформление. Сохраните документ под именем Skazka.doc.

Некоторые кислоты и их соли

Азотная	HNO_3
Дихромовая	H_2CrO_7
Серная	H_2SO_4
Угольная	H_2CO_3
Хромовая	H_2CrO_4
Хлорноватая	$HClO_3$

Выполнил ученик 9б класса Менделеев Илья.

Рис. 5.9. Некоторые кислоты и их соли

5.1.5. Абзацное выравнивание

Теперь, когда текст набран и шрифт его оформлен в соответствии с нашими пожеланиями, настала пора оформить абзацы. Наверняка ваш текст уже хорош, но вот маленькая беда — скорее всего правая граница его очень неровная, заголовки не по центру, переносы не расставлены и межстрочный интервал какой был, такой и остался. Но это все поправимо. Начнем поправлять. Обратим свой взгляд на следующие кнопки на панели инструментов — . Если на каждую из них навести указатель мыши , то мы увидим так называемые *всплывающие подсказки*, которые сообщат нам назначение этих кнопок. Вот они, слева направо: **По левому краю**, **По центру**, **По правому краю** и **По ширине**. Действие этих кнопок распространяется как на текущий абзац (в котором в данный момент находится курсор), так и на выделенные абзацы.

Кнопка **По левому краю** включена по умолчанию и выравнивает левый край текста, а правый уж как получится, например:

"К предвестникам возможных землетрясений следует отнести следующие признаки, которые особенно должно знать население сейсмически опасных районов; это — появление запаха газа в районах, где до этого воздух был чист и ранее подобное явление не отмечалось, беспокойство птиц и домашних животных, вспышки в виде рассеянного света зарниц, искрение близко расположенных, но не касающихся друг друга электрических проводов, голубоватое свечение внутренней поверхности стен домов, самопроизвольное загорание люминесцентных ламп незадолго до подземных толчков. Все эти признаки могут являться основанием для оповещения населения о возможном землетрясении".

Кнопка **По центру** располагает каждую строку абзаца или выделенного фрагмента текста по центру рабочего поля страницы, например:

"К предвестникам возможных землетрясений следует отнести следующие признаки, которые особенно должно знать население сейсмически опасных районов; это — появление запаха газа в районах, где до этого воздух был чист и

ранее подобное явление не отмечалось, беспокойство птиц и домашних животных, вспышки в виде рассеянного света зарниц, искрение близко расположенных, но не касающихся друг друга электрических проводов, голубоватое свечение внутренней поверхности стен домов, самопроизвольное загорание люминесцентных ламп незадолго до подземных толчков. Все эти признаки могут являться основанием для оповещения населения о возможном землетрясении".

Кнопка **По правому краю** выравнивает правый край текста. На сей раз в пострадавших ходит левый край:

"К предвестникам возможных землетрясений следует отнести следующие признаки, которые особенно должно знать население сейсмически опасных районов; это — появление запаха газа в районах, где до этого воздух был чист и ранее подобное явление не отмечалось, беспокойство птиц и домашних животных, вспышки в виде рассеянного света зарниц, искрение близко расположенных, но не касающихся друг друга электрических проводов, голубоватое свечение внутренней поверхности стен домов, самопроизвольное загорание люминесцентных ламп незадолго до подземных толчков. Все эти признаки могут являться основанием для оповещения населения о возможном землетрясении".

Ну и, наконец, кнопка **По ширине** выравнивает оба края за счет увеличения расстояния между словами:

"К предвестникам возможных землетрясений следует отнести следующие признаки, которые особенно должно знать население сейсмически опасных районов; это — появление запаха газа в районах, где до этого воздух был чист и ранее подобное явление не отмечалось, беспокойство птиц и домашних животных, вспышки в виде рассеянного света зарниц, искрение близко расположенных, но не касающихся друг друга электрических проводов, голубоватое свечение внутренней поверхности стен домов, самопроизвольное загорание люминесцентных ламп незадолго до подземных толчков. Все эти признаки могут являться основанием для оповещения населения о возможном землетрясении".

Чтобы максимально выровнять текст и уменьшить некрасивые пустые места между словами, возникающие при выравнивании по ширине, следует произвести *автоматическую расстановку переносов*. Это делается командой меню **Сервис | Язык | Расстановка переносов**.

В появившемся диалоговом окне необходимо выполнить следующие действия.

1. Установить флажок **Автоматическая расстановка переносов**.
2. Установить флажок **Переносы в словах из ПРОПИСНЫХ БУКВ**, если вы хотите, чтобы слова типа НАТО или ЮНЕСКО тоже могли бы быть перенесены.
3. Ширину зоны переноса слов рекомендуется сделать минимальной — 0,25 см (это способствует лучшему выравниванию правой границы текста, но увеличивает количество переносов).

4. Максимальное число последовательных переносов лучше ограничить двумя-тремя (большее количество уже считается дурным тоном).

Расстановка переносов применяется ко всему документу, а не только к выделенной его части.

Практикум 9. Абзацное выравнивание и расстановка переносов

1. Загрузите файл Zapiska.doc. Примените к тексту следующее оформление:
2. Часть текста до слов "Объяснительная записка" разместите по правому краю, размер 14. Выделите фамилию, имя и отчество директора полужирным курсивом.
3. Слова "Объяснительная записка" сделайте размером 18, полужирным подчеркнутым, разместите по центру.
4. Основной текст разместите по ширине, с расстановкой переносов, размером 14, курсивом.
5. Дату оставьте по левому краю, размером 12.
6. Сохраните файл под тем же именем.

Должно получиться так, как показано на рис. 5.10.

Директору школы №777

Крипаку Сергею Владимировичу

от ученика 10^{е} класса

Глебова Алексея

Объяснительная записка

Я, Глебов Алексей, опоздал сегодня на урок информатики, потому что всю ночь сидел в Интернете и активно чатился. Подхватил там несколько троянов и червей, вызывал доктора Касперского, с трудом выпечился. Больше не буду.

26 апреля 2002 года.

Рис. 5.10. Объяснительная записка-2

Изменение границ абзаца. Отступы. Межстрочные интервалы

Иногда в связи с выравниванием абзацев встает вопрос, а если я не желаю по центру, а хочу сдвинуть надпись на 2 см вправо, тогда как? И делают это невоспитанные люди с помощью пробелов. Мы же будем делать это культурно, сначала вручную, а потом с помощью команды меню **Формат | Абзац**.

Обратите внимание на горизонтальную линейку. Слева на ней есть вот такие маркеры:

- отступ первой строки — . Если захватить и перетащить его мышью, то во всех выделенных абзацах изменится отступ первой строки. Это часто применяется для создания красной строки;
- выступ — . Этот маркер сдвинет все строки абзаца, кроме первой.
- отступ слева — . Этот маркер сдвинет все строки.

Справа — лишь один маркер, отступ справа — . Он позволяет изменить правую границу всех строк абзаца или выделенной области.

Изменение границ абзаца относится, как всегда, к текущему абзацу или к выделенному фрагменту текста.

Все это допускается сделать и через меню **Формат | Абзац**. Здесь вы можете указать все отступы в сантиметрах, а, кроме того, установить *межстрочный интервал*. По умолчанию стоит одинарный интервал, но для дипломов и диссертаций требуется двойной (что не способствует экономии бумаги).

Практикум 10. Изменение границ абзаца и межстрочных интервалов

1. Откройте файл Zapiska.doc.
2. Сдвиньте правую границу всего текста на отметку 15 см (на горизонтальной линейке).
3. Текст до слов "Объяснительная записка" сделайте с полуторным интервалом.
4. В тексте объяснительной записки установите красную строку 1 см, а левую границу этого текста сдвиньте на отметку 2 см.
5. Текст объяснительной записки сделайте с двойным интервалом.
6. Дату сдвиньте вправо на отметку 4 см.
7. Сохраните документ под тем же именем.

Границы и заливка абзацев

Чтобы окончательно закончить с оформлением абзацев, рассмотрим их *обрамление и заливку*. Как всегда (не устаем напоминать) эти действия будут относиться к текущему абзацу или выделенному фрагменту текста.
Выберем команду меню **Формат | Границы и заливка**.

На вкладке **Граница** можно выбрать тип рамки, потом тип линии, ее цвет и ширину. Если нужна не полная рамка, а обрамление абзаца, скажем, только слева и справа или углом, то на рисунке в правой части вкладки надо щелкать левой кнопкой мыши на нужных линиях рамки — они будут исчезать или появляться. Кнопкой **Параметры** допускается задать отступ рамки от текста в типографских пунктах. После того как ваш образец начнет соответствовать задуманному, можно перейти к вкладке **Заливка**, где выбрать цвет заливки, а также тип узора и фона.

На вкладке **Страница** можно выбрать те же варианты оформления, что и на вкладке **Граница**, но только относиться они будут не к абзацу, а ко всей странице. Кроме того, странице можно задать обрамление рисунками из набора.

Вот примеры оформленных абзацев (рис. 5.11—5.13).

Чтобы окончательно закончить с оформлением абзацев, рассмотрим их обрамление и заливку. Как всегда (не устаю об этом напоминать) эти действия будут относиться к текущему абзацу или выделенному фрагменту текста.

Рис. 5.11. Абзац-1

Чтобы окончательно закончить с оформлением абзацев, рассмотрим их обрамление и заливку. Как всегда (не устаю об этом напоминать) эти действия будут относиться к текущему абзацу или выделенному фрагменту текста.

Рис. 5.12. Абзац-2

Чтобы окончательно закончить с оформлением абзацев, рассмотрим их обрамление и заливку. Как всегда (не устаю об этом напоминать) эти действия будут относиться к текущему абзацу или выделенному фрагменту текста.

Рис. 5.13. Абзац-3

Практикум 10. Обрамление и заливка абзацев

1. Откройте файл Newyear.doc.
2. Слова "Новогодняя песенка" разместите по центру, уменьшите соответственно левую и правую границы этого абзаца, задайте ему обрамление рамкой двойной линией с тенью. Заливка — светло-зеленым. Через меню **Формат | Шрифт** выберите анимацию **Фейерверк**.
3. Выделите куплеты песни. Сдвиньте левую границу на отметку 5 см, правую на 12 см. Задайте тексту обрамление слева и справа перевитой ленточкой. Заливка — светло-зеленым.
4. Задайте странице обрамление рисунком с елочками.
5. Разместите весь текст по центру страницы (команда меню **Файл | Параметры страницы | Макет | Вертикальное выравнивание | По центру**). Должно получиться так, как на приведенном фрагменте (рис. 5.14).
6. Сохраните файл под тем же именем.

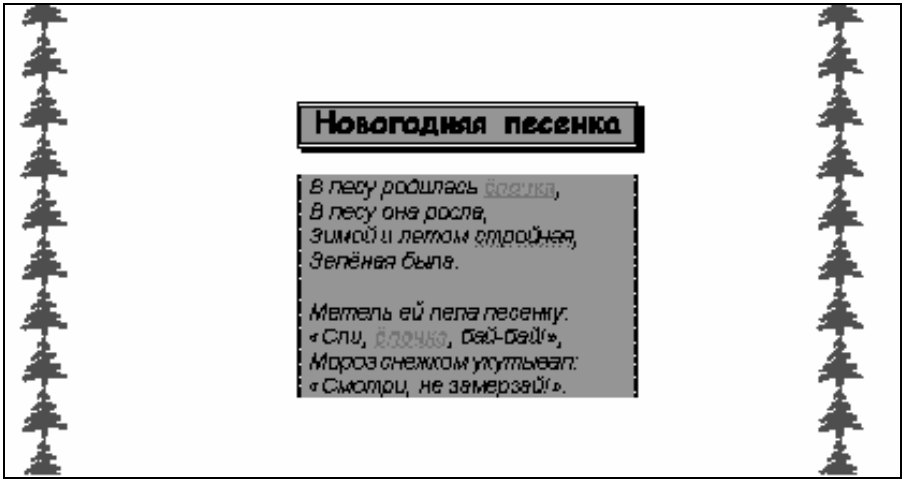


Рис. 5.14. Новогодняя песня-3

5.1.6. Оформление списков

Word позволяет оформлять списки двух основных форматов:

- *Маркированный список*. Абзацы маркируются специальными маркерами, вид которых и остальные настройки задаются командой меню **Формат | Список** или с помощью контекстного меню **Список**.

  Пример:

  Крупнейшими компаниями мира являются:

  - Microsoft.
  - Coca-Cola.
  - McDonalds.
  - Pepsi-Cola.
  - Sony.

- *Нумерованный список*. Абзацы нумеруются арабскими, римскими цифрами или буквами.

  Пример:
  Крупнейшими компаниями мира являются:

  1. Microsoft.
  2. Coca-Cola.
  3. McDonalds.
  4. Pepsi-Cola.
  5. Sony.

  или

  I. Microsoft.
  II. Coca-Cola.
  III. McDonalds.
  IV. Pepsi-Cola.
  V. Sony.

  или

  A. Microsoft.
  B. Coca-Cola.
  C. McDonalds.
  D. Pepsi-Cola.
  E. Sony.

  Возможно оформление и сложных, многоуровневых списков.

Практикум 11. Оформление списков

Дан следующий перечень:

Компоненты компьютера.

Системный блок.

Монитор.

Клавиатура.

Мышь.

Программное обеспечение.

Системные программы.

Языки программирования.

Прикладные программы.

Компьютерные сети.

Локальные сети.

Глобальные сети.

Оформите его тремя различными видами списков по нижеприведенной технологии.

1. *Оформление маркированным списком.*

1.1. Выделить весь первый список.

1.2. Выполнить команду: **Формат | Список | Маркированный** (Выбрать вариант маркера) | **Изменить | Шрифт | Размер** (выбрать **16**) | **Начертание** (выбрать **Полужирный**) | **OK | OK**.

1.3. Выделить первый подсписок.
Выполнить команду: **Формат | Список | Маркированный** (Выбрать вариант маркера) | **Изменить | Положение маркера | Отступ** (выбрать **0,5 см**) | **Положение текста | Отступ** (выбрать **1,1 см**) | **Шрифт | Размер | 12 | OK | OK**.

1.4. Повторить п. 1.3 для второго и третьего подсписков.

Должно получиться как на рис. 5.15.

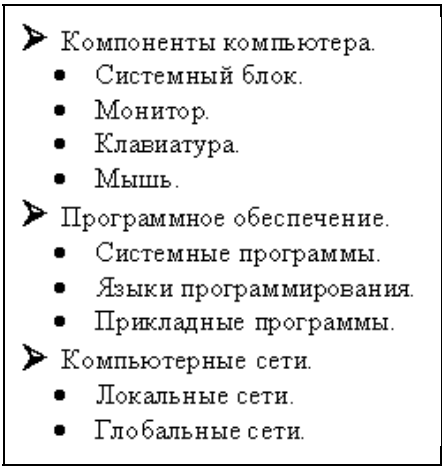


Рис. 5.15. Маркированный список

2. *Оформление нумерованным списком.*

2.1. Выделить строку "Компоненты компьютера".

2.2. На панели инструментов в окне **Стиль**, где видно слово **Обычный**, установить **Заголовок 1**.

2.3. Повторить п. 2.2 для строк "Программное обеспечение" и "Компьютерные сети".

2.4. Выделить все.

2.5. Выполнить команду: **Формат | Список | Нумерованный | Римские цифры | Изменить | Шрифт | Размер | 14 | Начертание | Полужирный | OK | OK**.

2.6. Выделить первый подсписок, на панели инструментов отжать кнопку **Нумерация**.

2.7. Повторить п. 2.6 для второго и третьего подсписков.

Должно получиться как на рис. 5.16.

I. Компоненты компьютера.

Системный блок.
Монитор.
Клавиатура.
Мышь.

II. Программное обеспечение.

Системные программы.
Языки программирования.
Прикладные программы.

III. Компьютерные сети.

Локальные сети.
Глобальные сети.

Рис. 5.16. Нумерованный список

3. *Оформление многоуровневым списком.*

3.1. Выделить строку "Компоненты компьютера".

3.2. На панели инструментов в окне **Стиль**, где видно слово **Обычный**, установить **Заголовок 1**.

3.3. Повторить п. 3.2 для строк "Программное обеспечение" и "Компьютерные сети".

3.4. Выделить первый подсписок. Установить для него опцию **Заголовок 2**.

3.5. Повторить п. 3.4 для второго и третьего подсписков.

3.6. Выполнить команду: **Формат | Список | Многоуровневый** (выбрать нужную маркировку) **| Изменить | Уровень 2 | Положение номера | По левому краю на 1 см | Положение текста | Отступ на 2,5 см | Шрифт | Размер | 12 | Начертание | Обычное**.

Должно получиться как на рис. 5.17.

1 Компоненты компьютера.

1.1 *Системный блок.*

1.2 *Монитор.*

1.3 *Клавиатура.*

1.4 *Мышь.*

2 Программное обеспечение.

2.1 *Системные программы.*

2.2 *Языки программирования.*

2.3 *Прикладные программы.*

3 Компьютерные сети.

3.1 *Локальные сети.*

3.2 *Глобальные сети.*

Рис. 5.17. Многоуровневый список

Контрольные вопросы по темам "Набор и редактирование текста", "Шрифтовое и абзацное оформление"

1. Как компьютер различает слова?
2. А как абзацы?
3. Какие минимальные и максимальные размеры шрифта разрешает задавать Microsoft Word?
4. Какой символ будет удален при нажатии клавиши <Delete> в слове Сфинкс, если курсор установлен между буквами "ф" и "и"?
5. Какой символ будет удален при нажатии клавиши <Backspace> в слове Сфинкс, если курсор установлен между буквами "н" и "к"?
6. Что произойдет, если при курсоре, установленном в начало строки, нажать клавишу <Enter>?
7. Что произойдет, если при курсоре, установленном в середину набранной строки, нажать клавишу <Enter>?
8. Какую клавишу надо нажать, чтобы получать только заглавные буквы?

9. Как правильно ставить пробелы около знаков препинания?
10. Какая разница существует при вводе знаков "дефис" и "тире"?
11. Какие правила существуют при заключении фрагментов текста в кавычки и скобки?
12. Какие ориентации страницы вы знаете? Как изменить ориентацию страницы?
13. Какие действия Word позволяет производить над шрифтами?
14. Как единовременно удалить фрагмент текста?
15. Как установить красную строку?
16. Как изменить левую и правую границы абзаца?
17. Как компьютер проверяет орфографию?
18. Как заставить Word автоматически расставлять переносы?
19. В чем отличие команд **Сохранить** и **Сохранить как**?
20. Можно ли создать папку непосредственно в программе Word?
21. Какие типы выравнивания вы знаете?
22. Какие виды списков вы знаете?
23. Можно ли изменить маркер для маркированного списка и как это сделать?

Зачетное задание по темам "Набор и редактирование текста", "Шрифтовое и абзацное оформление"

1. Набрать следующий текст (не разделяя его предварительно на строки). Сохранить под именем Before.doc (или загрузить уже подготовленный файл, если это возможно):

   В одной старой-старой стране, в старом-старом городе, в старом-старом замке, в старой-старой комнате сидит, окутанный старой-старой паутиной, старый-старый скелет. Перед ним монитор, а на мониторе надпись: Windows is now loading. Please wait! Хирург разглядывает рентгеновский снимок: "Да-а-а! Ключица поломана, два ребра сломаны, трещина в малой берцовой кости. Ну ничего, в Photoshope все исправим!" — У вас какой бензин? — Девяносто пятый. — А нет девяносто восьмого или двухтысячного? — Чем компьютер отличается от арабской семьи? — В арабской семье — один папа и несколько мамок, а в компьютере — одна мама, но МНОГО, ОЧЕНЬ МНОГО папок!!!

2. Оформить введенный текст следующим образом:
   - Предварить текст заголовком "Компьютерные анекдоты", расположив заголовок по центру, размер 18, полужирный курсив, гарнитура Monotype Corsiva. Подчеркнуть штрих-пунктирной линией, красным цветом.
   - Через строчку написать "Подобрал ученик 10в класса", и на следующей строке "Николай Барахтин". Шрифт 13, курсив, гарнитура Arial, по правому краю.

- Основной текст: шрифт 12, гарнитуры Courier New. Выравнивание по ширине. Красная строка 0,7 см. Разбить текст на четыре анекдота, которым последовательно задать обрамление и расположить, как показано. Применить к каждому анекдоту буквицу. Залить разными цветами.
- Выполнить автоматическую расстановку переносов.
- Сохранить документ Before.doc под именем Anekdot.doc.

Должно получиться так, как показано на рис. 5.18.

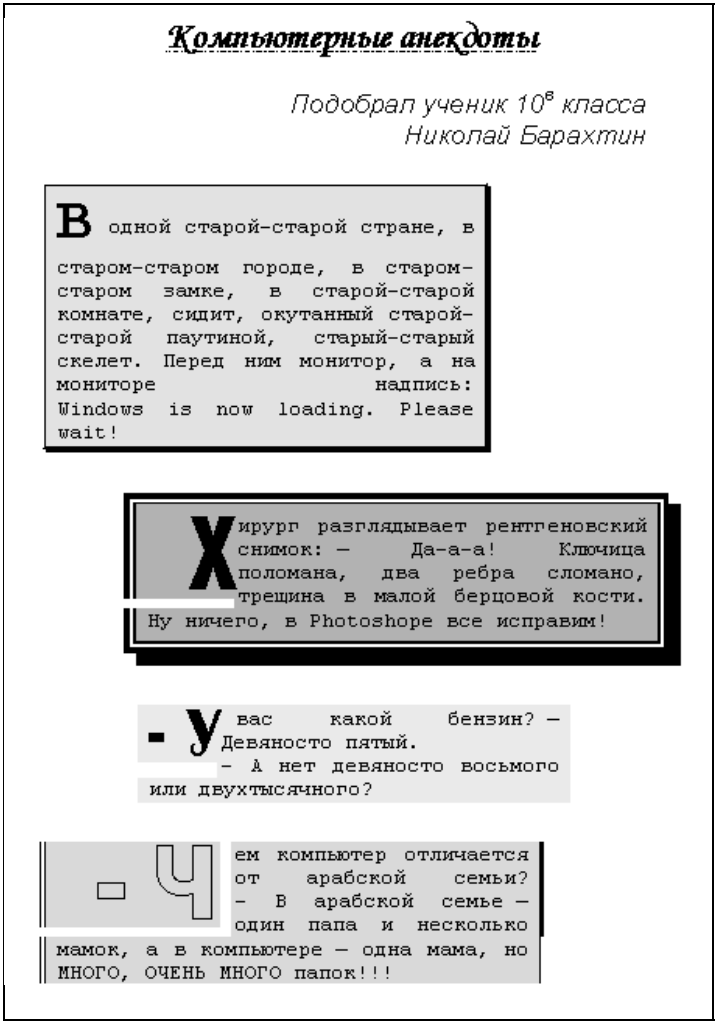


Рис. 5.18. Компьютерные анекдоты

5.1.7. Настройка абзацев по образцу

Иногда может существенно помочь *настройка абзацев по образцу*. Настроив соответствующим образом (так как нужно) один абзац, мы, по мановению волшебной мышки, настроим и все остальные. Для этого на панели инструментов присутствует кнопка **Формат по образцу** . Работа с ней ведется следующим образом:

1. Настроить абзац, который будет служить образцом.
2. Щелкнуть двойным щелчком левой кнопкой мыши по кнопке .
3. Установить указатель мыши (который должен принять вид веника рядом со стандартным курсором I) внутрь абзаца.
4. Щелкнуть внутри настраиваемого абзаца левой кнопкой мыши.
5. Повторить предыдущую операцию требуемое количество раз.
6. Отключить кнопку **Формат по образцу** повторным щелчком левой кнопкой мыши.

Практикум 12. Настройка абзацев по образцу

1. Откройте файл Zemletr.doc.
2. Настройте средний абзац следующим образом: шрифт Arial, размер 12, курсив. Красная строка 1 см. Правая граница абзаца на отметке 15. Обрамление справа и слева двойными линиями. Выравнивание по ширине. Заливка серым узором, 15%.
3. Настройте по этому образцу первый и второй абзацы.
4. Ниже допишите свои имя, фамилию, класс и дату исполнения.
5. Сохраните документ под именем Obrazets.doc.

5.1.8. Копирование и перенос фрагментов текста, одновременная работа с несколькими окнами

Бывает, что какая-то часть документа в том же самом или слегка измененном виде должна повториться в этом же или другом документе. Для Word и эта задача легка!

Последовательность действий всегда одна и та же.

1. Сначала выделяем фрагмент (выделять мы уже умеем).
2. Потом выполняем команду **Копировать** (или **Вырезать**).

3. Устанавливаем курсор в место вставки.
4. Выполняем команду **Вставить**.
   - Команда **Копировать** позволяет сохранить фрагмент текста на старом месте и поместить его копию в новое место.
   - Команда **Вырезать** переносит фрагмент со старого места на новое.

Команды **Копировать** или **Вырезать** можно выполнить одним из трех доступных способов.

- ❒ Через меню **Правка**.
- ❒ Через панель инструментов: кнопка **Копировать** , кнопка **Вырезать** .
- ❒ С помощью так называемого *контекстного меню*. Если щелкнуть правой копкой мыши на выделенном фрагменте, то вызовется контекстное меню, в котором левой кнопкой можно выбрать соответствующие команды.

Курсор установить в нужное место мы уже умеем.

Выполнить команду **Вставить** можно теми же способами, что и **Копировать**, а кнопка для нее выглядит так — .

Если же фрагмент копируется (перемещается) на небольшое расстояние, это можно сделать так: непосредственно навести указатель мыши на выделенный фрагмент, нажать левую кнопку мыши и, удерживая ее, перетащить фрагмент на новое место. Указывать на место, куда фрагмент вставится, будет вертикальная серая линия. Следите за ней! Если же во время этой операции еще удерживать и клавишу <Ctrl>, то осуществится копирование. Обе эти операции требуют определенной сноровки, и если вы не очень уверены в своих силах, пользуйтесь другими способами.

Если же необходимо скопировать (переместить) фрагмент в другой документ, то все выполняется так же, как описано выше с помощью кнопок **Копировать** или **Вырезать**, только на третьем шаге копирования надо будет перейти к этому документу через меню **Окно**.

Практикум 13. Копирование и перемещение фрагментов текста

1. Откройте уже подготовленный файл Putanica.doc.
2. Используя копирование и перемещение фрагментов, расположите куплеты и припевы песни в следующей последовательности: 1 куплет, Припев, 2 и 3 куплеты, Припев, 4 куплет, Припев, 5 куплет.
3. Уберите номера куплетов и слово Припев. Расположите название песни по центру, размером 14, полужирным курсивом. Авторов — размером 11, курсивом, по правому краю. Основной текст — размером 12, отступ всех строк на 2 см слева.

4. Сохраните файл под именем Nautilus.doc.

"Я хочу быть с тобой"

Вячеслав Бутусов и "Наутилус Помпилиус"

5. в комнате с белым потолком
с правом на надежду
в комнате с видом на огни
с верою в любовь

3. твое имя давно стало другим
глаза навсегда потеряли свой цвет
пьяный врач мне сказал, тебя больше нет
пожарный выдал мне справку, что дом твой сгорел

Припев:

я хочу быть с тобой
я хочу быть с тобой
я так хочу быть с тобой
я хочу быть с тобой
и я буду с тобой

1. я пытался уйти от любви
я брал острую бритву и правил себя
я укрылся в подвале я резал
кожаные ремни, стянувшие слабую грудь

4. я ломал стекло как шоколад в руке
я резал эти пальцы за то что они
не могут прикоснуться к тебе
я смотрел в эти лица и не мог им простить
того что у них нет тебя и они могут жить

2. в комнате с белым потолком
с правом на надежду
в комнате с видом на огни
с верою в любовь

Практикум 14. Копирование и перемещение фрагментов текста при работе с несколькими документами

1. Откройте последовательно файлы Zagadka1.doc, Zagadka2.doc, Otgadka1.doc и Otgadka2.doc.
2. Создайте новый документ.
3. Скопируйте в него тексты из открытых файлов следующим образом: первая загадка, вопрос "Отгадай, что это?", отгадка, вторая загадка, вопрос, отгадка.
4. Сохраните полученный файл под именем Zagadki.doc.

5.1.9. Автозамена

Автозамена — очень удобный инструмент при наборе типовых документов с часто повторяющимися словами или словосочетаниями. Вызывается командой меню **Сервис | Автозамена** и позволяет на любой символ (букву, буквосочетание, знак) запрограммировать слово или предложение, которыми этот символ будет заменяться после его набора и нажатия клавиши <Пробел>. Например, если на буквы "рф" задать название "Российская Федерация", то при наборе в тексте рф<Пробел> мы сразу получим Российская Федерация. Удобно! Попробуем.

Практикум 15. Автозамена

1. Задайте для автозамены следующие сокращения:
   - п — Пифагор;
   - пр — прямоугольны треугольник;
   - ка — катет;
   - г — гипотенуз;
   - т — теорем;
   - су — Сумма квадратов катетов равна квадрату гипотенузы.
2. Используя автозамену, наберите следующий текст:

   Теорема Пифагора.

   Пифагор — древнегреческий ученый, который первым доказал, что "сумма квадратов катетов равна квадрату гипотенузы". Пифагор рассмотрел внимательно прямоугольный треугольник и увидел, что у него есть катеты и гипотенуза. Он выпилил несколько фанерных прямоугольных треугольников, произвел измерения катетов и гипотенуз и гениально доказал свою теорему, которая гласит:

   "Сумма квадратов катетов равна квадрату гипотенузы".

   Пифагор с тех пор пользуется необычайной популярностью у школьников, прямоугольные треугольники живут среди людей, а теорема Пифагора в народе звучит так "Пифагоровы штаны во все стороны равны".
3. Расположите заголовок по центру, размер 14, полужирный курсив. Основной текст — размер 12, выравнивание по ширине, автоматическая расстановка переносов, красная строка 1 см.
4. Сохраните под именем Pifagor.doc.

5.1.10. Форматирование текста в колонки

Для придания тексту большей наглядности или при верстке печатных изданий (школьных или личных) применяется *размещение текста в две или несколько колонок*. Чаще всего это делается самым простым способом — выделяется фрагмент текста, который должен быть размещен в колонки, на панели инструментов нажимается кнопка **Колонки** и мышью указывается

количество колонок. Текст после этого равномерно размещается в указанное количество колонок. Правда таким образом можно получить колонки только одинаковой ширины. Если же надо расположить текст в колонки разной ширины, то придется воспользоваться меню **Формат | Колонки**. Рекомендуется следить, чтобы после выделения текста сверху и снизу от него был либо другой текст, либо символы конца абзаца.

Практикум 16. Размещение текста в колонки

Задание 1.

1. Открыть файл Nautilus.doc. Под текстом песни дописать фразу: `Главный располагатель текста в колонки Антон Мирошниченко`. Задать размер шрифта 12, выравнивание по правому краю. Расположить текст песни в две одинаковые колонки. Добиться того, чтобы куплеты или припевы не были разбиты посередине.
2. Сохранить документ под тем же именем.

Должно получиться как на рис. 5.19.

"Я хочу быть с тобой..."

Вячеслав Бутусов и "Наутилус Помпилиус"

я пытался уйти от любви
я брал острую бритву и правил себя
я укрылся в подвале я резал
кожаные ремни, стянувшие слабую
грудь

я хочу быть с тобой
я хочу быть с тобой
я так хочу быть с тобой
я хочу быть с тобой
и я буду с тобой

в комнате с белым потолком
с правом на надежду
в комнате с видом на огни
с верою в любовь

твое имя давно стало другим
глаза навсегда потеряли свой цвет
пьяный врач мне сказал, тебя больше
нет
пожарный выдал мне справку, что
дом твой сгорел

я хочу быть с тобой
я хочу быть с тобой
я так хочу быть с тобой
я хочу быть с тобой
и я буду с тобой

я ломал стекло как шоколад в руке
я резал эти пальцы за то что они
не могут прикоснуться к тебе
я смотрел в эти лица и не мог им
простить
того что у них нет тебя и они могут
жить

я хочу быть с тобой
я хочу быть с тобой
я так хочу быть с тобой
я хочу быть с тобой
и я буду с тобой

в комнате с белым потолком
с правом на надежду
в комнате с видом на огни
с верою в любовь

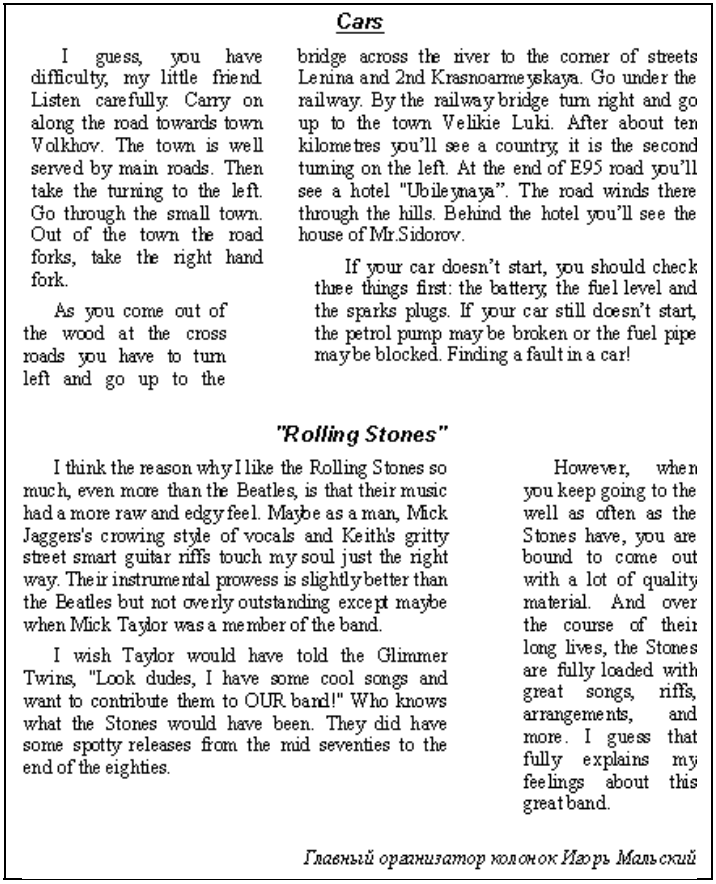
Главный располагатель текста в колонки Антон Мирошниченко

Рис. 5.19. "Я хочу быть с тобой..."

Задание 2.

1. Открыть файлы Cars.doc и Stones.doc. Вставить содержимое Stones.doc после текста в Cars.doc. Предварить текст заголовком "Cars". В конце всего текста набрать фразу: `Главный организатор колонок Игорь Мальский`. Применить следующее шрифтовое и абзацное оформление. Заголовки — размер 14, полужирный курсив с подчеркиванием, Arial, по центру. Подпись — размер 12, курсив, по правому краю. Основной текст — размер 12, по ширине, красная строка 1 см.
2. Расположить первый текст в две колонки (вариант "Слева") и второй текст — в две колонки, но вариант "Справа".
3. Сохранить документ под именем Kolonki.doc

У вас должно получиться в результате вот так (рис. 5.20).



Cars

I guess, you have difficulty, my little friend. Listen carefully. Carry on along the road towards town Volkhov. The town is well served by main roads. Then take the turning to the left. Go through the small town. Out of the town the road forks, take the right hand fork.

As you come out of the wood at the cross roads you have to turn left and go up to the bridge across the river to the corner of streets Lenina and 2nd Krasnoarmeyskaya. Go under the railway. By the railway bridge turn right and go up to the town Velikie Luki. After about ten kilometres you'll see a country, it is the second turning on the left. At the end of E95 road you'll see a hotel "Ubileynaya". The road winds there through the hills. Behind the hotel you'll see the house of Mr.Sidorov.

If your car doesn't start, you should check three things first: the battery, the fuel level and the sparks plugs. If your car still doesn't start, the petrol pump may be broken or the fuel pipe may be blocked. Finding a fault in a car!

"Rolling Stones"

I think the reason why I like the Rolling Stones so much, even more than the Beatles, is that their music had a more raw and edgy feel. Maybe as a man, Mick Jaggers's crowing style of vocals and Keith's gritty street smart guitar riffs touch my soul just the right way. Their instrumental prowess is slightly better than the Beatles but not overly outstanding except maybe when Mick Taylor was a member of the band.

I wish Taylor would have told the Glimmer Twins, "Look dudes, I have some cool songs and want to contribute them to OUR band!" Who knows what the Stones would have been. They did have some spotty releases from the mid seventies to the end of the eighties.

However, when you keep going to the well as often as the Stones have, you are bound to come out with a lot of quality material. And over the course of their long lives, the Stones are fully loaded with great songs, riffs, arrangements, and more. I guess that fully explains my feelings about this great band.

Главный организатор колонок Игорь Мальский

Рис. 5.20. Расположение текста в колонки

5.1.11. Работа с таблицами

Таблицы, позволяющие обрабатывать большие объемы числовых данных, следует создавать в Microsoft Excel. А вот таблицы, содержащие какую-нибудь справочную информацию, списки и т. п., мы будем делать в Word. С таблицами можно выполнять следующие действия:

- изменять ширину столбцов и высоту строк;
- вставлять и удалять столбцы и строки;
- сортировать данные по возрастанию и убыванию;
- применять к данным шрифтовое и цветовое оформление;
- наносить сетку

и т. д.

Вставить таблицу можно тремя способами.

- Нажать мышью на панели инструментов кнопку **Добавить таблицу** и выделить необходимое количество столбцов и строк.
- Выполнить команду меню **Таблица | Добавить | Таблица** и задать необходимое количество столбцов и строк.
- Выполнить команду меню **Таблица | Нарисовать таблицу** и нарисовать нужную таблицу, используя подходящие инструменты.

Ячейка таблицы выделяется одинарным щелчком левой кнопкой мыши в левой части ячейки, строка — так же, как и строка текста, а столбец — щелчком левой кнопкой мыши на его верхней границе.

Практикум 17. Изготовление визитки

Стандартная визитка имеет размеры 9×5 см и содержит обычно следующие сведения: учреждение, фамилию, имя, отчество, должность, адрес, телефон, факс, E-mail. Сейчас мы сделаем простой вариант визитки, без логотипа или иного графического объекта.

1. Настройте параметры страницы так, чтобы левое и правое поля были по 1,5 см, что даст нам ширину рабочего поля 18 см — как раз на 2 визитки.
2. Вставьте таблицу — 5 строк и 2 столбца (на странице разместится 10 визиток).
3. Настройте высоту строк по 5 см (командой меню **Таблица | Свойства таблицы**).
4. Занесите в первую ячейку данные о себе по следующему образцу (рис. 5.21).
5. Задайте элементам визитки следующее оформление: название школы — размер 14, полужирный, по центру; свои фамилию, имя, отчество — размер 12, курсив; должность — размер 10, полужирный, по центру; адреса и средства связи — размер 10, по правому краю. Значки конверта и телефона — в гарнитуре Wingdings соответственно <Shift>+8 и <Shift>+9.

Средняя школа №777

Скипский
Михаил Игоревич

Ученик 12[а] класса

✉ 198270 СПб, ул. им. Собаки Павлова, 45
☎ (812) 777-13-13 fax (812) 1313-777
E-mail dog@pavlov.spb.ru

Рис. 5.21. Визитка-1

6. После оформления скопируйте ячейку таблицы с визиткой в буфер обмена и вставьте ее в остальные девять ячеек. У вас получится 10 замечательных визиток. Можно печатать их на хорошем принтере и бумаге и с деловым видом вручать своим друзьям и подругам.
7. Сохраните вашу работу под именем Vizitka1.doc.

Практикум 18. Форматирование таблиц. Сортировка данных

Вариант 1.

1. Вставьте, заполните и оформите таблицу следующего вида (рис. 5.22). Используйте команды меню **Таблица | Объединить ячейки** и **Формат | Границы и Заливка**. При оформлении обратите внимание на использование надстрочных и подстрочных символов, двойное обрамление всей таблицы, волнистое подчеркивание. Название таблицы — 16, полужирный курсив, шапка и основной текст — размер шрифта 14.

   Так как наша книга черно-белая, а возможности Word и вашего монитора, надеемся, многоцветные, то цветовое решение таблицы придется объяснить словами: шапка таблицы заполнена светло-серым; в столбце "Цвет" слово "Красный" — красного цвета на желтом фоне, слово "Желтый" — желтого цвета на черном фоне, "Зеленый" — зеленого цвета на сером фоне и "Синий" — синего на голубом.
2. Отсортируйте строки таблицы:
   - по столбцу "Цвет";
   - по столбцу "Длина волны".
3. Вставьте следующую строку в конец таблицы (Слово "Фиолетовый" — белыми буквами на фиолетовом фоне):

392,0	**Фиолетовый**	1,4844	1,891	1,7442	1,4235

4. Отсортируйте строки таблицы по столбцу "Цвет" еще раз.
5. Сохраните полученную таблицу под именем Table1.doc.

Физические параметры различных длин волн

Длина Волны нм	Цвет	Среда			
		Стекло		Вода	Каменная соль
		Тяжёлый флинт	Лёгкий крон	Темп. 20°С	
656,3	Красный	1,6444	1,5145	1,3311	1,5407
589,3	Жёлтый	1,6499	1,5170	1,3330	1,5443
546,1	Зелёный	1,6546	1,5191	1,3345	1,5475
480,0	Синий	1,6648	1,5235	1,3374	1,5665

Автор: Сидоров Иван, 10а класс.

Рис. 5.22. Физические параметры различных длин волн

Вариант 2.

1. Вставьте, заполните и оформите таблицу следующего вида (рис. 5.23). Используйте команды меню **Таблица | Объединить ячейки** и **Формат | Границы и Заливка**. Требования по шрифтовому оформлению — как в варианте 1.

   Цветовое оформление: шапка залита серым, столбец "ВОДА" — голубым, УГЛЕВ. — желтым, БЕЛКИ — салатным.
2. Отсортируйте строки таблицы:
   - по столбцу "ОВОЩИ, ПЛОДЫ и ЯГОДЫ";
   - по столбцу "Кислотность, %".
3. Вставьте после строки с цветной капустой следующую строку:

Хрен огородный	66,6	8,7	4,5	50,0	6,6

4. Отсортируйте строки таблицы по столбцу "ОВОЩИ, ПЛОДЫ и ЯГОДЫ" еще раз.
5. Слева от столбца с кислотностью вставьте столбец "Содержание железа в %". Цифры расставьте произвольно (в пределах от 0,02 до 0,09).
6. Сохраните таблицу под именем Table2.doc.

Химический состав и калорийность плодов, овощей и ягод

ОВОЩИ, ПЛОДЫ и ЯГОДЫ	Состав съедобной части, %			Калорий-ность, ккал/100г	Кислот-ность, %
	ВОДА	УГЛЕВ.	БЕЛКИ		
Картофель	78,0	21,0	2,0	91,0	5,5
Капуста Белокочанная	90,0	5,0	1,8	29,1	6,1
Капуста цветная	91,0	5,0	2,8	30,5	5,8
Свекла столовая	86,5	10,0	1,5	48,0	5,2
Морковь	88,0	8,7	1,3	41,0	5,1

Рис. 5.23. Химический состав и калорийность плодов, овощей и ягод

Вариант 3.

1. Вставьте, заполните и оформите таблицу следующего вида (рис. 5.24). Используйте команды меню **Таблица | Объединить ячейки** и **Формат | Границы и Заливка**. Требования по шрифтовому оформлению — как в варианте 1.

   Цветовое оформление: ячейки с адресами магазинов и реклама про широкий выбор мониторов — заливка серым, столбцы "Компьютеры IKS" — заливка оранжевым, "Компьютеры Zet" — заливка желтым.

2. Отсортируйте строки таблицы:
   - по столбцу "Системный блок";
   - по столбцу "Компьютеры IKS".
3. После строки с Intel® Pentium® 4 1700 вставьте еще 3 строки и заполните их аналогичными данными.
4. Перед адресами магазинов вставьте символ конверта, а перед телефонами — символ телефона *(см. Практикум 17)*.
5. Отсортируйте строки таблицы по столбцу "Системный блок" еще раз.
6. Сохраните таблицу под именем Table3.doc.

Компьютерный магазин IKS

Парковая пл., 13, 890-45-32 **Весенняя ул., 21, 823-14-12**	**Футбольный пр., 11, 780-17-65** **Чемпионский пер., 2, 863-29-27**	
www.iks.ru	Курс Нал.=31.89	Курс Б/нал.=33.49
Системный блок Корпус, дисковод, видео Мат.плата, процессор, память	Компьютеры IKS Стоимость программного обеспечения не входит в цену	Компьютеры Zet В комплекте Windows'2099 Гарантия 2 года
Intel® Pentium® 4 1500	408	509
Intel® Pentium® 4 2200	715	816
Intel® Pentium® 4 1700	435	536
Всегда широкий выбор мониторов — более 100 моделей!		

Рис. 5.24. Компьютерный магазин IKS

Практикум 19. Оформление бланков

1. Оформите бланк заявления по следующему образцу (рис. 5.25).

   Для оформления шапки заявления и подписи используйте таблицы из трех столбцов. На образце границы этих таблиц прочерчены, вам предстоит их, конечно, убрать.

   Шрифтовое оформление:

   - символ 🐜 найти и вставить из гарнитуры Webdings, размер 50;
   - шрифт надписи про Урюпинский ГДТТ — гарнитура Arial, размер 14, полужирный;
   - шрифт надписи "Директору" и подпись — гарнитура Times New Roman, размер 14, полужирный курсив;
   - основной текст — гарнитура Times New Roman, размер 14.

   Абзацное оформление: основной текст — абзацный отступ 1 см, выравнивание по ширине. Дата — выравнивание по левому краю. "Директору" и Мышкин — по правому краю.

2. Сохраните документ под именем Zoo.doc.

УРЮПИНСКИЙ ГОРОДСКОЙ ДВОРЕЦ ДЕТСКОГО ТВОРЧЕСТВА СТАНЦИЯ ЮНЫХ НАТУРАЛИСТОВ _____ № _____ "___" _____ 200 _ г.		*Директору Нью-Васюковского зоопарка господину Звереву З.А.*

Заявка.

Прошу выделить 2 (двух) слонов для постоянного проживания в живом уголке специальной школы с углублённым изучением биологии №13.

Сохранность хоботов и бивней гарантируем.

Главный любитель животных		*Мышкин А.И.*

31 февраля 2002 г. от Р.Х.

Рис. 5.25. "Зверская" заявка

Практикум 20. Автоформат

1. Загрузите файл Table2.doc. В меню **Таблица** выберите команду **Автоформат**. Просмотрите возможные варианты. Выберите наиболее, на ваш взгляд, подходящий. Обратите внимание на изменившийся вид таблицы. Воспользуйтесь в меню **Таблица** командой **Автоподбор | Автоподбор по содержимому**.
2. Сохраните документ под тем же именем.

Практикум 21. Использование формул

Для того чтобы в таблицах Word можно было использовать *формулы подсчета*, например суммы, среднего значения, нахождения минимального или максимального значения и т. п., ее ячейкам назначаются адреса, формируемые следующим образом: строки нумеруются натуральными числами, начиная с единицы, а столбцы — буквами латинского алфавита, начиная с А.

Ячейка находится на пересечении столбца и строки и получает адрес как в игре "Морской бой", например A2, C4 или D5.

Используя команду меню **Таблица | Формула**, можно вставить стандартную функцию или вычислить свою.

1. Создайте таблицу размером 4 столбца на 11 строк, в которой будут содержаться следующие данные. В первом столбце "Номер по порядку", во втором — "Фамилия", в третьем — "Рост (см)", в четвертом — "Вес (кг)". Заполните таблицу данными десяти учащихся своего класса (рис. 5.26).

№	Фамилия	Рост (см)	Вес (кг)
1	Малафеев	175	80
2	Желудков	188	78
3	Аршавин	176	79
4	Морозов	168	82
5	Бирюков	186	90
6	Кержаков	180	72
7	Овсепян	172	74
8	Казаченок	190	88
9	Бышовец	174	76
10	Садырин	170	68

Рис. 5.26. Друзья-товарищи

2. Вставьте в полученную таблицу еще 4 строки после всех данных. Вычислите суммарный рост и вес всех десяти человек, средний рост и вес, а также максимальный и минимальный рост и вес. Должно получиться как на рис. 5.27.

   Так, например, чтобы получить сумму всех ростов, необходимо установить курсор в ту ячейку, в которой мы хотим получить результат, выбрать в меню **Таблица | Формула** функцию **Sum()** и написать в скобках интервал вычисляемых ячеек **C2:C11**. Компьютер просуммирует все числа, начиная с **C2** и заканчивая **C11**. Остальное — аналогично.

3. Сохраните документ под именем Rost_ves.doc.

№	Фамилия	Рост (см)	Вес (кг)
1	Малафеев	175	80
2	Желудков	188	78
3	Аршавин	176	79
4	Морозов	168	82
5	Бирюков	186	90
6	Кержаков	180	72
7	Овсепян	172	74
8	Казаченок	190	88
9	Бышовец	174	76
10	Садырин	170	68
	Суммарный	1779	787
	Средний	177.9	78.7
	Минимальный	168	68
	Максимальный	190	90

Рис. 5.27. Сосчитай-ка!

5.1.12. Колонтитулы

Колонтитул — слово немецкое, обозначающее надпись, помещаемую на каждой странице документа над или под текстом и заключающую в себе фамилию автора, название книги или какой-либо ее части. В Word это понятие более широко и очень полезно в использовании. Для установки колонтитулов необходимо воспользоваться командой меню **Вид | Колонтитулы**. Появится вот такая картинка (рис. 5.28).

Рис. 5.28. Колонтитулы

Прежде чем выполнить маленькое упражнение, рассмотрим содержимое этого окна. Если выбрать команду **Вставить автотекст**, то мы увидим следующее (рис. 5.29).

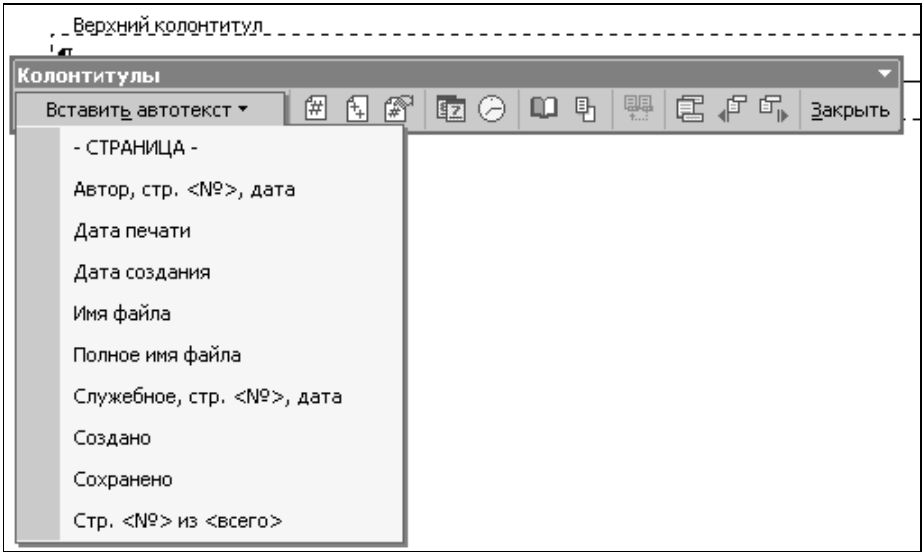


Рис. 5.29. Автотекст

Тут все по-русски и должно быть понятно.

Замечание

Очень полезно вставлять в печатаемые документы учащихся полное имя файла, дату и фамилию — во избежание дальнейших недоразумений, чья это работа, и откуда она взялась.

Со всеми остальными кнопками предлагается разобраться самостоятельно. Форматирование текста колонтитула производится так же, как и любого другого текста.

Практикум 22. Колонтитулы

1. Загрузите файл Parametr.doc. Оформите колонтитулы следующим образом. Верхний колонтитул должен содержать полное имя файла (размер шрифта 8, выравнивание по левому краю страницы) и имя автора (размер шрифта 10, полужирный курсив, по правому краю страницы). Нижний колонтитул должен содержать дату (размер 8, по левому краю страницы) и номер страницы в формате стр. № (шрифт 10, по правому краю страницы).
2. Сохраните документ под тем же именем.

5.1.13. Поиск и замена элементов текста

Если документ по объему большой, то иногда возникает необходимость нахождения в нем какого-либо ключевого слова (фамилии или термина) во всем тексте или замены одних элементов на другие.

Делается это командами меню **Правка | Найти** (рис. 5.30) или **Правка | Заменить** (рис. 5.31).

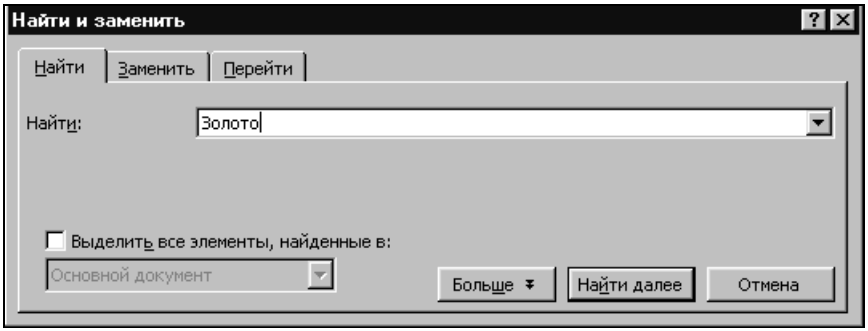


Рис. 5.30. Поиск в тексте

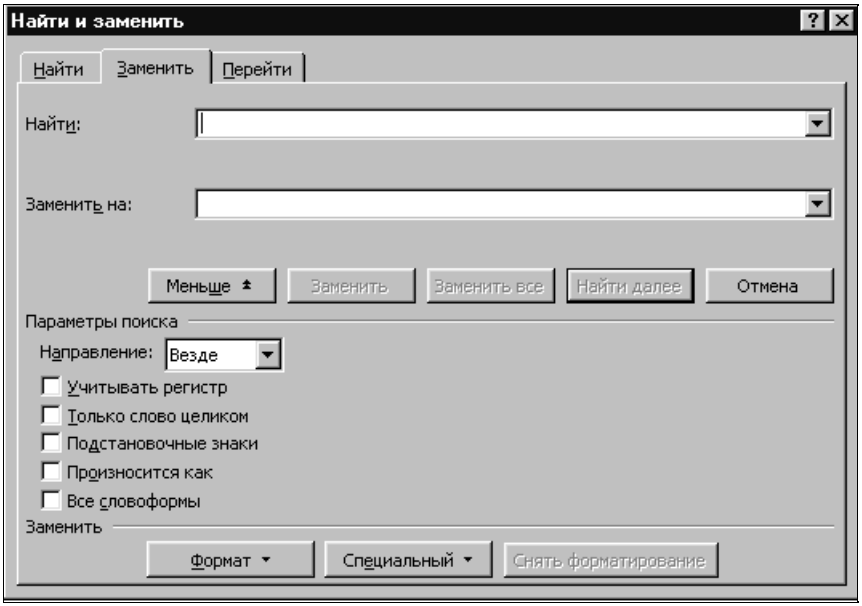


Рис. 5.31. Замена в тексте

Практикум 23. Поиск и замена в тексте

1. Загрузите файл Parametr.doc. При помощи функции поиска найдите в тексте слово "Будюм".
2. Получилось? А теперь при помощи функции замены замените все буквосочетания "машин" на "ДУБИН". Посмотрите, что изменилось в тексте.

Более обижать наши любимые компьютеры не будем, поэтому текст в таком виде не сохраним. Однако замену делать научились.

5.1.14. Написание математических формул

Когда в *главе 4* мы разбирались с арифметикой и алгеброй в программировании, нам приходилось записывать алгебраические выражения в виде, пригодном для решения компьютером. Но в тексте контрольной работы, диплома, реферата или диссертации, выполняемых на компьютере, эти самые математические формулы приходится писать довольно часто. Раньше приходилось печатать на компьютере текст, а формулы потом вписывать от руки. Теперь в Word включен очень удобный инструмент написания формул — Microsoft Equation 3.0.

Чтобы воспользоваться этим инструментом необходимо выбрать команду меню **Вставка | Объект** (рис. 5.32).

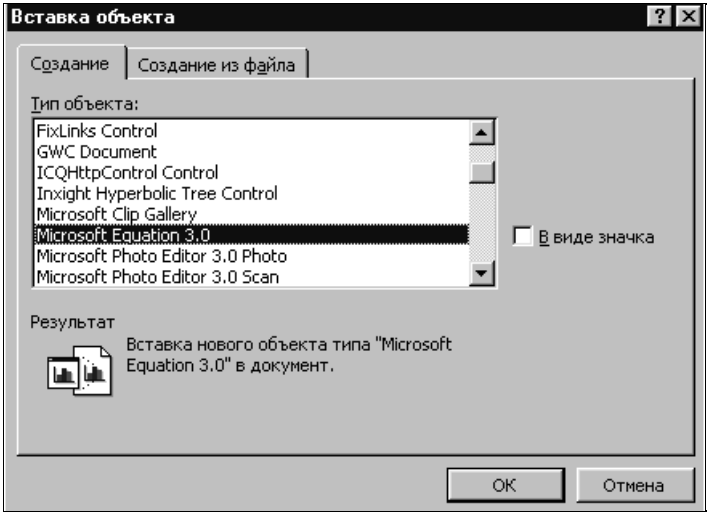


Рис. 5.32. Вставка объекта

Появится панель инструментов Microsoft Equation 3.0 (рис. 5.33).

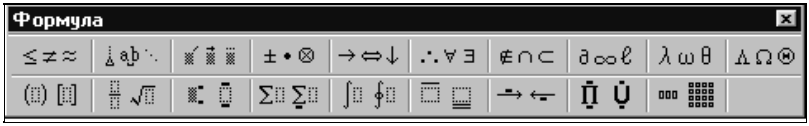


Рис. 5.33. Панель инструментов Microsoft Equation 3.0

Практикум 24. Написание математических формул

Используя Microsoft Equation 3.0, напишите следующие формулы:

1. $\frac{1}{5}+\frac{1}{6}+\frac{1}{7}+\frac{1}{8}+\frac{1}{9}$
2. $2^2+3^2+4^2+5^2+6^2$
3. $\frac{1+2}{3+4}:\frac{5+6}{7+8}$
4. $\frac{2^3+3^2+4^3}{9^3+5\times6\times7}$
5. $10\times\sqrt{25+11}$
6. $\cfrac{1}{2+\cfrac{3}{4+\cfrac{5}{6+\cfrac{7}{8}}}}$
7. $\sqrt{\sqrt[3]{2+\sqrt[4]{3}}}$
8. $\frac{\sqrt[4]{6}}{(5-\sqrt{8})^3}$
9. $\sqrt{3+\sqrt{3+\sqrt{3+\sqrt{3}}}}$
10. $\sqrt{12^2+5^2}$
11. $\frac{a+\sin^3 b^2}{\cos 25+\operatorname{ctg} 60}$
12. $\cfrac{(x^8+8^x)^{\frac{1}{8}}(a^2-(b+\sqrt[3]{\operatorname{ctg} x})}{3-\cfrac{4}{\sqrt{x}-\sin^2 a^3}}$

13. $\dfrac{xyz - 3{,}3\left|x - \sqrt[4]{y}\right|}{10^7 + \ln\sin^2 x}$

14. $\dfrac{(b + \cos^2 y^4)(ab + \mathrm{tg}(x + \sqrt[3]{y^2})}{\cos 11^0 + \left|\mathrm{ctg}(y)\right|}$

15. $\dfrac{\sqrt{\dfrac{xb}{a}} + \cos^2(x+b)^3}{\dfrac{x^2(x+1)}{b} - \sin^2(x+a)}$

16. $\dfrac{\sin^3(x^2+a)^2 - \sqrt[3]{\dfrac{x}{b}}}{\dfrac{x^2}{a} + \cos^3(x+b)}$

17. $\dfrac{m\,\mathrm{tg}(t) + \left|\sqrt[3]{c\sin(t)}\right|}{m\cos(bt\sin^2 t) + \mathrm{ctg}\,z}$

18. $b\,\mathrm{tg}^2 x - \dfrac{a}{\sin^2\dfrac{x}{a}} + \dfrac{a\mathrm{e}^{-\sqrt{a}}}{\cos(\dfrac{bx}{a})}$

Корни квадратного уравнения вычисляются по формуле:

1. $x_{1,2} = \dfrac{b \pm \sqrt{b^2 - 4ac}}{2a}$

2. $\displaystyle\int_0^{\infty}\sum_{n=4}^{8}\frac{n^2 - \sqrt[3]{n-2}}{n^4}$

5.1.15. Оформление текста графическими объектами

Для придания тексту особой выразительности его можно украсить графическими объектами. Для этого имеется несколько возможностей.

- Если вы — будущий Репин, Ван Гог или Леонардо да Винчи, то можете воспользоваться средствами графических редакторов типа Photoshop, Corel Draw или, на худой конец, просто Paint, изобразить нужную вам картинку и разместить ее в тексте.

- Если вы ощущаете себя "всего-навсего" Казимиром Малевичем и способны лишь на "Черный квадрат", то можете воспользоваться панелью инструментов **Рисование**, встроенной в Word.
- Если никаких художественных способностей нет, но есть технические возможности типа сканера или цифровой камеры, то вы получаете с их помощью готовое изображение и размещаете его в вашем документе.
- Если и технические средства отсутствуют, то вам поможет только встроенная в Word библиотека готовых картинок.

Прежде чем воспользоваться всеми этими возможностями, надо напомнить об особенностях работы с графическими объектами в текстовых документах.

После размещения объекта в тексте с ним можно выполнять следующие основные действия.

- Объект можно *выделить*. Для этого достаточно щелкнуть на нем левой кнопкой мыши в тот момент, когда указатель мыши имеет вид . Изображение выделенного объекта приведено на рис. 5.34.

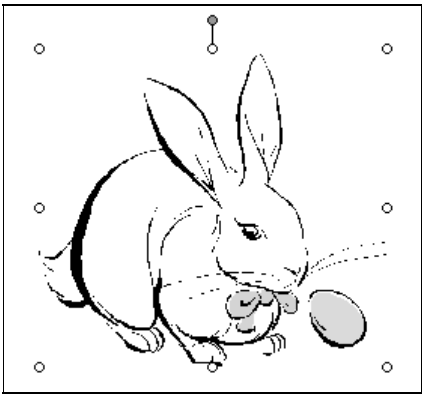
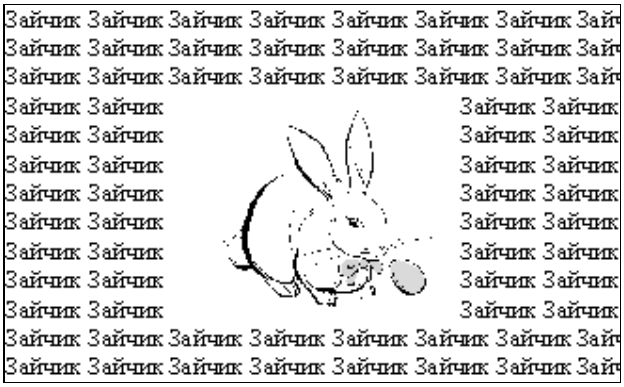


Рис. 5.34. Выделенный графический объект

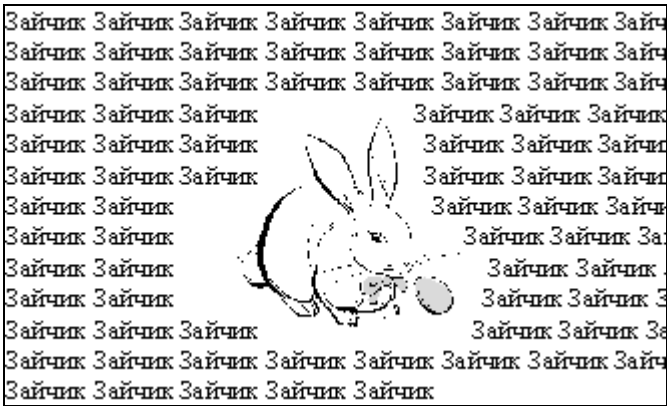
- Можно *изменить размеры* объекта. Для этого на белые кружочки (или квадратики, в зависимости от версии Word), имеющиеся вокруг объекта, необходимо навести указатель мыши, чтобы он принял вид черной двунаправленной стрелочки, и при нажатой левой кнопке мыши уменьшить либо увеличить размеры объекта. Причем, если потянуть за угловые маркеры, объект изменяется пропорционально, а за средние — растягивается (сжимается) по горизонтали или вертикали.
- Выделенный объект можно *перемещать* по документу. Для этого надо навести на выделенный объект указатель мыши до принятия им вида

и при нажатой левой кнопке мыши перетащить объект в нужное место. Если при этом удерживать еще и клавишу <Ctrl>, то объект скопируется.

- Объект можно разместить в тексте по-разному с точки зрения его *обтекания текстом*. Делается это при выделенном объекте и выборе команды меню **Формат | Рисунок | Положение | Обтекание | Дополнительно**. Можно выбрать различные варианты, например обтекание, показанное на рис. 5.35, называется **Вокруг рамки**.


Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зай
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зай
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зай
Зайчик Зайчик Зайчик Зайчик
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик

А вот это — **Сквозное** (рис. 5.37).


Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайч
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайч
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайч
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчи
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчи
Зайчик Зайчик Зайчик Зайчик Зайчи
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчи
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик
Зайчик Зайчик Зайчик Зайчик Зайчик З
Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик Зайчик

МАШИННЫЕ «ШЕДЕВРЫ»

Машины не перестают пробовать свои творческие» силы во многих жанрах. Электронные стихи и рассказы не новость для искушенного читателя.
Кибернетические машины написали несколько сценариев пьес. С изрядным успехом уже исполнялись машинные сюиты, гимны и песенки, вроде довольно популярной в некоторых странах «Красотки с кнопочным управлением».
Сумела машина и графически, в «художественных образах», воплотить смысл математических абстракций.
Итак, перед вами выступают «электронные авторы».
Сначала машины-прозаики.
Автор Мук — электронный мозг Манчестерского университета.

Рис. 5.38. Машинные шедевры

Любовное письмо

Мое маленькое сокровище! Моя вразумительная привязанность чудесно привлекает твой ласковый восторг. Ты мое любящее обожание, мое распирающее грудь обожание. Мое братское чувство с затаенным дыханием ожидает твоего дорогого нетерпения. Обожание моей любви нежно хранит твой алчный пыл.
Твой тоскующий Мук.

Рис. 5.39. Любовное письмо

Машинная проза представляет собой, конечно, образец самой модернистской литературы. Для сравнения приведу отрывок из сборника «Дневник мертвеца» французского литератора Марселя Беалю.

Залы готового платья

Я не опускаюсь, я скорее падаю, камнем падаю в воду. На поверхности образуется круг, в центре которого — я. Потом какой-то яростный катаклизм перемещает меня, я оказываюсь на

Рис. 5.40. Дневник мертвеца

Практикум 26. Изготовление визитки с внедрением графических объектов и использованием панели *Рисование*

В *Практикуме 17* мы уже делали визитку с использованием таблицы. Теперь же мы хотим сделать визитку более эффектной, украсив ее графическим объектом (это может быть как готовая картинка, так и ваша отсканирован-

ная фотография, разработанный вами логотип или эмблема проводимого мероприятия, если вы делаете badge — нагрудную визитку). Технология может быть примерно такой.

1. Настроить параметры страницы (командой **Файл | Параметры страницы**). Поля — все по 1,5 см. Ориентация — книжная.
2. Вставить подходящий рисунок (командой **Вставка | Рисунок | Картинки**). Командой **Формат | Рисунок** (или командой контекстного меню **Формат рисунка**) установить размеры 4×3,5 см.
3. Ниже рисунка вставить (командой **Вставка | Надпись**) и растянуть надпись 9×5 см (стандартные размеры визитки).
4. Переместить рисунок внутрь рамки визитки в левый верхний угол. Командой панели **Рисование | Действия | Порядок** поместить рисунок на передний план.
5. Инструментом **Надпись** растянуть внутри визитки справа от картинки рамку размером 5×3,5 см. Занести туда данные — фирма, фамилия, имя, отчество, должность. Разместить по центру, применить шрифтовое оформление. Убрать заливку и цвет линий рамки (инструментами панели **Рисование**).
6. Инструментом **Надпись** растянуть внутри визитки внизу рамку размером 9×1,5 см.
7. Занести туда адрес и телефон. Используя шрифт Wingdings, вставить перед адресом символ конверта (<Shift>+<8>) и перед телефоном символ телефона (<Shift>+<9>). Задать шрифтовое оформление.
8. Используя инструмент **Выбор объектов** панели **Рисование**, выделить всю визитку и из раскрывающегося списка **Действия** выбрать команду **Группировать**.

Рис. 5.41. Визитка-2

9. Передвинуть ставшую единым объектом визитку в левый верхний угол рабочего пространства страницы.
10. Удерживая <Ctrl>, скопировать визитку справа от созданной, а затем вниз. На странице должно получиться 10 визиток.
11. Сохранить работу под именем Vizitka2.doc.

Визитка должна иметь приблизительно такой вид, как на рис. 5.41.

Практикум 27. Зачетное задание на работу с инструментами панели *Рисование*, графическими объектами, шрифтовым оформлением, спецсимволами, группированием графических объектов и их копированием

1. Выполните оформление следующего документа (рис. 5.42).

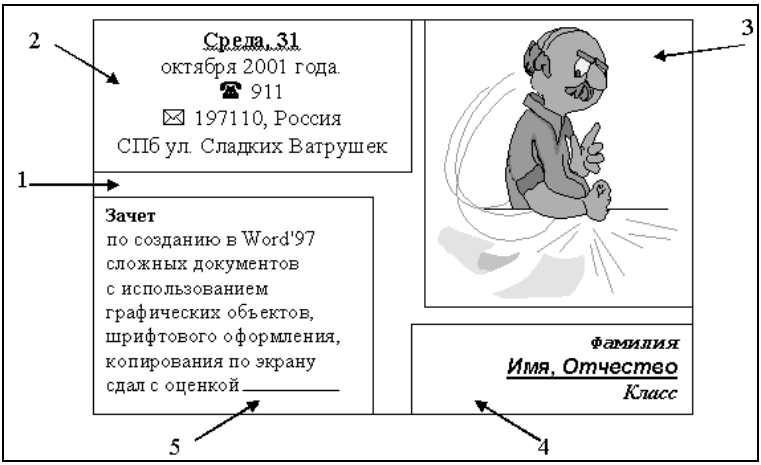


Рис. 5.42. Визитка-3

2. Скопируйте его на альбомную страницу формата А4 с полями: верхнее и нижнее — по 1 см, левое — 2,5 см, правое — 1,5 см.
3. Соблюдайте размеры и оформление всех рамок.
   - Общая рамка надписи 12×8 см.
   - Рамка 6,5×3 (текст тот же самый, шрифт Times New Roman, размер 13. Первая строка выделена полужирным, подчеркнута волнистой линией. Выравнивание по центру).
   - Рамка 5×5,5 см. Рисунок после вставки уменьшен до нужных размеров.
   - Рамка 5,5×2 см. Напишите *свои* фамилию, имя, отчество. Фамилия — шрифт Courier New, размер 12, полужирный курсив. Имя, отчество —

шрифт Arial, размер 12, полужирный курсив с подчеркиванием. Класс — шрифт Times New Roman, размер 12, полужирный курсив. Выравнивание по правому краю.

- Рамка 6×4,5 см. Текст тот же. Шрифт Times New Roman, первая строка — размер 11, полужирный. Остальной текст — размер 10, обычный. Выравнивание по левому краю.

4. Сгруппируйте все объекты, входящие в визитку, скопируйте визитки по странице встык, как указано на рис. 5.43 (их должно быть 4).

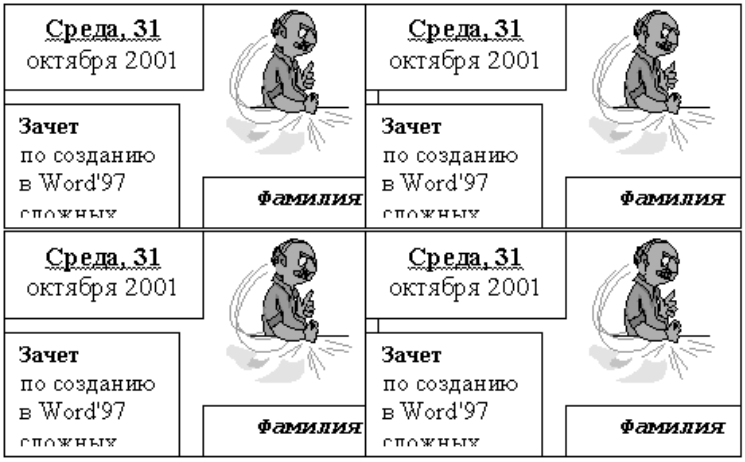


Рис. 5.43. Четыре визитки

5. Сохраните работу под именем Vizitka3.doc.

Практикум 28. Редактирование готового графического объекта в графическом редакторе Paint

Если вы вставили готовый графический объект, например, из библиотеки Word, то вы можете не только изменять его размеры и перемещать по документу, но и производить с ним более решительные действия. Давайте выполним следующее. В готовых картинках есть человечек с японским флагом, но нет с российским — несправедливо как-то. Попробуем восстановить справедливость.

1. Создадим новый документ. Напишем заголовок "Вперед, Россия!". Вставим человечка (рис. 5.44).
2. Выделим его. Вырежем в буфер обмена и, открыв графический редактор Paint, вставим в его рабочее окно. Выделим и удалим японский красный

круг. Проведем внутри флага две полоски. Закрасим среднюю полосу синим, а нижнюю — красным. Выделим то, что получилось, и командой меню **Рисунок | Отразить/повернуть** развернем нашего человечка слева направо.

3. А затем вырежем в буфер обмена и вставим в документ Word, где и разместим под надписью.

Должно получиться как на рис. 5.45.

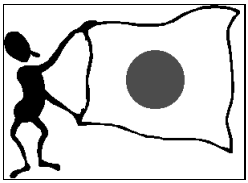


Рис. 5.44. Человечек с японским флагом

Рис. 5.45. Вперед, Россия!

Практикум 29. Редактирование готового графического объекта средствами панели *Рисование* Word

1. Вставьте из библиотеки, например, такой рисунок (рис. 5.46).
2. Выделите его, затем нажатием кнопки активизируйте панель **Рисование**. Из раскрывающегося списка **Действия** выберите команду **Разгруппировать**, тем самым разделив объект на его элементарные составляющие. Теперь, выделяя их по отдельности, мы можем производить над ними действия, которые нам предоставляют инструменты панели **Рисование**: **Повернуть/отразить**, **Надпись**, **Цвет заливки**, **Цвет линий** и т. д. Когда все необходимые действия будут выполнены, инструментом **Выбор объектов** заключите рисунок в прямоугольную область и выполните команду **Действия | Группировать**.

Должно получиться как на рис. 5.47.

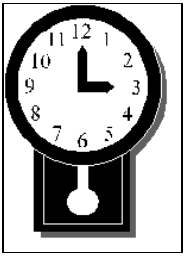


Рис. 5.46. Часы-1

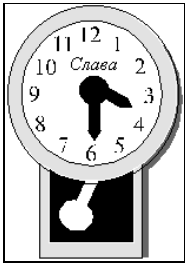


Рис. 5.47. Часы-2

5.1.16. Создание фигурного текста

Работая со шрифтовым оформлением, мы могли применять к введенному тексту выбор его размера, гарнитуры и начертания, но не могли, скажем, придать ему объем или тень, изменить угол поворота или пустить волной.

Все это возможно при пользовании инструментом **Добавить объект WordArt** панели **Рисование**. Порядок действий очень прост — щелкаете по этому инструменту, выбираете **Стиль надписи**, потом вводите текст, задаете ему размер, гарнитуру, начертание и размещаете в документе.

В документе работа с фигурным текстом ведется как с графическим объектом, кроме того, дополнительные возможности вызываются в контекстном меню командой **Формат Объекта WordArt**, а также на появляющейся панели инструментов **WordArt**. Возможностей там очень много, но полезно разобраться с ними самостоятельно, выбирая по очереди различные инструменты.

Чаще фигурный текст применяется при оформлении неофициальных документов или заголовков.

Практикум 30. Фигурный текст

Попробуйте представить, что вам нужно очень коротко заявить себя на межгалактические соревнования по игре в подкидного дурака. Разрешается представить свой ник (от англ. *Nickname* — прозвище), девиз и эмблему на формате А5 (148×210 мм).

Ник и девиз исполняются фигурным текстом. В качестве эмблемы допустимо использование готовой картинки, но с внесением заметных невооруженным глазом изменений. Получится у каждого свое, неповторимое. У одного товарища получилось так (рис. 5.48).

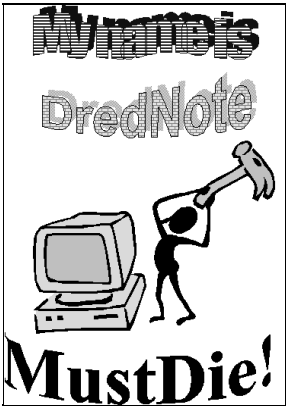


Рис. 5.48. Это — я!

5.1.17. Связывание документов гиперссылками

Наверняка многие из вас были в Интернете и видели, как работают там *гиперссылки*, когда по такому указателю мыши мы легко можем переходить от одной страницы к другой. Оказывается это легко организовать и в документах Microsoft Office. Сначала все связываемые между собой документы должны быть созданы и обязательно сохранены на диске. Затем в тексте документа выделяется ключевое слово или объект (в том числе и графический!), выбирается команда меню **Вставка | Гиперссылка** или инструмент **Добавить гиперссылку** на стандартной панели, выбирается файл, с которым осуществляется связь, и **OK**! Ключевое слово на глазах изменяет свой цвет, а указатель мыши при наведении на него принимает вид . При щелчке в этот момент левой кнопкой мыши происходит быстрый переход к нужному файлу.

Практикум 31. Гиперссылки

Все вы знаете телевизионную игру "О, счастливчик!" Многие, наверное, видели разнообразные ее компьютерные версии. Попробуем при помощи гиперссылок создать сложный документ, реализующий хотя бы один вопрос из этой игры (рис. 5.49).

Рис. 5.49. "О, счастливчик!"

1. Командой **Сервис | Параметры | Правка** уберите флажок **Ctrl+щелчок для выбора гиперссылки** (только для версии MSOffice XP).
2. Создайте в своей папке папку Миллионер.
3. Создайте в Word 10 фрагментов и сохраните их на диске (имена файлов указаны под фрагментами).

- Фрагмент 1.

  Игра "О, счастливчик!" + картинка-эмблема (рис. 5.49).

  Файл Igra.doc.

- Фрагмент 2.

  Хотите сыграть?

Да Нет

  Файл Yes_No.doc

- Фрагмент 3.

  Вопрос 1. На 100 рублей.

  Какую монетку достаточно было бросить для разговора в ленинградском телефоне-автомате в 1981 году?

  А) 1 копейка.

  Б) 2 копейки.

  В) 3 копейки.

  Г) 5 копеек.

  Подсказки:

  Мнение зала

  50:50

  Звонок другу

  Файл Quest100.doc

- Фрагмент 4.

  А жаль!

  Ну, хоть познакомьтесь с автором.

  Файл Pity.doc

- Фрагмент 5.

  Автор программы Фамилия, Имя

  Файл Author.doc

- Фрагмент 6.

  Ответ неверный. Ваш выигрыш 0 рублей. До свидания!

  Познакомьтесь с автором!

  Файл Error.doc

- Фрагмент 7.

  Это правильный ответ. Ваш выигрыш 100 рублей.

  Поздравляем!

До свидания!

Познакомьтесь с автором!

Файл True.doc

- Фрагмент 8.

  Мнения в зале распределились следующим образом:

  А) — 23%

  Б) — 37%

  В) — 18%

  Г) — 22%

  Возврат к вопросу.

  Файл Zal.doc

- Фрагмент 9.

  Остались два ответа:

  Б) 2 копейки.

  В) 3 копейки.

  Возврат к вопросу.

  Файл Fifty.doc

- Фрагмент 10.

  Ваш друг думает, что правильный ответ: Б) 2 копейки.

  Возврат к вопросу.

  Файл Drug.doc

4. Установите требуемые гиперссылки.
5. Проведите тестирование гипертекстового документа.
6. Сдайте работу преподавателю.

Замечание

В первом фрагменте установите гиперссылку на эмблему, связав ее с фрагментом 2.

5.1.18. Добавление закладки

Закладка — элемент документа, которому присвоено уникальное имя. Это имя можно использовать для последующих ссылок. Например, можно использовать закладку для определения текста, который необходимо прове-

рить позже. Чтобы в следующий раз не искать данный текст во всем документе, можно перейти к нему с помощью диалогового окна **Закладка**.

1. Выделите элемент, которому следует назначить закладку, или щелкните на том месте в документе, куда следует вставить закладку.
2. В меню **Вставка** выберите команду **Закладка**.
3. В поле **Имя закладки** введите или выберите нужное имя.

   Имя закладки должно начинаться с буквы, в нем могут использоваться цифры, но не должно быть пробелов. Если требуется разделить слова в имени закладки, пользуйтесь знаком подчеркивания, например `Первый_заголовок`.
4. Нажмите кнопку **Добавить**.

Переход к определенной закладке

1. В меню **Вставка** выберите команду **Закладка**.
2. В группе **Сортировать по** выберите необходимый способ сортировки имен закладок.
3. Если необходимо отображать скрытые закладки, такие как перекрестные ссылки, то установите флажок **Скрытые закладки**.
4. В поле **Имя закладки** выберите закладку, к которой необходимо перейти.
5. Нажмите кнопку **Перейти**.

Практикум 32. Использование закладок

1. Загрузите файл Parametr.doc. Установите первую закладку на слово "абдрапапупа". Назовите ее `Стих1`. Вторую закладку поставьте на слова "Злой музыкант" и назовите ее `Стих2`. Вернитесь в начало документа. Перейдите сначала к первой закладке, а потом ко второй.
2. Сохраните документ под тем же именем.

5.1.19. Создание простейших Web-страниц

В меню **Файл | Создать** имеется мастер Web-страниц, который и поможет вам создать простейшую страничку с использованием встроенных шаблонов. По сути, сайт в Интернете не что иное, как гипертекстовый документ.

Практикум 33. Создание Web-страницы с использованием встроенного мастера Web-страниц Word

Подготовьте следующие материалы для создания вашей Web-страницы.

- Свою краткую биографию: дата рождения, состав семьи, школа, яркие детские воспоминания, увлечения.
- Фотоальбом из заранее сканированных фотографий.

При помощи мастера Web-страниц организуйте свой гипертекстовый документ в папке Webpage. Там должно содержаться три связанных между собой файла: Main.htm (Главная страница), на которой расположены название сайта и ссылки на биографию и фотоальбом, файл Biograph.doc (биография) и Photo.doc (фотоальбом). С любой страницы мы должны иметь возможность перейти на любую другую.

5.1.20. Создание документов по шаблону

В Word имеются встроенные шаблоны для создания деловых документов — резюме, писем, факсов и т. д. Найти их можно в меню **Файл | Создать**. Там есть мастера по созданию этих документов, которые по шагам помогут вам в работе.

Практикум 34. Создание документа по шаблону

- Представьте себе, что вам надо устроиться на работу в фирму Microsoft. Создайте изысканное резюме.
- А теперь вам представилась возможность послать факс самому Биллу Гейтсу. Создайте факс по шаблону.

Практикум 35. Последнее задание по Word

Последнее задание состоит из двух частей — теоретической и практической. Выполнить надо обе.

Теоретическая часть

Настольная издательская система (НИС) — это совокупность аппаратных и программных средств для подготовки и создания образца печатной продукции, готового для тиражирования.

- Аппаратные средства. Перечислите:
  - устройства ввода информации в НИС;
  - устройства обработки и хранения информации в НИС;
  - устройства вывода информации в НИС.

- Программные средства. Перечислите наиболее известные:
  - программы обработки изображений для НИС;
  - программы разработки печатного издания для НИС.
- Редактирование текста.
  - какой символ будет удален при нажатии клавиши <Delete> в слове "Сфинкс", если курсор установлен между буквами "ф" и "и"?
  - какой символ будет удален при нажатии клавиши <Backspace> в слове "Сфинкс", если курсор установлен между буквами "н" и "к"?
  - что произойдет, если при курсоре, установленном в начало строки, нажать клавишу <Enter>?
  - что произойдет, если при курсоре, установленном в середину набранной строки, нажать клавишу <Enter>?
  - какую клавишу надо нажать, чтобы получать только заглавные буквы?
  - как правильно ставить пробелы около знаков препинания?
  - какая существует разница при вводе знаков "дефис" и "тире"?
  - какие правила существуют при заключении фрагментов текста в кавычки и скобки?
  - что является словом для программы Word?
  - что является абзацем для программы Word?
- Какие ориентации листа бумаги вы знаете? Как изменить ориентацию бумаги?
- Какие действия Microsoft Word позволяет производить над шрифтами?
- Как удалить фрагмент текста?
- Как добавить строку в таблицу?
- Как удалить строку из таблицы?
- Что такое гипертекст? Где он применяется?
- Как изменить размеры вставленного рисунка? Как разместить его в центре текста?
- Как компьютер проверяет орфографию?

Практическая часть

Подготовьте гипертекстовый документ Microsoft Word, содержащий три страницы, каждая из которых должна быть сохранена в папке Гипертекст, вложенной в вашу папку, под именами Главная, Биография и Фотоальбом соответственно.

- Первая страница. **Титульный лист**.
  - Параметры страницы. Поля: левое 2,5 см, правое 1,5 см, верхнее 2 см, нижнее 2 см, ориентация книжная.

- Вверху надпись: Министерство образования Российской Федерации (по центру, размер 14, полужирный, Arial Cyr).
- На уровне 4 см сверху (ориентироваться по вертикальной линейке) надпись: Средняя школа № (по центру, размер 13, полужирный, подчеркнутый двойной линией, Courier New).
- На уровне 8 см надпись: Зачетная работа по настольным издательским системам (фигурный текст в 2 строки, размер 24, стиль по выбору).
- На следующей строке гиперссылка: Биография (по правому краю, отступ справа на отметке 12 см).
- На следующей строке гиперссылка: Фотоальбом (по правому краю, отступ справа на отметке 12 см).
- На уровне 15 см надпись: Учащийся __ класса ФИО (по правому краю, размер 14, Times New Roman).
- Ниже надпись: Преподаватель ФИО (по правому краю, размер 14, Times New Roman).
- На уровне 23 см надпись: Санкт-Петербург (по центру, размер 14, полужирный, Arial Cyr).
- Ниже надпись: 2003 (по центру, размер 14, полужирный, Arial Cyr).
- Между надписями "Преподаватель" и "Санкт-Петербург" разместить рисунок (по выбору) размером 4×4 см, по центру.
- Задать титульному листу обрамление рисунками (**Формат | Границы и заливка | Страница | Рисунок** (выбор рисунка) и **Применить | Только к первой странице**).
- Сохраните под именем Главная.doc.

❒ Вторая страница. **Биография**.

- Изложите вымышленную биографию кандидата на пост президента РФ по следующим пунктам (оформить в виде таблицы с двумя столбцами: первый — название пункта, второй — содержание пункта):

  ФИО, дата рождения, место рождения, образование, партийная принадлежность, места работы (не менее трех), семейное положение, наличие детей, основные черты характера, хобби (выравнивание по ширине, автоматическая расстановка переносов, размер 14, ФИО выделить полужирным с подчеркиванием).
- Внизу листа разместите фигурным текстом рекламный слоган для данного кандидата, кратко выражающий его политическую платформу и программу.
- В самом низу разместите гиперссылки "Главная" и "Фотоальбом" (по правому краю).
- Сохраните под именем Биография.doc.

- Третья страница. **Фотоальбом**.
  - Разместите в качестве фотографий кандидата четыре готовых из библиотеки. Снабдите их подписями (с помощью инструмента **Надпись** панели рисования).
  - Внизу разместите гиперссылки "Главная" и "Биография" (по правому краю).
  - Сохраните под именем Фотоальбом.doc.
  - Свяжите документы между собой гиперссылками. Отладить их работу.

5.2. Обработка числовой информации MS Excel

Если Word каждому культурному человеку знать просто необходимо, то с *электронными таблицами* Excel мы тоже очень рекомендуем познакомиться, а уже потом решить для себя, будут ли у вас задачи, которые удобно решать в этой среде.

После изучения электронных таблиц Excel учащийся средней школы должен уметь:

- осуществлять ввод данных;
- редактировать данные;
- наносить шрифтовое и цветовое оформление;
- применять к таблице обрамление;
- выделять фрагменты таблицы;
- копировать и перемещать данные, пользоваться автозаполнением;
- использовать в работе арифметические функции, функции автосуммирования, нахождения среднего значения, максимума и минимума, логическую функцию **Если**;
- изменять ширину столбцов и высоту строки, удалять и вставлять столбцы и строки;
- применять выделение и размещение по центру;
- вставлять рисованные объекты;
- форматировать ячейки;
- пользоваться различными форматами данных;
- применять Мастера функций и Мастера диаграмм;
- писать простейшие макросы;
- вставлять документы электронных таблиц в документы других приложений Microsoft Office.

Проведем ряд практикумов для освоения всего вышеперечисленного.

5.2.1. Ввод и редактирование данных. Использование простейших формул. Относительная и абсолютная адресация

Практикум 36. Прайс-лист

1. Создайте таблицу следующего вида (рис. 5.50).

	A	B	C	D	E	F	G
1	ООО "Аленький цветочек"						
2	Прайс-лист на		13.10.03				
3	Курс $=	31.7					
4	№ поз.	Наим.товара	Цена($)	Цена(руб)	Ед.изм.	Кол-во	Стоимость(руб)
5	1	Палочка волш.	300		шт.	3	
6	2	Зелье-приворотное	100		литр	10	
7	3	Вода живая	50		литр	0.5	
8	4	Вода мертвая	400		литр	0.7	
9	5	Ковер-самолет	1200		шт.	1	
10	6	Скатерть-самобр.	700		шт.	8	
11	7	Сапоги-скороходы	200		пар	12	
12	8	Лягушка-царевна	5000		шт.	6	
						Итого:	

Рис. 5.50. Прайс-лист

2. Номера позиций введите, используя автозаполнение.
3. Напишите, используя абсолютную адресацию, в ячейке **D5** формулу для вычисления цены товара в рублях, затем за маркер заполнения скопируйте ее до **D12**.
4. Напишите, используя относительную адресацию, в ячейке **G5** формулу для стоимости, затем за маркер заполнения скопируйте ее до **G12**.
5. Используя автосуммирование, вычислите "Итого" в ячейке **G13**.
6. Нанесите сетку таблицы там, где это необходимо.
7. Сохраните документ под именем Price.xls.
8. Измените курс доллара на 2,5. Посмотрите, что изменилось.
9. Поменяйте произвольно количество товара. Посмотрите, что изменилось.

Практикум 37. Прибыль ООО "Аленький цветочек" за 2003 год

1. Создайте следующую таблицу (рис. 5.51).
2. Вместо ФИО введите свои фамилию, имя и отчество.

	A	B	C	D
1	ООО "Аленький цветочек"			
2	Отчет по прибыли за 2003 год			
3	Главный бухгалтер ФИО			
4	Месяц	Доход	Расход	Прибыль
5	Январь	13 000р.	7 800р.	
6	Февраль	25 000р.	13 500р.	
7	Март	51 000р.	37 000р.	
8	Апрель	17 000р.	7 000р.	
9	Май	23 000р.	25 000р.	
10	Июнь	42 000р.	40 000р.	
11	Июль	20 000р.	18 000р.	
12	Август	48 000р.	20 000р.	
13	Сентябрь	35 000р.	30 000р.	
14	Октябрь	98 000р.	8 000р.	
15	Ноябрь	37 000р.	7 500р.	
16	Декабрь	21 000р.	12 200р.	
17	***Итого***			
18	***Среднее***			
19	***Макс.***			
20	***Мин.***			

Рис. 5.51. Прибыль ООО "Аленький цветочек"

Совет

Для введения названий месяцев используйте функцию автозаполнения.

Совет

При заполнении столбцов прихода и расхода настройте предварительно денежный формат ячейки.

3. В ячейку **D5** введите формулу расчета прибыли — `Приход-Расход`. Настройте ячейки так, чтобы в случае отрицательного баланса цифры выделялись красным цветом.
4. При помощи маркера заполнения скопируйте формулу до **D16**.
5. "Итого" подсчитайте при помощи автосуммирования.
6. Среднее, минимальное и максимальное значение — с использованием функций **СРЗНАЧ**, **МАКС** и **МИН**.
7. Нанесите сетку.
8. Примените шрифтовое оформление и заливку шапки таблицы.
9. Сохраните документ под именем Plus.xls.

Практикум 38. Сальдо фирмы "Аленький цветочек"

1. Откройте таблицу Plus.xls.

Совет

Для того чтобы в одной ячейке осуществлять ввод в две или более строки, необходимо при вводе в месте переноса нажимать комбинацию клавиш: левый <Alt>+<Enter>.

2. Введите формулы расчета общей прибыли в ячейки **B6:D6**.
3. В ячейке **E4** задайте формулу для суммирования данных в ячейках **B4:D4** и скопируйте ее в ячейки **E5:E6**.
4. Введите формулу суммирования платежей в ячейку **B17** и скопируйте ее в ячейки **C17**, **D17**.
5. В ячейке **E9** задайте формулу для суммирования данных в ячейках **B9:D9** и скопируйте ее в ячейки **E10:E16**.
6. Введите в ячейку **E17** формулу суммы всех платежей.
7. Задайте в ячейке **B19** формулу расчета "Сальдо" (разность между общей прибылью и всеми платежами) и скопируйте ее в ячейки **C19:E19**.

	A	B	C	D	E
1	**Фирма "Аленький цветочек"**				
2		*Январь*	*Февраль*	*Март*	*Всего*
3	**Прибыль**				
4	Продажа	1000	2200	2800	
5	Банковский заём	500			
6	***Общая прибыль***				
7					
8	**Платежи**				
9	Материалы	700		1200	
10	Услуги	450			
11	Телефоны		1050		
12	Зарплата	1200		1200	
13	Письменные принадлежности	50		40	
14	Почтовый сбор	30	1200	33	
15	Реклама	700	40		
16	Выплата займа	136	25	136	
17	***Всего платежей***				
18					
19	***Сальдо***				

Рис. 5.52. Сальдо фирмы "Аленький цветочек"

8. Примените шрифтовое оформление.
9. Нанесите сетку и примените цветовое оформление (рис. 5.52).
10. Сохраните документ по именем Saldo.xls.

Практикум 39. Кроссворд

Имеется следующий набор компьютерных терминов (мы, кстати, по ним иногда проводим и терминологический диктант. К сожалению, с русским языком у учащихся не все бывает в порядке. А очень хочется...).

Информатика, Информация, Компьютер, Алгоритм, Программа, Процессор, Монитор, Принтер, Сканер, Клавиатура, Дискета, Винчестер, Дисковод, Мультимедиа, Модем, Интернет, Курсор, Корзина, Картридж, Ярлык, Значок, Копирование, Виртуальность, Палитра, Транслятор, Интерпретатор, Обеспечение, Компилятор, Архиватор, Сервер, Хакер, Утилита, Драйвер, Микросхема, Табуляция, Телеконференция, Контроллер, Анимация, Перфокарта, Провайдер, Протокол, Дефрагментация, Ноутбук, Терабайт, Гейтс, Паскаль, Перфокарта, Бод.

Понятно, что учащиеся могут блеснуть своими познаниями и добавить своих терминов, так или иначе связанных с компьютером и информатикой.

А для желающих есть и англоязычные термины.

Abort, Acceleration, Accept, Accessories, Action, Adapter, Add, Application, Attachment, Background, Beta, Bit, Boot, Break, Browser, Bug, Button, Byte, Cable, Cache, Cancel, Cartrige, Chat, Chip, Clear, Click, Clipboard, Cluster, Configuration, Connection, Copy, crack, Create, Custom, Desktop, Destination, Device, Directory, Disk, Domain, Drive, Driver, Enter, Error, Explorer, Fail, File, Format, Folder, Font, Forward, Freeware, Game, Gamer, Hack, Hacker, Hardware, Header, Hyperlink, Hypertext, Icon, Ignore, Image, Input, Install, Internet, Join, Joystick, Jumper, Keyboard, Lamer, Link, Level, Load, Login, Macros, Mail, Manual, Memory, Menu, Message, Motherboard, Mouse, Multimedia, Network, Notepad, Offline, Online, Open, Options, Page, Password, Paste, Patch, Port, Power, Print, Printer, Properties, Protocol, Provider, Readme, Refresh, Resolution, Reset, Rules, Rulez, Run, Sample, Save, Scan, Scanner, Screen, Search, Security, Send, Server, Settings, Shareware, Shortcut, Size, Slot, Software, Suxx, Taskbar, Temporary, Unable, Update, Upgrade, User, Utilities, Version, Window, Windows, Zoom.

Надо выполнить следующее.

1. Построить кроссворд на бумаге (это можно использовать как домашнее задание).
2. Загрузить электронные таблицы.
3. Подсчитать в кроссворде количество столбцов и строк.

4. Выделить необходимое количество столбцов и командой меню **Формат | Столбец | Ширина столбца** задать ширину столбца равной 3.
5. Выделить необходимое количество строк и командой меню **Формат | Строка** настроить высоту строки равной 18.
6. Нанести границы клеток. Для этого — выделить необходимые клетки и на панели инструментов нажать кнопку **Границы**.
7. Нанести зачернение клеток. Для этого — выделить необходимые клетки и нажать на панели инструментов кнопку **Цвет заливки**.
8. Расставить цифры в клетках. Задать их выравнивание по левому краю по горизонтали и по верхнему краю по вертикали. Подобрать такой размер, чтобы оставалось место для записи букв.
9. Сохранить работу под именем Cross.xls.
10. Свернуть электронные таблицы.
11. Открыть Word.
12. Набрать определения к словам (хотелось бы, чтобы они были оригинальными или хотя бы смешными, а не списанными просто из учебника или словаря).
13. Развернуть электронные таблицы.
14. Выделить кроссворд.
15. Скопировать его в буфер обмена.
16. Перейти в Word.
17. Установить курсор в место вставки кроссворда (перед словом `По вертикали`).
18. Вставить из буфера.
19. Подобрать шрифтовое оформление таким образом, чтобы кроссворд помещался на одну страницу.
20. Подписать внизу свою фамилию и сохранить работу под именем My_cross.doc.

5.2.2. Использование функции *ЕСЛИ*

Практикум 40. Проходной балл

1. Создайте таблицу как на рис. 5.53 без заполнения числовыми данными.
2. Оформите название таблицы командой **Объединить и поместить в центре**.
3. Заполните данные по оценкам с использованием функции получения случайных чисел в пределах от 5 по 10 включительно (**СЛЧИС()** и **ЦЕЛОЕ**). Применив спецвставку, оставьте в этих ячейках только их значения.

4. Выполните подсчет общего балла командой **Автосуммирование**.
5. Выполните подсчет средних значений по каждому абитуриенту и экзамену с помощью функции **СРЗНАЧ**.
6. В столбце **Сообщение о зачислении** используйте функцию **ЕСЛИ** для сообщения "Зачислен", если сумма баллов больше проходного, и сообщения "Отказать" в обратном случае.
7. Через условное форматирование настройте ячейки столбца **Сообщение о зачислении** так, чтобы при появлении в них слова "Зачислен" она заливалась желтым цветом, а само сообщение было бы выделено полужирным синим.
8. Примените шрифтовое оформление, заливку.
9. Поменяв оценки у какого-нибудь абитуриента, убедитесь в работоспособности всех ваших формул.
10. Сдайте работу преподавателю, сохранив ее под именем Ball.xls.

	A	B	C	D	E	F	G		
1					**Ведомость**				
2					приема на обучение				
3					в Санкт-Петербургский вольный университет хакерства				
4									
5			***Проходной балл***		36.6				
6									
7	**№ абит.**	**Фамилия**	**Взлом паролей**	**Атака на сайт**	**Бесплатный выход в Интернет**	**Написание вирусов**	**Общий балл**	**Средний балл**	**Сообщение о зачислении**
8	1	Паскаль	5	5	9	10	*29*	*7.25*	отказать
9	2	Вирт	7	10	10	10	*37*	*9.25*	**зачислен**
10	3	Гейтс	10	5	9	6	*30*	*7.50*	отказать
11	4	Бэббидж	5	6	8	8	*27*	*6.75*	отказать
12	5	Буль	9	9	8	6	*32*	*8.00*	отказать
13	6	Нортон	6	8	8	6	*28*	*7.00*	отказать
14	7	Возняк	7	5	5	9	*26*	*6.50*	отказать
15	8	Нейман	7	5	10	9	*31*	*7.75*	отказать
16	9	Винер	8	8	8	10	*34*	*8.50*	отказать
17	10	Ваша фамилия	10	10	10	10	*40*	*10.00*	**зачислен**
18	**Средний балл за экзамен**		*7.40*	*7.10*	*8.50*	*8.40*			

Рис. 5.53. Проходной балл

Практикум 41. Начисление зарплаты

1. Подготовьте таблицу как на рис. 5.54.
2. Напишите соответствующие формулы в столбцах: **Оклад (руб)**, **% премии**, **Премия (руб.)**, **Итого (руб.)**, **Налог (руб.)** и **На руки (руб.)**.
   - *Оклад в рублях* рассчитывается исходя из оклада в долларах, курса доллара плюс 10% к получившейся сумме от доброго хозяина фирмы.

- *Процент премии* рассчитывается так: если работник отработал все рабочие дни, то получает 100% премии, если больше 14 дней, но не 22, то — 50%, во всех остальных случаях 25%.
- *Премия в рублях* рассчитывается как оклад в рублях, умноженный на % премии.
- *Итого в рублях* получается по формуле:

  (отработка × оклад / количество рабочих дней) + сумма премии.
- *Налог*: если Итого получается меньше 10 000 рублей, то 13%, иначе 25%.
- *На руки*: Итого − Налог.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Таблица расчета зарплаты за				октябрь	2003										
2																
3	*Рабочих дней*		22													
4	*Курс доллара*		31													
5																
6													Премия	Итого	Налог	На руки
7	№№	Фамилия	Имя	Отчество	Таб.номер	Отдел	Должность	Дата приема	Отработка	Оклад ($)	Оклад(руб.)	Премия %	руб.	руб.	руб.	руб.
8	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
9	1	Иванов	Иван	Иванович	001	1	инженер	01.01.92	15	300						
10	2	Петров	Николай	Борисович	002	1	инженер	03.01.89	18	300						
11	3	Смирнов	Евгений	Петрович	004	2	ст. инженер	12.09.96	22	500						
12	4	Жуков	Петр	Сергеевич	120	1	техник	23.07.96	10	250						
13	5	Степанов	Борис	Николаевич	451	3	инженер	05.08.94	1	400	1	2	3	4	5	6
14	6	Кузьмин	Владимир	Иванович	340	3	инженер	21.10.92	22	320						
15	7	Иванов	Петр	Петрович	005	2	ст.инженер	04.05.93	21	350						
16	8	Сурнова	Ольга	Николаевна	006	2	ст.инженер	11.11.91	18	550						
17	9	Зорина	Елена	Петровна	180	2	нач.лаб.	31.10.94	22	800						
18	10	Тюрина	Надежда	Ивановна	450	1	инженер	12.04.93	13	400						
19	11	Моисеева	Елена	Сергеевна	321	2	инженер	01.04.92	10	300						
20	12	Федоров	Степан	Борисович	900	2	инженер	24.03.93	6	300						

Рис. 5.54. Начисление зарплаты

3. Добейтесь, чтобы во всех рассчитываемых ячейках был настроен денежный формат с двумя десятичными знаками.
4. Примените шрифтовое оформление и заливку.
5. Сохраните под именем Zarplata.xls.
6. "Обрушив" курс доллара до 10 рублей за 1 доллар, убедитесь, что все ваши формулы работают нормально.
7. Сдайте работу преподавателю.

5.2.3. Функция *ЕСЛИ*. Построение графиков и диаграмм

Практикум 42. Рыбку жалко...

Вариант 1

Запас рыбы в пруду оценен в 1200 тонн. Ежегодный прирост рыбы составляет 15%. Ежегодный план отлова — 300 тонн. Наименьший запас рыбы, ниже которого запас уже не восстанавливается, — 400 тонн. Постройте таблицу, рассчитывающую количество рыбы в пруду на 15 лет. Пометьте, начиная с какого момента невозможно выполнить заданный план отлова. Постройте график изменения количества рыбы в пруду.

Вариант 2

Запас рыбы в пруду оценен в 1000 тонн. Ежегодный прирост рыбы составляет 13%. Ежегодный план отлова — 180 тонн. Наименьший запас рыбы, ниже которого запас уже не восстанавливается, — 250 тонн. Постройте таблицу, рассчитывающую количество рыбы в пруду на 20 лет. Пометьте, начиная с какого момента невозможно выполнить заданный план отлова. Постройте график изменения количества рыбы в пруду.

Вариант 3

Запас рыбы в пруду оценен в 1800 тонн. Ежегодный прирост рыбы составляет 17%. Ежегодный план отлова — 400 тонн. Наименьший запас рыбы, ниже которого запас уже не восстанавливается, — 500 тонн. Постройте таблицу, рассчитывающую количество рыбы в пруду на 16 лет. Пометьте, начиная с какого момента невозможно выполнить заданный план отлова. Постройте график изменения количества рыбы в пруду.

Практикум 43. Олимпиада по программированию

Вариант 1

Олимпиада по программированию оценивается по сумме очков, полученных за каждую из трех задач, плюс 0,1 от набранной суммы для учащихся классов младше 10-го. В олимпиаде принимало участие 12 человек: 4 из 8-го класса, 3 — из 9-го, 3 — из 10-го и 2 — из 11-го. Первое задание оценивалось максимум в 10 баллов. Второе — в 8, третье — в 12. Набравшие больше 27 баллов получают диплом 1-й степени, больше 25 — 2-й степени, больше 23 — третьей степени. Создайте таблицу участников и их результатов. Определите дипломы участников. Постройте диаграмму по сумме набранных очков для получивших диплом 1-й степени.

Вариант 2

Олимпиада по программированию оценивается по сумме очков, полученных за каждую из трех задач, плюс 0,1 от набранной суммы для учащихся классов младше 10-го. В олимпиаде принимало участие 14 человек: 3 из 8-го класса, 4 — из 9-го, 4 — из 10-го и 3 — из 11-го. Первое задание оценивалось максимум в 12 баллов. Второе — в 10, третье — в 12. Набравшие больше 30 баллов получают диплом 1-й степени, больше 27 — 2-й степени, больше 25 — третьей степени. Создайте таблицу участников и их результатов. Определите дипломы участников. Постройте диаграмму по сумме набранных очков для получивших диплом 2-й степени.

Вариант 3

Олимпиада по программированию оценивается по сумме очков, полученных за каждую из трех задач, плюс 0,1 от набранной суммы для учащихся классов младше 10-го. В олимпиаде принимало участие 10 человек: 2 из 8-го класса, 3 — из 9-го, 3 — из 10-го и 2 — из 11-го. Первое задание оценивалось максимум в 15 баллов. Второе — в 12, третье — в 10. Набравшие больше 34 баллов получают диплом 1-й степени, больше 30 — 2-й степени, больше 27 — третьей степени. Создайте таблицу участников и их результатов. Определите дипломы участников. Постройте диаграмму по сумме набранных очков для получивших диплом 3-й степени.

Практикум 44. Лесорубы

Для расстановки участников соревнований по местам используйте функцию **РАНГ**.

Вариант 1

Соревнование лесорубов состоит из 2-х этапов: 1. Рубка дерева, 2. Метание топора. Если участник рубил больше 10 мин., то он получает 0 очков, если меньше, то — (600$-t$ с) очков. Если участник метнул топор меньше, чем на 5 м, то он получает 0 очков, если больше, то за каждый 1 см свыше 5 м — 1 очко. В зачет идет сумма очков за 1 и 2 этапы. Расставьте участников по местам, набравшие одинаковое количество очков располагаются в алфавитном порядке. Постройте диаграмму по очкам участников, занявших 3 первых места. Всего участвовало 10 лесорубов.

Вариант 2

Соревнование лесорубов состоит из 2-х этапов: 1. Рубка дерева, 2. Метание топора. Если участник рубил больше 15 мин., то он получает 0 очков, если меньше, то — (900$-t$ с) очков. Если участник метнул топор меньше, чем на 6 м, то он получает 0 очков, если больше, то за каждый 1 см свыше 6 м — 2 очка. В зачет идет сумма очков за 1 и 2 этапы. Расставьте участников по местам, набравшие одинаковое количество очков располагаются в алфавит-

ном порядке. Постройте диаграмму по очкам участников, занявших 5 первых мест. Всего участвовало 12 лесорубов.

Вариант 3

Соревнование лесорубов состоит из 2-х этапов: 1. Рубка дерева, 2. Метание топора. Если участник рубил больше 12 мин., то он получает 0 очков, если меньше, то — (720−t с) очков. Если участник метнул топор меньше, чем на 7 м, то он получает 0 очков, если больше, то за каждый 1 см свыше 7 м — 3 очка. В зачет идет сумма очков за 1 и 2 этапы. Расставте участников по местам, набравшие одинаковое количество очков располагаются в алфавитном порядке. Постройте диаграмму по очкам участников, занявших 4 первые места. Всего участвовало 13 лесорубов.

Практикум 45. Деньги в банке

Допустим, вам досталось в наследство 100 000 рублей. Надо бы ими как-то распорядиться. Вы склоняетесь к двум вариантам: либо купить на все деньги доллары и спрятать их в банку, а по истечении года продать и нажить небольшой капиталец за счет роста курса, либо вложить в родной Сбербанк и получить проценты.

Необходимо спрогнозировать, как выгоднее распорядиться этими деньгами. Предположим, что банк дает 18% годовых для рублевых вкладов, а доллар растет равномерно в течение года на 20 копеек в месяц. После выполнения задания необходимо варьировать значения процентных ставок, чтобы выяснить, при каком их значении было бы выгодно хранить деньги в банке из-под кофе. Образец таблицы приведен на рис. 5.55.

Расчёт выгодности хранения денег в России.						
Нач.капитал=	100000	курс $=	31.00		Кол-во $:	3226
	В банке				В долларах	
Месяц	Рост руб.		Рост $		Кол-во $	Рубли на $
январь	101 500р.		31.20		$3 226	100645.16
февраль	103 023р.		31.40		$3 226	101290.32
март	104 568р.		31.60		$3 226	101935.48
апрель	106 136р.		31.80		$3 226	102580.65
май	107 728р.		32.00		$3 226	103225.81
июнь	109 344р.		32.20		$3 226	103870.97
июль	110 984р.		32.40		$3 226	104516.13
август	112 649р.		32.60		$3 226	105161.29
сентябрь	114 339р.		32.80		$3 226	105806.45
октябрь	116 054р.		33.00		$3 226	106451.61
ноябрь	117 795р.		33.20		$3 226	107096.77
декабрь	119 562р.		33.40		$3 226	107741.94

Рис. 5.55. Наследство

Постройте график прироста денежных средств по первому и второму вариантам (рис. 5.56).

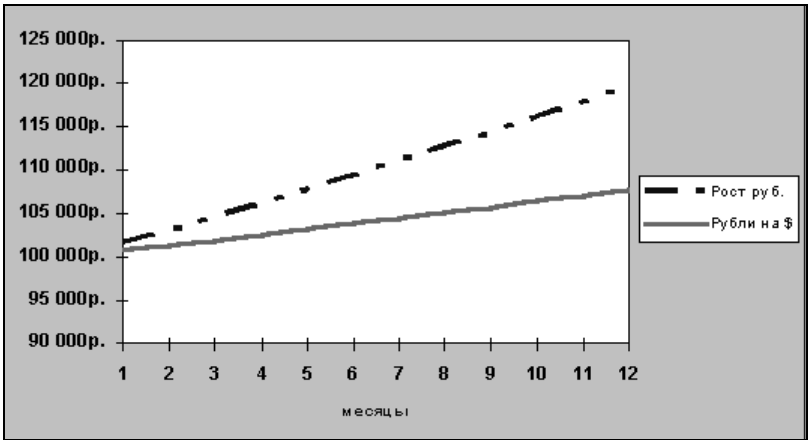


Рис. 5.56. График роста денежной массы

5.2.4. Комплексные задания

Практикум 46. Комплексное задание по электронным таблицам Excel. Книжные магазины

1. Загрузите Excel.
2. Добавьте к имеющимся трем еще один лист.
3. Переименуйте листы следующим образом:
   - Лист 1 — в `Выручка`;
   - Лист 2 — в `Себестоимость`;
   - Лист 3 — в `График`;
   - Лист 4 — в `Прибыль`.
4. На лист **Выручка** занесите и оформите первую таблицу (рис. 5.57).

   Произведите следующие расчеты:
   - в столбце **Всего за 6 лет, долл.** — суммирование по строке;
   - в столбце **Доля в общей выручке** — вычисление доли каждого магазина в общей выручке в %;
   - в столбце **Ранг** — расстановка магазинов по местам, с помощью функции **РАНГ**;

Выручка от реализации книжной продукции издательства ЭКОМ

№ п/п	Магазин	Годы						Всего за 6 лет, долл.	Доля в общей вы-	Ранг	Тенденция за последние 3 года
		1994, долл.	1995, долл.	1996, долл.	1997, долл.	1998, долл.	1999, долл.				
1	Дом книги	2 456.1	2 562.0	3 323.0	4 564.5	3 983.2	3 654.3	20 543.1	15.66%	4 место	Стабильное снижение
2	Книжным мир	3 520.0	3 645.2	4 553.0	3 656.2	4 564.3	4 564.4	24 503.1	18.68%	3 место	Стабильный рост
3	Знание	676.2	525.2	589.0	756.0	654.1	644.1	3 844.6	2.93%	7 место	Стабильное снижение
4	Наука	353.3	632.0	863.0	863.1	1 023.2	752.1	4 486.7	3.42%	6 место	Нестабильные изменения
5	Мысль	7 547.5	6 587.7	5 896.2	5 987.0	5 987.0	5 987.0	37 992.4	28.96%	1 место	Неизменное состояние
6	Книжный двор	655.4	731.0	926.1	952.1	756.2	755.0	4 775.8	3.64%	5 место	Стабильное снижение
7	Книголюб	4 000.6	4 562.3	5 698.2	6 549.2	6 987.5	7 256.2	35 054.0	26.72%	2 место	Стабильный рост
	Итого							**131 199.7**	100.00%		

Рис. 5.57. Выручка сети книжных магазинов

- в столбце **Тенденция за последние 3 года** должны быть следующие слова:
  - ◊ **Стабильный рост**, если выручка за шестой год больше выручки за пятый, а выручка пятого больше, чем четвертого;
  - ◊ **Стабильное снижение**, если выручка за шестой год меньше выручки за пятый, а выручка пятого меньше, чем четвертого;
  - ◊ **Неизменное состояние**, если выручка на протяжении последних трех лет неизменна;
  - ◊ **Нестабильные изменения** — во всех остальных случаях;
- в строке **Итого** — просуммируйте соответствующие столбцы.

5. На лист **Себестоимость** занесите и оформите вторую таблицу (рис. 5.58).

Себестоимость книжной продукции издательства ЭКОМ

№ п/п	Магазин	Годы						Всего за 6 лет	Среднегод. себе-стоимость	Макси-мальная себе-стоимость	Мини-мальная себе-стоимость
		1994	1995	1996	1997	1998	1999				
1	Дом книги	1356.1	1452	2323	2584.5	2563.2	2586.1	12 864.9	2 144.2	2 586.1	1 356.1
2	Книжным мир	1523.5	1675.2	2153	2256.2	2464.3	2756.2	12 828.4	2 138.1	2 756.2	356.2
3	Знание	456.2	356.2	563	563	456.1	412.1	2 806.6	467.8	563.0	257.3
4	Наука	257.3	520	636	763.1	915.2	612.1	3 703.7	617.3	915.2	257.3
5	Мысль	6547.5	5624.7	4563.2	4423	4213.1	3918.1	29 289.6	4 881.6	6 547.5	545.4
6	Книжный двор	545.4	631	736.1	823.1	615.1	918.4	4 269.1	711.5	918.4	545.4
7	Книголюб	2664.6	3651.3	4668.2	4789.2	4987.3	5123	25 883.6	4 313.9	5 123.0	2 664.6
	Итого	**15344.6**	**15905.4**	**17638.5**	**18199.1**	**18212.3**	**18325**	**91 645.9**	**15 274.3**	**19 409.4**	**5 982.3**

Рис. 5.58. Себестоимость

Произведите следующие расчеты:

- в столбце **Всего за 6 лет** — суммирование по строке;
- в столбце **Среднегодовая себестоимость** — вычислите среднее значение;

- в столбцах **Максимальная себестоимость** и **Минимальная себестоимость** воспользуйтесь функциями **МАКС** и **МИН**;
- в строке **Итого** — просуммируйте соответствующие столбцы.

6. На листе **График** постройте график выручки по годам для всех магазинов (рис. 5.59).

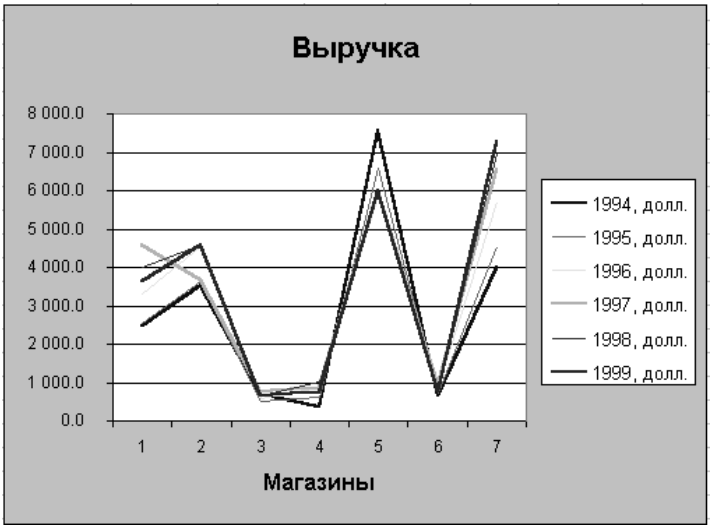


Рис. 5.59. График выручки магазинов

7. На лист **Прибыль** занесите и оформите третью таблицу (рис. 5.60).

Прибыль от реализации книжной продукции издательства ЭКОМ

Магазин	Годы						Всего за 6 лет	Доля в общей выручке, %
	1994	1995	1996	1997	1998	1999		
Дом книги	1100	1110	1000	1980	1420	1068.2	7678.2	19.41%
Книжным мир	1996.5	1970	2400	1400	2100	1808.2	11674.7	29.52%
Знание	220	169	26	193	198	232	1038	2.62%
Наука	96	112	227	100	108	140	783	1.98%
Мысль	1000	963	1333	1564	1773.9	2068.9	8702.8	22.00%
Книжный двор	110	100	190	129	141.1	-163.4	506.7	1.28%
Книголюб	1336	911	1030	1760	2000.2	2133.2	9170.4	23.18%
Итого	**7852.5**	**7330**	**8202**	**9123**	**9739.2**	**9286.1**	**39553.8**	**100.00%**

Рис. 5.60. Прибыль магазинов

- "Прибыль" рассчитывается по формуле: "Выручка" – "Себестоимость";
- "Всего" и "Долю" рассчитайте аналогично первой таблице.

- На этом же листе расположить две диаграммы:
  - ◊ круговую объемную — отражающую долю магазинов в общей прибыли;
  - ◊ объемную нормированную гистограмму для отражения прибыли магазинов по годам.

8. Сохраните работу в своей папке под именем Books.xls.

Практикум 47. Работники бюджетной сферы

Возможности электронных таблиц Excel позволяют автоматизировать выполнение многих экономических и бухгалтерских операций, поэтому вам предлагается создать электронную таблицу, позволяющую автоматизировать начисление заработной платы работникам бюджетной сферы.

Для выполнения задания строго следуйте нижеприведенному алгоритму.

1. Заработная плата работников бюджетной сферы исчисляется на основе Единой тарифной сетки, утвержденной Правительством Российской Федерации (табл. 5.2).

Таблица 5.2. *Единая тарифная сетка*

Разряды оплаты труда	Тарифные коэффициенты	Разряды оплаты труда	Тарифные коэффициенты
1	1,0	10	3,99
2	1,3	11	4,51
3	1,69	12	5,1
4	1,91	13	5,76
5	2,16	14	6,51
6	2,44	15	7,36
7	2,76	16	8,17
8	3,12	17	9,08
9	3,53	18	10,07

На листе 2 электронных таблиц Excel расположите вышеприведенную таблицу, а также таблицу с константами, которые будут необходимы для расчета заработной платы (табл. 5.3).

Таблица 5.3. *Константы*

Ставка первого разряда	500 руб.
Минимальная заработная плата	300 руб.
Ставка социального налога	38%
Прожиточный минимум в Санкт-Петербурге	1544 руб.

2. Лист 2 переименуйте, назвав его `Константы`. Перейдите на лист 1 и переименуйте его, назвав `Зарплата`, остальные листы книги удалите.
3. На листе **Зарплата** сформируйте таблицу `Расчет заработной платы`, состоящую из столбцов, имеющих заголовки:
   - **№ п/п**;
   - **Фамилия, инициалы работника**;
   - **Должность**;
   - **Разряд**;
   - **Количество ставок**;
   - **Процент доплат**;
   - **Начисленная зарплата с районным коэффициентом**;
   - **Количество льгот по подоходному налогу**;
   - **Подоходный налог**;
   - **Удержания в Пенсионный фонд**;
   - **Заработная плата к выдаче**;
   - **Доплаты из социальных фондов**.
4. Заполните составленную таблицу на основании следующих данных:
   - Федоров А. В., терапевт, имеет 13-й разряд, работает на 1,5 ставки, установлена доплата 15% за выполнение общественных обязанностей, 1 льгота по подоходному налогу;
   - Терехина В. М., повар, работает по 2-му разряду, имеет 2 ставки, доплат не имеет, 1 льгота по подоходному налогу;
   - Иванова А. В., санитарка, имеет 7-й разряд, 1,25 ставки, установлена доплата в размере 15% за вредные условия труда, 1 льгота по подоходному налогу;
   - Борщев А. Н., программист, имеет 11-й разряд, 1 ставка, доплата в размере 12% за работу с излучающей аппаратурой, 2 льготы по подоходному налогу.

5. В заполненной таблице рассчитайте путем задания формул:
   - начисленную зарплату с районным коэффициентом: (Тарифный коэффициент соответствующего разряда (с листа **Константы**) × Ставка первого разряда (с листа **Константы**) × Количество ставок + Величина доплат) × 1,15 (умножение всей суммы на 1,15 дает поправку на районный коэффициент, который составляет 15%);
   - подоходный налог: (Начисленная зарплата с районным коэффициентом − Минимальная зарплата (с листа **Константы**) × Количество льгот по подоходному налогу) × 0,13;
   - удержания в Пенсионный фонд: 1% от Начисленной зарплаты с районным коэффициентом;
   - заработную плату к выдаче: Начисленная зарплата с районным коэффициентом − Подоходный налог − Удержания в Пенсионный фонд;
6. Столбец **Доплаты из социальных фондов** заполните по условию:
   - если Зарплата к выдаче меньше прожиточного минимума (с листа **Константы**), то Доплата из социальных фондов равна: Прожиточный минимум − Зарплата к выдаче;
   - если наоборот, то Доплата из социальных фондов равна нулю.
7. К ячейкам столбца **Зарплата к выдаче** примените условное форматирование: Если Зарплата к выдаче меньше прожиточного минимума, то ячейка должна закрашиваться в красный цвет.
8. В построенной таблице подсчитайте общую сумму платежей по Подоходному налогу, общую сумму удержаний в Пенсионный фонд и общую величину Зарплаты к выдаче.
9. Отформатируйте ячейки, содержащие цифровые данные, измеряемые в денежном выражении, с точностью до 0,01 (до двух знаков после запятой).
10. По данным заполненной таблицы постройте столбиковую диаграмму, которая бы отображала фамилии сотрудников и величину заработной платы к выдаче.
11. Выполненную работу сохраните, назвав файл Money.xls.

Практикум 48. Директор кинотеатра "Аврора"

Предположим, вам предлагают стать директором кинотеатра "Аврора", но, сколько денег вы будете зарабатывать, говорить не спешат. Тогда вы решаете провести свое маркетинговое исследование (рис. 5.61).

1. Заполните зал (13 рядов по 25 мест в каждом) случайными цифрами от 1 до 4, кроме последнего ряда, который заполните цифрами 5. Цифры 1 и 2 будут обозначать посетителей, на которых распространяются скидки (С). Цифры 3 и 4 — обычных посетителей (О), а цифры 5 — "места для

поцелуев" (Л, от англ. Loveseat). Сделайте так, чтобы после заполнения в ячейках рассадки зала остались только сами цифры, а не функция **СЛЧИС()**.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB		
1	**Кинотеатр "АВРОРА"**																													
2													**Сеанс**							**Цена**										
3																		С		О		Л					***курс $=***	***31.5***		
4	"Сказ про Федота-стрельца"												10.40					30		70		200								
5	"Властелин колец"												12.30					50		70		200					Всего зрителей	*ф4*		
6	"Властелин колец"												15.40					70		100		300					***Из них:***			
7	"13 привидений"												18.50					100		120		350					со скидкой	*ф1*		
8	"Властелин колец"												20.30					150		150		400					обычных	*ф2*		
9																											LoveSeat	*ф3*		
10	**Рассадка зала**																													
11	2	1	1	1	2	4	4	1	4	1	1	2	1	2	4	1	2	3	1	3	3	1	1	1	2		***Выручка:***			
12	4	3	3	4	3	1	4	2	2	3	1	4	2	1	4	3	3	3	3	4	2	1	4	1	2		сеанс 10.40	*ф5*		
13	4	4	1	1	4	3	3	1	1	2	1	4	3	2	4	4	3	4	2	3	4	3	4	4	3		сеанс 12.30	*ф6*		
14	1	4	3	4	2	4	1	1	3	1	2	1	1	3	2	3	4	2	2	1	4	4	1	2	3		сеанс 15.40	*ф7*		
15	3	2	3	3	3	4	3	3	2	1	3	4	4	1	2	3	3	3	3	2	4	2	3	3	1		сеанс 18.50	*ф8*		
16	1	3	4	2	1	4	4	1	1	1	3	4	2	1	4	3	2	3	1	2	4	3	2	4	1		сеанс 20.30	*ф9*		
17	1	1	4	2	2	2	4	3	1	4	4	2	3	3	3	1	2	3	1	2	2	3	1	2	3		*Итого за день:*	*ф10*		
18	3	4	3	1	2	4	2	1	1	2	1	1	3	2	1	3	4	3	2	4	4	2	2	1	1		*Итого за месяц:*	*ф11*		
19	4	1	4	1	3	1	4	2	4	3	3	3	3	2	1	2	4	3	1	3	2	4	2	1	1		***Из них:***	%	в рублях	в $
20	2	2	1	3	2	4	1	4	4	4	2	3	2	4	4	1	1	2	4	2	1	1	3	2	1		амортизация	12	*ф12*	*ф18*
21	4	2	1	3	1	1	4	3	3	4	3	4	3	2	3	1	2	3	2	4	3	3	2	1	1		налоги	22	*ф13*	*ф19*
22	4	2	3	1	1	2	2	2	4	3	3	3	1	1	1	1	1	4	4	3	4	2	1	3	4		з/п сотрудникам	16	*ф14*	*ф20*
23	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		охранные структуры	25	*ф15*	*ф21*
24																											новые фильмы	22	*ф16*	*ф22*
25	Согласны ли Вы, Фамилия Имя, стать директором кинотеатра?																										директор	3	*ф17*	*ф23*
26	*ф24*																													

Рис. 5.61. Кинотеатр "Аврора"

2. Формулы ф1— ф3 при помощи функции **СЧЕТЕСЛИ** подсчитывают в зале:
   - количество посетителей 1 и 2, т. е. количество льготников;
   - количество 3 и 4, т. е. обычных посетителей;
   - количество 5, т. е. влюбленных.
3. Формула 4 суммирует полученные числа, чтобы получить контрольную цифру 325.
4. Формулы ф5—ф9, исходя из известной стоимости билетов и подсчитанного количества зрителей, вычисляют выручку соответствующего сеанса. Предполагаем, что рассадка зала на всех сеансах одинакова.
5. Формула ф10 вычисляет выручку за день, формула ф11 — за месяц (принимаем, что в месяце 30 дней).
6. Исходя их выручки в рублях за месяц и процентов расходов по статьям, формулы ф12—ф17 вычисляют денежные расходы на соответствующие статьи затрат.

7. Формулы ф18—ф23 показывают, сколько эти суммы будут значить в долларах при известном заранее курсе.
8. Формула ф24 определяет, что если директор в результате получает больше 3000 долларов в месяц, то мы согласны быть директором, т. е. в ячейке **A26** должно появиться слово "Да" или "Нет". Это и будет итогом работы.
9. Постройте столбиковую объемную диаграмму расходов. Столбики должны быть в виде цилиндров.
10. Сохраните работу под именем Director.xls.

Практикум 49. Антреприза

Задание, аналогичное *Практикуму 48*, но более самостоятельное.

Исходные данные:

Московский театр "Ленком" привез в Санкт-Петербург антрепризный спектакль Олега Меншикова "Ревизор". Гастроли проходят в помещении БДТ.

Спектакль идет два вечера.

Цены на билеты:

- Партер — 1500 руб.
- Места за креслами — 1000 руб.
- Бельэтаж — 800 руб.
- 2-й ярус — 500 руб.
- 3-й ярус — 300 руб.

1. Представьте зал в виде квадрата 10×10, заполненного случайными цифрами от 0 до 9, которые обозначают:
   - цифры 1, 2, 3, 4 и 5 — зрителей партера;
   - цифра 6 — места за креслами;
   - 7 и 8 — бельэтаж;
   - 9 — 2-й ярус;
   - 0 — 3-й ярус;

   Каждая цифра обозначает 10 зрителей (всего их должно быть 1000).

   Курс доллара примите равным 31 руб. за 1 доллар.

   Курс евро примите равным 27 руб. за 1 евро.
2. Вычислите выручку за оба вечера в рублях, евро и долларах.
3. Произведите расчет отчислений в рублях, евро и долларах, если известно, что:
   - на аренду помещения уходит 5% выручки;
   - на организацию гастролей 25%;

- на рекламу 10%;
- на налоги 13%;
- на зарплату актерам 20%;
- на зарплату обслуживающему персоналу 7%;
- транспортные расходы 3%;
- непредвиденные расходы 2%;
- продюсеру — 15%.

4. Оформите работу в соответствии с рекомендациями *Практикума 48*.
5. Аналогично, используя функцию **ЕСЛИ**, согласитесь быть продюсером только при получении дохода более 5000 долларов.
6. Постройте круговую объемную диаграмму затрат.
7. Сохраните документ под именем Bdt.xls.

Практикум 50. Царь-пушка

Выполните теперь уже в Excel задание "Царь-пушка" из *главы 4*. В электронных таблицах оно решается гораздо проще, чем на любом языке программирования.

5.2.5. Макросы в Excel

Практикум 51. Макросы на складе

Создать таблицу как на рис. 5.62.

	A	B	C	D	E	F	G	H	I	J
1		**Дата:**		*03 апреля 2002 года.*						
2		**Дежурный по складу:**		*ФИО*						
3			Было:				Цена продажи		Цена покупки	
4	№ поз.	Имеется	кол-во(шт)	Продано шт.	Куплено	Стало	за 1 штуку	Стоимость	за 1 штуку	Стоимость
5	1	Шайба	2400			2400	0,03	=D5*G5	0,02	=E5*I5
6	2	Винт	120			120	0,05	=D6*G6	0,03	=E6*I6
7	3	Гайка	80			80	0,03	=D7*G7	0,02	=E7*I7
8	4	Уховерт	50			50	4,5	=D8*G8	4	=E8*I8
9	5	Ключ	24			24	3,8	=D9*G9	3,3	=E9*I9
10	6	Гвоздь 200 мм	4000			4000	0,02	=D10*G10	0,01	=E10*I10
11										
12		**Было в кассе**								
13		**утром**	**150**							
14		**Напродавали**								
15		**Напокупали**								
16		**Стало**	**=C13+C14-C15**							

Рис. 5.62. Склад

Цель: макрос должен после занесения данных о проданных и купленных изделиях автоматически вычислять, сколько изделий стало на складе, а также сумму продаж и покупок и наличность в кассе.

Макрос создаем по следующему подробному словесному алгоритму.

1. Выделить базовую ячейку (**B11**).
2. Выбрать команду меню **Сервис | Макрос | Начать запись** (дать макросу имя `Касса`, назначить "горячую" клавишу).
3. Занести в ячейку **D11** формулу `=F5–D5`.
4. Скопировать ее в интервал **D11:D16**.
5. Скопировать этот интервал в буфер обмена.
6. Курсор поместить в ячейку **F5**.
7. Выбрать команду меню **Правка | Спецвставка | Значения | OK**.
8. Занести в ячейку **E11** формулу `=F5+E5`.
9. Скопировать ее в интервал **E11:E16**.
10. Скопировать этот интервал в буфер обмена.
11. Курсор поместить в ячейку **F5**.
12. Выбрать команду меню **Правка | Спецвставка | Значения | OK**.
13. Занести в ячейку **H11** формулу `=сумм(H5:H10)`.
14. Занести в ячейку **H12** формулу `=H11+C14`.
15. Курсор поместить в ячейку **H12**.
16. Скопировать ее в буфер обмена.
17. Курсор поместить в ячейку **C14**.
18. Выбрать команду меню **Правка | Спецвставка | Значения | OK**.
19. Занести в ячейку **J11** формулу `=сумм(J5:J10)`.
20. Занести в ячейку **J12** формулу `=J11+C15`.
21. Курсор поместить в ячейку **J12**.
22. Скопировать ее в буфер обмена.
23. Курсор поместить в ячейку **C15**.
24. Выбрать команду меню **Правка | Спецвставка | Значения | OK**.
25. Выделить интервал **D5:E16**.
26. Очистить.
27. Выделить интервал **H11:J16**.
28. Очистить.
29. Остановить запись макроса.

30. Проверить работу макроса.
31. В случае ошибок повторить запись, если все нормально, сдать работу преподавателю.
32. Сохранить документ под именем Macros1.xls.

Практикум 52. Макросы в статистике

1. Построить таблицу, содержащую следующие столбцы (в начале укажите себя в качестве главного демографа):
   - **№**.
   - **Название страны**.
   - **Столица**.
   - **Численность населения на 8.00 сегодняшнего числа (в тыс. чел.)**.
   - **Рождаемость (тыс. чел./сутки)**.
   - **Смертность (тыс. чел.сутки)**.
   - **Иммиграция (тыс. чел./сутки)**.
   - **Эмиграция (тыс. чел/сутки)**.
   - **Численность населения на 8.00 следующего дня (в тыс. чел)**.

   Длинные названия столбцов — в 2 строки, шрифт 12.

   Изначально столбцы 4 и 9 содержат одинаковые сведения, столбцы 5—8 сведений не содержат.
2. Напишите макрос, который после занесения данных о рождаемости, смертности, иммиграции и эмиграции, подсчитал бы численность населения на 8.00 следующего утра, а также суммарную сводку о рождаемости, смертности, иммиграции и эмиграции за сутки, если данные проступают в демографический центр три раза в сутки — в 16.00, в 00.00 и в 8.00.
3. Сохраните работу под именем Makros2.xls.

Практикум 53. Последняя работа в Excel

1. На листе 1 создайте таблицу "Класс".
   - Переименуйте лист 1 в `Класс`.
   - Параметры страницы — ориентация альбомная, поля: левое 2 см, правое 1 см, верхнее 1,5 см, нижнее 1,5 см.
   - Вставьте картинку (по выбору).
   - Дайте таблице заголовок `Мой любимый 10а(б) класс` (размер 14, полужирный курсив, двойное подчеркивание).

- Введите графы таблицы (размер 13, Arial, полужирный, по центру, цвет — темно-синий, заливка — желтый, обрамление шапки — двойной линией).
  - ◊ `Номер по порядку`;
  - ◊ `Фамилия`;
  - ◊ `Имя`;
  - ◊ `Отчество`;
  - ◊ `Дата рождения`;
  - ◊ `Рост (см)`;
  - ◊ `Вес (кг)`.
- Заполните содержание таблицы: размер 12, обычная сетка, свои ФИО: размер 14, курсив. **Номер по порядку**, **Дата рождения**, **Вес и Рост** — выравнивание по центру.
- Под таблицей вычислите по формуле: средний рост, средний вес, общий рост, общий вес.
- Сохраните под именем Class.doc.

2. На листе 2 создайте таблицу "Котельная ".
   - Переименуйте лист 2 в `Котельная`.
   - Параметры страницы — ориентация книжная, поля: левое 2 см, правое 1,5 см, верхнее 2 см, нижнее 2 см.
   - Вставьте картинку (по выбору).
   - Дайте таблице заголовок `Моя любимая котельная` (размер 14, полужирный курсив, двойное подчеркивание).
   - Ниже занесите `Главный кочегар`: — ваши личные ФИО (размер 13, Arial, полужирный).
   - Ниже введите `Потребление угля в день при t>[G.D.8]—20°C 20 тонн` (цифра 20 в отдельной ячейке!).
   - Введите графы таблицы (размер 13, Arial, полужирный, по центру, цвет — темно-синий, заливка — желтый, обрамление шапки — двойной линией):
     - ◊ `День недели`;
     - ◊ `t°C`;
     - ◊ `t°F`;
     - ◊ `Расход угля (т)`.
   - Заполните содержание таблицы: размер 12, обычная сетка, числовые данные по центру (табл. 5.4).

Таблица 5.4. *Котельная*

День недели	t°C	t°F	Расход угля (т)
Понедельник	–19		
Вторник	–22		
Среда	–25		
Четверг	–31		
Пятница	–47		
Суббота	–32		
Воскресенье	–15		

- Температуру по Фаренгейту вычислите по формуле из следующего соотношения: $t \times C = 5/9(t \times F - 32)$.
- Расход топлива вычисляется при помощи функции **ЕСЛИ**:
  - ◊ при температуре от −20 до −30 по Цельсию потребление топлива увеличивается на 5 тонн;
  - ◊ при понижении температуры ниже −30 по Цельсию — еще на 7 тонн.

3. Вычислите среднее потребление топлива за сутки, суммарное потребление и постройте под таблицей график расхода топлива за неделю.
4. Сохраните работу под именем Zachet.xls в своей папке с паролем.

Тест 1. Знаешь ли ты электронные таблицы

1. Относительная ссылка это:

   а) когда адрес, на который ссылается формула, изменяется при копировании формулы;

   б) когда адрес, на который ссылается формула, при копировании не изменяется;

   в) ссылка, полученная в результате копирования формулы;

   г) ссылка не в Сибирь, а в Швейцарию (дорев.).

2. В вашем распоряжении имеется громоздкая таблица, где требуется одновременно просмотреть столбцы, расположенные далеко друг от друга. Как это сделать, не нарушая структуры таблицы?

   а) уменьшить масштаб на экране так, чтобы на нем присутствовали все нужные столбцы;

   б) скрыть ненужные промежуточные столбцы с помощью команды **Формат | Столбец | Скрыть**;

в) скопировать нужные столбцы на другой лист и просмотреть их там;

г) увеличить ширину экрана монитора.

3. Обычно, при написании формул используются данные, расположенные в нескольких ячейках, так называемый диапазон ячеек, который выглядит в строке формул следующим образом:

   а) **A1\B3**;

   б) **A1+B3**;

   в) **A1:B3**;

   г) **A1—B3**.

4. После ввода числа в ячейку вы наблюдаете `######` вместо результата. В чем причина такой ситуации?

   а) не хватает ширины клетки, чтобы показать введенное число;

   б) число введено с ошибкой;

   в) число введено в защищенную ячейку;

   г) получилось отрицательное число.

5. Вы вводите в ячейку данные в виде `01/01/00`. Какая дата будет храниться в этой клетке?

   а) 1-е января 1900-го года;

   б) 1-е января 2000-го года;

   в) число 1,0100;

   г) компьютер выведет сообщение об ошибке.

6. Вы защитили на листе ячейки с формулами, но оставили незащищенными клетки с исходными данными для них. Изменятся ли результаты расчета формул при изменении ячеек с исходными данными?

   а) да, изменятся;

   б) это зависит от типа формул;

   в) нет, не изменятся;

   г) компьютер выведет сообщение об ошибке.

7. Для переименования листа в Excel требуется:

   а) сохранить данные на диске;

   б) щелкнуть левой кнопкой мыши на листе и ввести новое имя;

   в) щелкнуть правой кнопкой мыши на листе и изменить имя;

   г) дважды щелкнуть на имени листа и изменить имя.

8. Вы построили диаграмму по ряду данных из таблицы, а через некоторое время изменили эти данные. Как перестроить диаграмму для новых данных?

   а) достаточно один раз щелкнуть мышью на диаграмме;

   б) достаточно дважды щелкнуть мышью на диаграмме;

   в) пересчет диаграммы в стандартном режиме произойдет автоматически;

   г) заново построить диаграмму.

9. Вы часто строите однотипные диаграммы нестандартного вида для различных данных из разных таблиц. Каким образом зафиксировать этот нестандартный вид диаграммы и использовать его, не оформляя каждый раз подобную диаграмму заново?

   а) сохранить все поля одной из диаграмм как отдельные графические объекты и вставлять их всякий раз в новые диаграммы;

   б) при создании очередной диаграммы копировать какую-либо старую диаграмму и переопределять в ней область исходных данных;

   в) оформить диаграмму, затем сделать ее стандартной (выбрать команду меню **Диаграмма | Тип** или правой кнопкой мыши);

   г) настроить шаблон документа.

10. После ввода числа в ячейку вы наблюдаете ###### вместо результата. Как исправить ошибку?

    а) изменить ширину столбца;

    б) изменить высоту строки;

    в) исправить неточности в формуле;

    г) снять защиту с ячейки.

11. В вашей таблице имеется столбец с названиями месяцев, расположенными в случайном порядке. Как отсортировать строки такой таблицы, чтобы названия месяцев в указанном столбце шли по порядку, а не по алфавиту?

    а) названия месяцев при сортировке автоматически ставятся по порядку, а не по алфавиту;

    б) необходимо щелкнуть на заголовке столбца с месяцами, а затем воспользоваться кнопкой **Сортировка по возрастанию**;

    в) необходимо щелкнуть на заголовке столбца с месяцами и воспользоваться командой **Данные | Сортировка**, указав в ее параметрах порядок сортировки по месяцам;

    г) это невозможно.

12. Вы хотите построить в Excel сводную таблицу на основе сведений из внешней базы данных. Какая из команд поможет решить эту проблему?

    а) команда **Файл | Открыть**, где в поле **Тип файла** указан тип файлов нужной базы данных;

    б) команда **Вставка | Объект**;

    в) команда **Данные | Сводная таблица**;

    г) команда **Данные | Внешние данные**.

13. Как наиболее простым способом получить на листе Excel в пригодном для расчетов виде данные из таблицы, созданной средствами Word?

    а) набрать все данные на листе Excel вручную;

    б) воспользоваться командой **Вставка | Объект** и указать файл с таблицей Word;

    в) в Word выделить таблицу и воспользоваться командой **Копировать**, в Excel воспользоваться командой **Вставить**;

    г) воспользоваться командой **Данные | Внешние данные**.

14. Для обозначения адреса ячейки электронной таблицы используются:

    а) буквы латинского алфавита и цифры;

    б) буквы русского и латинского алфавита;

    в) только русские буквы и цифры;

    г) специальные символы.

15. Исходными данными в электронных таблицах называют:

    а) числа, получаемые в результате расчетов;

    б) числа, вводимые в ячейки таблицы;

    в) используемую для вычислений информацию, вводимую в ячейки таблицы;

    г) защищенные ячейки.

16. Формула это:

    а) связь между исходными и рассчитываемыми данными;

    б) адреса ячеек и знаки арифметических операций;

    в) буквы и цифры, обозначающие адреса ячеек, и знаки арифметических операций;

    г) набор стандартных констант.

17. При составлении (написании) формул используются:

    а) числа, находящиеся в ячейках с исходными данными;

    б) адреса ячеек, хранящие исходные данные;

в) русские буквы и цифры, обозначающие адреса ячеек;

г) надбуквенные символы.

18. В связи с разбиением электронной таблицы на столбцы и строки каждая ячейка:

а) имеет свой собственный оригинальный адрес;

б) помещается в определенную позицию;

в) может использоваться для хранения данных или формул;

г) может копироваться в буфер обмена.

19. Внутри ячеек электронной таблицы могут находиться следующие типы данных:

а) только числа, текст и рисунки;

б) только числа и формулы;

в) числа, формулы, текст, рисунки;

г) ни один из перечисленных объектов.

20. При изменении исходных данных все результаты:

а) автоматически пересчитываются;

б) заносятся в память компьютера;

в) сохраняются на диске;

г) попадают в буфер обмена.

21. Требуется написание одинаковых формул для нескольких ячеек одного столбца. Ваши действия:

а) в каждой ячейке ввести формулу вручную;

б) воспользоваться при составлении формул щелчками по клеткам с данными, чтобы не писать адреса;

в) написать формулу и "растянуть" ее с помощью маркера заполнения на необходимое число строк;

г) использовать Мастера функций.

22. Абсолютная ссылка это:

а) когда адрес, на который ссылается формула, изменяется при копировании формулы;

б) когда адрес, на который ссылается формула, при копировании не изменяется;

в) ссылка, полученная в результате копирования формулы;

г) ссылка в Сибирь.

23. Для построения диаграммы с минимальным количеством шагов необходимо:

    а) выделить диапазон ячеек, щелкнуть на ярлыке Мастера диаграмм;

    б) щелкнуть на ярлыке Мастера диаграмм и ответить на вопросы, указав диапазон ячеек;

    в) воспользоваться помощником;

    г) воспользоваться справкой.

24. Требуется расположить текст (данные) в ячейке таким образом, чтобы разные слова переходили на следующую строку. Ваши действия:

    а) изменить ширину столбца так, чтобы текст переходил на следующую строку;

    б) набрать данные в разных строках;

    в) выделить, выбрать команду меню **Формат | Ячейка | Выравнивание | Переносить по словам**;

    г) нажать <Enter>.

25. Для перехода на другой лист в Excel требуется:

    а) щелкнуть указателем мыши на листе;

    б) воспользоваться вертикальной прокруткой;

    в) воспользоваться горизонтальной прокруткой;

    г) воспользоваться меню **Окно**.

26. Требуется напечатать только выделенный диапазон ячеек имеющейся таблицы. Ваши действия:

    а) выбрать команду **Файл | Печать | Выделенный диапазон**;

    б) уменьшить масштаб (команда **Файл | Печать |** указать %);

    в) скопировать нужные ячейки на другой лист и распечатать;

    г) удалить все ненужное и выбрать команду **Файл | Печать**.

27. Как применить стандартные решения при оформлении таблицы?

    а) выделить диапазон ячеек, выбрать команду **Формат | Ячейки**;

    б) выделить диапазон ячеек, выбрать команду **Формат | Автоформат**;

    в) открыть ранее подготовленную таблицу и ввести новые данные и формулы;

    г) воспользоваться автозаменой.

28. Как связать данные расположенные на разных листах?

    а) использовать команду **Правка | Специальная вставка**;

    б) при написании формулы перейти на другой лист и щелкнуть на ячейке с данными;

в) использовать стандартную формулу (функцию);

г) они не могут быть связаны.

29. В клетке **C2** находится формула, имеющая ссылку на постоянную величину в ячейке **D2**. Как размножить формулу из клетки **C2** так, чтобы ее копии (**C3...C8**) использовали ссылку на **D2**?

    а) защитить клетку **D2** от изменений и скопировать формулу в нужные клетки;

    б) скопировать формулу в нужные клетки и отредактировать каждую копию;

    в) изменить в оригинале формулы адрес **D2** на адрес **D2** и скопировать формулу в нужные клетки;

    г) воспользоваться автозаменой.

30. Рассчитываемыми данными в электронных таблицах называют:

    а) числа, получаемые в результате расчетов;

    б) числа, вводимые в ячейки таблицы;

    в) формулы, по которым производятся вычисления;

    г) стандартные функции.

5.3. Обработка структурированной информации (базы данных)

5.3.1. Основные понятия баз данных

Иметь представление о базах данных безусловно надо, хотя бы потому, что на первых курсах вузов их изучают довольно детально.

По сути базы данных можно представить в виде больших таблиц, содержащих структурированные сведения об однотипных объектах.

Пример 1. Сведения об учащихся

Как известно, на каждого учащегося в конце классного журнала имеется мини-досье со следующими данными:

- № по порядку;
- фамилия;
- имя;
- отчество;
- дата рождения;

- домашний адрес;
- телефон;
- фамилия, имя, отчество мамы;
- место работы, рабочий телефон.

Эти мини-досье представлены в виде приблизительно такой таблицы (рис. 5.63).

№	Фамилия	Имя	Отчество	Дата рождения	Дом.адрес	Телефон	ФИО мамы	Место работы Раб.телефон
1	Орлов	Лев	Николаевич	12.12.88	Крылова, 23-41	230-67-81	Эльвира Ивановна	ОАО "ВМФ" 304-45-90
2	Дуров	Игорь	Петрович	17.01.89	Усова, 54-90	456-89-32	Анна Ивановна	Школа №890 689-01-02
3	Бауэр	Юрий	Сергеевич	30.06.87	Шаумяна, 11-54	765-47-78	Ирина Андреевна	ЗАО "DEEP" 256-40-13
4	Данилов	Кирилл	Алексеевич	04.03.88	Дыбенко, 13-39	578-27-10	Нина Анатольевна	Гор.суд 222-02-02
5	Ножкина	Любовь	Андреевна	08.03.89	Широкая, 25-40	231-09-55	Кира Геннадьевна	Детсад №45 346-34-23

Рис. 5.63. Мини-досье

А ведь можно еще добавить вес и рост, основные черты характера, провинности за первый класс и много-много другого.

Вот такой столбец, отражающий какую-нибудь одну характеристику или параметр рассматриваемого объекта, называется *полем* базы данных.

Основные типы данных, которые могут содержаться в полях, приведены в табл. 5.5.

Таблица 5.5. *Типы данных*

Тип данных	Характеристика
Текстовый	Строка текста (до 255 символов)
Поле MEMO	Текст в несколько строк (до 65 535 символов)
Числовой	Число любого типа
Дата/время	Дата или время
Денежный	Поле, выраженное в заданных денежных единицах
Счетчик	Поле, автоматически нумерующее записи
Логический	Принимает значение только `TRUE` (истина) или `FALSE` (ложь)

Таблица 5.5 *(окончание)*

Тип данных	Характеристика
Поле объекта OLE	Может содержать рисунки, звуковые файлы, документы Microsoft Office и т. д.
Гиперссылка	Ссылка, связывающая это поле с каким-либо файлом

Ну что же, полученных на теоретических занятиях сведений вполне достаточно, чтобы поупражняться. Сначала совсем простые базы данных.

Практикум 54. Конструирование и работа с простыми базами данных

Вариант 1

Разработайте базу данных, содержащую оценки 10-ти учащихся по пяти предметам и вычислить средний балл учащихся. Осуществите выборку всех учащихся, средний балл которых попадает в диапазон от 4,2 до 4,7 баллов.

Вариант 2

Составьте базу данных "Записная книжка", в которой представлены адреса и телефоны десяти родственников, друзей и знакомых. Составьте запрос, который по фамилии человека определял бы его адрес и телефон.

Вариант 3

Составьте базу данных "Домашняя библиотека", в которой представлены 10 книг по разделам, например фантастика, детективы и т. д. Составьте запрос, который по фамилии автора определял бы, какие книги этого автора есть в библиотеке и к какому разделу они относятся.

Вариант 4

Составьте базу данных "Телефонный справочник", в которой представлены адреса и телефоны десяти оптово-закупочных фирм, строительных фирм и т. д. Составьте запрос, который по названию фирмы определял бы ее адрес и телефон.

Вариант 5

Составьте базу данных "Телефонный справочник", в которой представлены адреса и телефоны магазинов, парикмахерских, библиотек и т. д. Составьте запрос, который по номеру магазина (или парикмахерской) определял бы его адрес и телефон.

Вариант 6

По сведениям о планетах Солнечной системы составьте базу данных "Планеты Солнечной системы" (табл. 5.6). Составьте запрос на выборку сведений о планетах, масса которых меньше массы Земли.

Таблица 5.6. *Планеты Солнечной системы*

Планета	Среднее расстояние от Солнца (млн. км)	Период обращения (лет)	Масса (в массах Земли)	Радиус (км)	Число спутников
Меркурий					
Венера					
Земля					
Марс					
Юпитер					
Сатурн					
Уран					
Нептун					
Плутон					

Вариант 7

На продовольственном складе хранятся:

- яблоки — 1000 кг по цене 50 рублей;
- лимоны — 2000 кг по цене 90 рублей;
- мандарины — 1500 кг по цене 40 рублей;
- курага — 1200 кг по цене 60 рублей;
- бананы — 2000 кг по цене 20 рублей;
- огурцы — 3000 кг по цене 15 рублей;
- груши — 2500 кг по цене 40 рублей;
- клубника — 500 кг по цене 80 рублей;
- киви — 1000 кг по цене 50 рублей;
- картофель — 9000 кг по цене 12 рублей.

Создайте базу данных. Занесите эти данные в базу данных. Выведите информацию о яблоках (количество, стоимость 1 кг), а также стоимость всей продукции.

Вариант 8

На аптечном складе хранятся лекарства. Сведения о десяти лекарствах есть в специальной ведомости, которая содержит следующую информацию:

- наименование лекарственного препарата;
- количество (в шт.);

- цена (в руб.);
- срок хранения на складе (в месяцах).

Выясните, сколько стоит самый дорогой и самый дешевый препарат; сколько препаратов хранится на складе более 3 месяцев; сколько стоят все препараты, хранящиеся на складе.

Вариант 9

Торговый склад производит уценку хранящейся продукции. Если продукция хранится на складе дольше 10 месяцев, то она уценивается в 2 раза, а если срок хранения превысил 6 месяцев, но не достиг 10, то в 1,5 раза. Ведомость уценки товаров должна содержать информацию:

- наименование товара;
- количество товара (шт.);
- цена товара до уценки;
- срок хранения товара;
- цена товара после уценки;
- общая стоимость товаров после уценки.

Выясните максимальный и минимальный сроки хранения товаров на складе; максимальную и минимальную цены товаров до уценки и после уценки.

Вариант 10

Составьте базу данных видеотеки на 10 фильмов со следующей информацией о каждом:

- название;
- режиссер;
- страна-производитель;
- год выхода на экран;
- продолжительность (мин.);
- жанр (комедия, боевик, ужастик и т. д.).

Составьте запросы на комедии российского производства и на фильмы продолжительностью не более 100 минут.

Ну, а теперь ответьте на вопросы.

1. Какие действия можно выполнять с текстовыми данными, с числовыми данными в БД?

2. Какие существуют логические операции и как они выполняются в БД?
3. Каким образом (по каким командам) происходит поиск в БД?
4. В какой форме отражаются результаты поиска в БД?
5. Каким образом (по каким командам) происходит организация запроса к БД?
6. Что такое реляционная и иерархическая модели БД? Определите их достоинства и недостатки. Приведите примеры.

Практикум 55. База данных "Борей"

В пакет Microsoft Office входит система управления базами данных Access. А в нее, в свою очередь, — учебная база данных "Борей", которая открывается командой меню **Справка | Примеры баз данных**.

Так вот, прежде чем выполнять самостоятельный практикум по созданию и обработке базы данных, рекомендуем внимательно рассмотреть предлагаемую учебную базу по следующему плану:

1. Изучите интерфейс СУБД Access на примере БД "Борей".
2. Поработайте с таблицами, запросами, фильтрами, формами, отчетами.
3. Ознакомьтесь с основными настройками СУБД и свойствами базовых объектов.

Практикум 56. Домашний зоопарк

1. Создайте новую базу данных под именем Zoo.mdb со следующей структурой (табл. 5.7).

Таблица 5.7. *Домашний зоопарк*

Наименование поля	Тип
Вид животного	Текстовый
Кличка	Текстовый
Дата приобретения	Дата/Время
Цена	Денежный (от 100 до 3000 €)
Пол	Логический

2. Создайте форму для ввода данных примерно следующего вида (рис. 5.64).

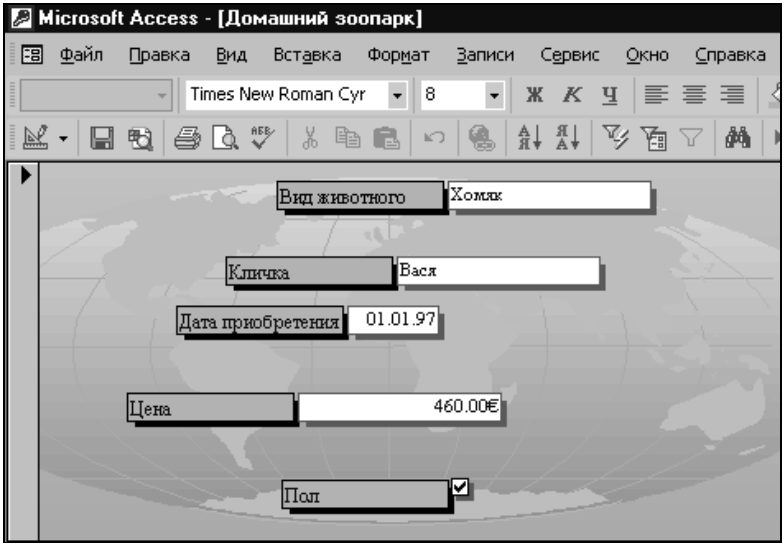


Рис. 5.64. Форма "Домашний зоопарк"

3. Заполните таблицу десятью данными.
4. Создайте в этом же файле еще одну таблицу (табл. 5.8).

Таблица 5.8. *Коэффициент*

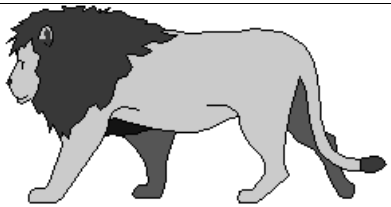
Наименование поля	Тип
Кличка	Текстовый
Возрастной ценз	Числовой (целое число от 1 до 10)
Коэффициент	Числовой (целая часть от деления цены на возрастной ценз)

5. Заполните таблицу десятью данными в соответствии с табл. 5.7.
6. Создайте в этом же файле еще одну таблицу (табл. 5.9).

Таблица 5.9. *Шанс продажи*

Наименование поля	Тип
Коэффициент	Числовой (из табл. 5.8)
Шанс продажи с аукциона (Коэффициент/3000×100, в %)	Числовой

7. Установите связи между табл. 5.7 и 5.8 по полю **Кличка**, а между табл. 5.8 и 5.9 — по полю **Коэффициент**.
8. Сделайте сводный запрос по трем таблицам по полям **Кличка**, **Цена**, **Возрастной ценз**, **Шанс продажи с аукциона** (условие выборки узнайте у преподавателя).
9. Перенесите сводный запрос в Microsoft Word. Нанесите шрифтовое и абзацное оформление, вставьте графический объект. Распечатайте полученный отчет. Пример оформления отчета приведен на рис. 5.65.



ЗАО «Домашний зоопарк»
ул. Восстания, д.4
тел. 275-90-56

Уважаемый Борис Николаевич!

Исходя из поставленных Вами условий, мы можем предложить Вам следующих животных из нашего зоопарка:

Вид животного	Кличка	Цена	Возрастной ценз	Шанс продажи
Медведь	Винни	660 €	3	7,3
Лев	Кинг	1 500 €	2	25
Питон	Мудрая	780 €	3	8,7
Конь	Педальный	2 000 €	2	33

С уважением

Генеральный директор **Козлов А.А.**

11 марта 2003 г.

Рис. 5.65. Пример оформления отчета

По просьбе преподавателя необходимо продемонстрировать следующие навыки и умения:

- уметь сохранить и открыть свою БД;
- создать и открыть любую таблицу из файла своей БД;
- отсортировать таблицу по заданному полю;
- удалить указанные записи в таблице;
- дополнить таблицу записями;
- изменить структуру БД;
- сделать запрос по заданному условию.

Практикум 57. Создание баз данных, имеющих практическое значение

Закончим мы тем, с чего начали — сведениями об учащихся в конце классного журнала. Как порой (и очень часто не вовремя!) администрации школы требуются бесконечные списки и какая-то информация о своих любимых учениках. Давайте же совместим приятное с полезным — воплотим в жизнь проект "База данных учащихся школы", куда должны войти полные сведения о каждом учащемся.

Такая база, будучи один раз создана, в дальнейшем может довольно легко поддерживаться секретарем школы, и не на словах, а на деле будет способствовать внедрению новых информационных технологий в школьную жизнь.

Практикум 58. Создание предметных баз данных

И еще о помощи школе. При освоении баз данных рекомендуется делать интегрированные проекты, которые в дальнейшем помогли бы в изучении различных школьных дисциплин.

Примеры:

- ❒ География — базы данных "Страны мира", "Моря, реки и озера", "Горы", "Улицы родного города" с указанием транспорта;
- ❒ Ботаника — базы данных "Растения моей области", "Растения моего сада";
- ❒ Химия — база данных "Химические элементы";
- ❒ История — базы данных "Правители России", "Величайшие полководцы", "Сражения русской армии";
- ❒ Литература — базы данных "Моя библиотека", "Поэты и писатели Серебряного века";
- ❒ Математика — базы данных "Великие математики от древности до наших дней", "Правильные многогранники".

и т. д., и т. п.

Все вышеперечисленные базы данных, будучи созданными и работающими, позволили бы в локальной сети решать комплексные задачи, примеры которых мы приведем ниже.

Ну, а в заключение раздела, как всегда, тест для проверки ваших знаний.

Тест 2. Базы данных

1. Системы управления базами данных — это:

   а) программное средство для автоматизации вычислений;

   б) программное средство для автоматизации хранения и поиска информации;

в) система для представления информационных массивов во внешней памяти компьютера.

2. СУБД позволяют выполнять следующие операции:

а) представлять информацию о предметах и явлениях реального мира в структурированной форме;

б) хранить информацию на внешних запоминающих устройствах компьютера;

в) передавать и получать информацию по телекоммуникационным каналам;

г) осуществлять поиск и отбор информации по заданным критериям;

д) выполнять сложные математические преобразования;

е) выполнять вычисления на основе информации, хранящейся в базе данных;

ж) объединять информацию, относящуюся к одним и тем же объектам и хранящуюся в разных базах данных;

з) выводить информацию из базы данных на печать;

и) строить и модифицировать графические объекты;

к) дополнять базу информацией о новых объектах;

л) редактировать информацию в базе данных.

3. Таблица данных содержит:

а) информацию о совокупности однотипных объектов;

б) информацию о совокупности всех объектов, относящихся к некоторой предметной области;

в) информацию о конкретном объекте.

4. Строка таблицы данных содержит:

а) информацию о совокупности однотипных объектов;

б) информацию о совокупности всех объектов, относящихся к некоторой предметной области;

в) информацию о конкретном объекте.

5. Столбец таблицы данных содержит:

а) информацию о совокупности однотипных объектов;

б) информацию о совокупности всех объектов, относящихся к некоторой предметной области;

в) информацию о конкретном экземпляре объекта;

г) совокупность значений одного из информационных атрибутов для всех однотипных объектов.

6. Структура таблицы данных определяется:

   а) размерностью таблицы;

   б) списком наименований столбцов таблицы;

   в) списком наименований столбцов и номеров строк таблицы.

7. Полем данных называется:

   а) значение атрибута для конкретного экземпляра объекта;

   б) элемент структуры таблицы;

   в) список значений конкретного атрибута для всех однотипных объектов.

8. Поле данных может содержать информацию следующего типа:

   а) число;

   б) текст;

   в) примечания;

   г) формулы;

   д) звуковые объекты;

   е) ссылки на элементы других таблиц данных.

9. Таблица данных называется главной, если:

   а) все данные базы представлены этой таблицей;

   б) все поля таблицы являются ссылками на другие таблицы данных;

   в) таблица данных содержит поля, являющиеся ссылками на другие таблицы данных;

   г) в составе базы данных есть таблица, содержащая поля-ссылки на данную таблицу.

10. Таблица называется подчиненной, если:

    а) все данные базы представлены этой таблицей;

    б) все поля таблицы являются ссылками на другие таблицы данных;

    в) таблица данных содержит поля, являющиеся ссылками на другие таблицы данных;

    г) в составе базы данных есть таблица, содержащая поля-ссылки на данную таблицу.

11. Таблицы называются связанными, если:

    а) одна из них является главной, а другая — подчиненной;

    б) если одна из них является подчиненной другой (непосредственно или через цепочку таблиц данных);

    в) их структуры совпадают частично.

12. Ключом таблицы данных называется:

    а) строка таблицы, содержащая уникальную информацию;

    б) совокупность полей таблицы, однозначно определяющая каждую ее строку.

13. Могут ли разные строки таблицы иметь одинаковое значение ключа?

    а) да;

    б) нет.

14. Запросом к базе данных называется:

    а) таблица, отсортированная по возрастанию или убыванию значений ключа;

    б) таблица, полученная из исходной путем выбора строк, удовлетворяющих заданным условиям на значения полей;

    в) таблица, полученная из совокупности связанных таблиц посредством выбора строк, удовлетворяющих заданным условиям.

15. На каком языке из перечисленных ниже записываются запросы для современных баз данных:

    а) Clipper;

    б) Pascal;

    в) SQL;

    г) Visual Basic.

16. В режиме СУБД могут использоваться следующие программы:

    а) FoxPro;

    б) Paradox;

    в) Windows;

    г) CorelDraw;

    д) Supercalc;

    е) MS Works;

    ж) MS Access;

    з) Лексикон.

17. Какие из СУБД предоставляют возможности визуального конструирования баз данных?

    а) FoxPro;

    б) Paradox;

    в) Windows;

г) CorelDRAW;

д) Supercalc;

е) MS Works;

ж) MS Access;

з) Лексикон.

18. В каких единицах измеряется длина поля в БД?

а) сантиметр;

б) килобайт;

в) символ;

г) поле не имеет единиц измерения.

19. Что такое форма?

а) окно на экране компьютера с местом для ввода данных и обозначением полей;

б) обозначение полей базы данных;

в) таблица, в которой находятся данные.

20. Что такое запись?

а) информация, занесенная в некоторые из полей, хранящаяся в БД под определенным номером;

б) порядковый номер информации;

в) название поля;

г) единица размерности поля.

21. В базе данных записи отсортированы в следующем порядке: Иванов, Журавлев, Антонов. Каков порядок сортировки?

а) убывающий;

в) возрастающий;

г) ни один из перечисленных.

5.4. Обработка графической информации

С бурным развитием компьютерной техники все большее значение стало придаваться не только содержанию получаемой и передаваемой информации, но и ее графическому представлению.

И если вы еще не художник, то, используя специальные программы для создания и обработки компьютерных изображений, библиотеки готовых изображений, вы легко сможете изобразить черный квадрат. Вот только признают ли вас как Казимира Малевича?

Навыки рисования на компьютере все же нужны — и детям и взрослым, и для отдыха и для дела.

Попробуем исполнить несколько практикумов — от простого к сложному.

Практикум 59. Навыки рисования при помощи графических примитивов. Компьютерная симметрия

Попробуйте изобразить следующие фигуры, используя возможности копирования нарисованных фрагментов и их повороты на 90, 180 и 270 градусов. (рис. 5.66—5.75).

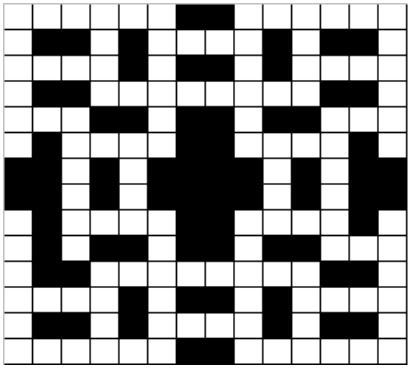


Рис. 5.66. Кроссворд-1

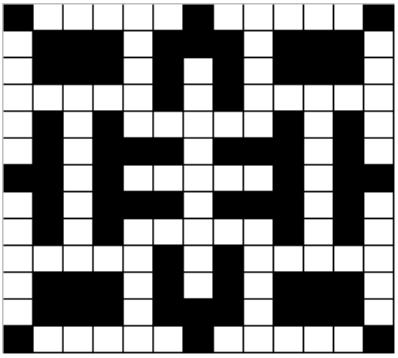


Рис. 5.67. Кроссворд-2

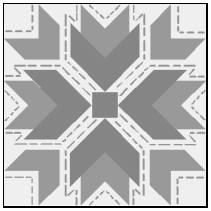


Рис. 5.68. Снежинка

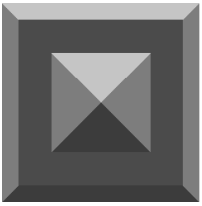


Рис. 5.69. Выпуклая штучка

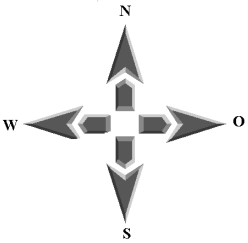


Рис. 5.70. Стороны света

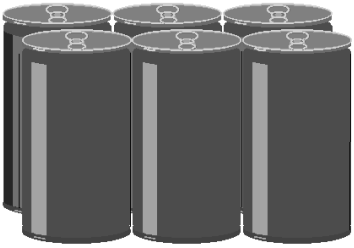


Рис. 5.71. Упаковка лимонада

Рис. 5.72. Почтовая марка

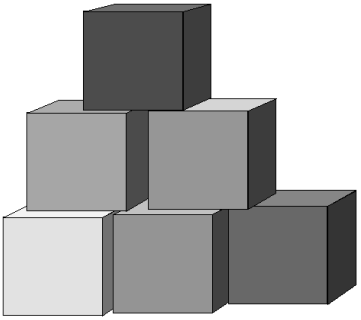


Рис. 5.73. Кубики

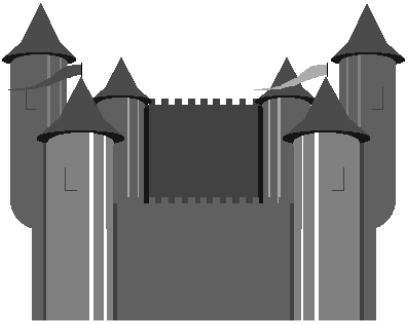


Рис. 5.74. Замок

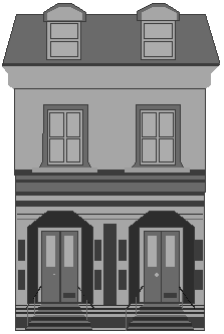


Рис. 5.75. Жилой дом

Практикум 60. Более сложные объекты

Усложняем рисунки (рис. 5.76—5.85).

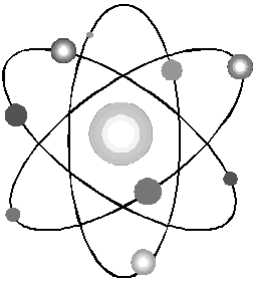


Рис. 5.76. Атом

Рис. 5.77. No smoking!

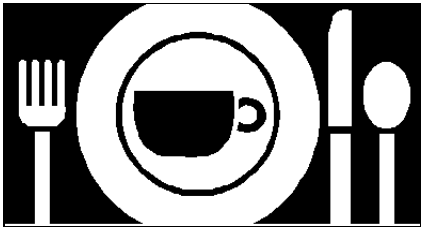


Рис. 5.78. Кушать подано!

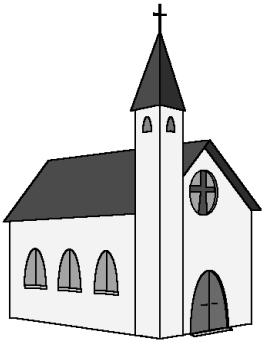


Рис. 5.79. Церковь

Рис. 5.80. Знак пешеходного перехода

Рис. 5.81. Свинья-копилка

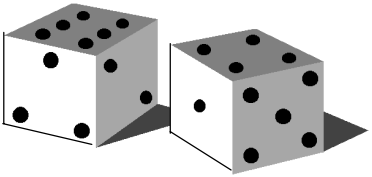


Рис. 5.82. Игральные кости

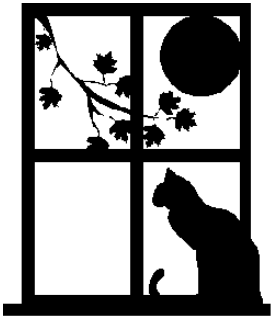


Рис. 5.83. Черный кот

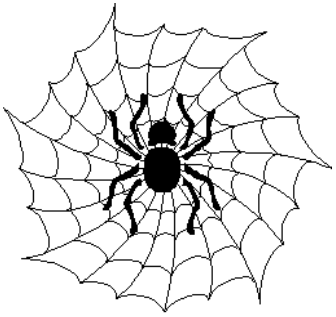


Рис. 5.84. Паучок

Рис. 5.85. Футбол

Практикум 61. Конкурс на лучшую антирекламу

Мы каждый год традиционно проводим такой конкурс среди учащихся старших классов. Конкурс мини-плаката "Анти — антиникотин, или антиалкоголь, или антинаркотики, или безопасный секс". Сначала идея воспринимается со смешками, но потом, кроме чистого остроумия, обязательно

обнаруживаются свежие, нетривиальные идеи. По итогам проводится школьная выставка, лучшие работы награждаются.

Вот некоторые примеры того, что получалось иногда (рис. 5.86—5.88).

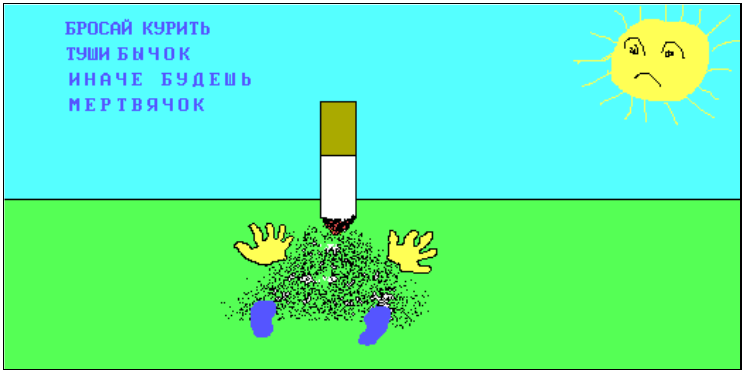


Рис. 5.86

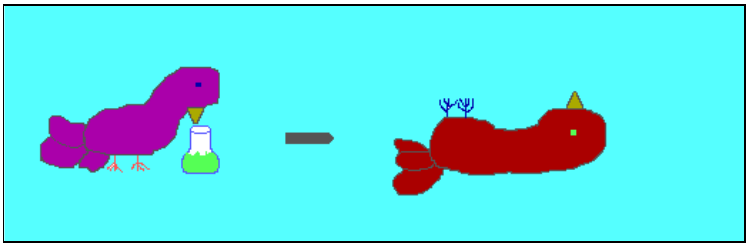


Рис. 5.87

Рис. 5.88

Практикум 62. Творческие задания

Графические редакторы таят в себе огромные резервы для творческой реализации ученических идей. Например:

- конкурс на лучшую обложку молодежного журнала;
- открытки к праздникам — к Новому году, ко Дню всех влюбленных, к 8 Марта и т. д.;
- иллюстрации к рефератам и компьютерным моделям по точным и естественным наукам, а также к тестовым компьютерным заданиям;
- разработка собственного фамильного герба;
- конкурс на герб или флаг школы;
- школьные комиксы

и т. д., и т. п.

Практикум 63. Обработка фотографий

При наличии в школе сканера дело упрощается, но и при отсутствии решается легко — возможностей отсканировать фотографию сейчас много. А дальше:

- при помощи CD "Cosmopolitan" — подбор причесок себе, любимому другу (подруге), любимым и не очень учителям и директору;
- всевозможные коллажи по поводу и без;
- компьютерные фотоальбомы;
- иллюстрирование школьных журналов и газет;
- восстановление домашнего фотоархива;
- изготовление беджей.

Ну а под конец раздела, как всегда, проверим ваши теоретические познания.

Тест 3. Графические редакторы

1. Графическим редактором называется программа, предназначенная для:
   а) создания графического образа текста;
   б) редактирования вида и начертания шрифта;
   в) работы с графическим изображением;
   г) построения диаграмм.
2. С помощью графического редактора Paint можно:
   а) создавать и редактировать графические изображения;
   б) редактировать вид и начертание шрифта;
   в) настраивать анимацию графических объектов;
   г) строить графики.

3. Минимальным объектом, используемым в растровом графическом редакторе, является:

   а) точка экрана (пиксел);

   б) объект (прямоугольник, круг и т. д.);

   в) палитра цветов;

   г) знакоместо (символ).

4. Минимальным объектом, используемым в векторном графическом редакторе, является:

   а) точка экрана (пиксел);

   б) объект (прямоугольник, круг и т. д.);

   в) палитра цветов;

   г) знакоместо (символ).

5. Деформация изображения при изменении размера рисунка — один из недостатков:

   а) растровой графики;

   б) векторной графики.

6. Примитивами в графическом редакторе называются:

   а) линия, круг, прямоугольник;

   б) карандаш, кисть, ластик;

   в) выделение, копирование, вставка;

   г) набор цветов.

7. К основным операциям, возможным в графическом редакторе, относятся:

   а) линия, круг, прямоугольник;

   б) карандаш, кисть, ластик;

   в) выделение, копирование, вставка;

   г) набор цветов.

8. Палитрой в графическом редакторе является:

   а) линия, круг, прямоугольник;

   б) карандаш, кисть, ластик;

   в) выделение, копирование, вставка;

   г) набор цветов.

9. Инструментами в графическом редакторе являются:

   а) точка экрана (пиксел);

   б) объект (прямоугольник, круг и т. д.);

в) палитра цветов;

г) знакоместо (символ).

10. Какой из указанных графических редакторов является векторным?

а) CorelDRAW;

б) Adobe Photoshop;

в) Paint;

г) Adobe Illustrator.

11. В цветовой модели RGB установлены следующие параметры: 0, 255, 0. Какой цвет будет соответствовать этим параметрам?

а) черный;

б) красный;

в) зеленый;

г) синий.

12. Большой размер файла — один из недостатков:

а) растровой графики;

б) векторной графики.

13. В модели RGB в качестве компонентов применяются основные цвета:

а) голубой, пурпурный, желтый;

б) красный, голубой, желтый;

в) красный, зеленый, синий;

г) пурпурный, желтый, черный.

14. В цветовой модели RGB установлены следующие параметры: 255, 0, 0. Какой цвет будет соответствовать этим параметрам?

а) черный;

б) красный;

в) зеленый;

г) синий.

15. В процессе сжатия растровых графических файлов по алгоритму JPEG его информационный объем обычно уменьшается в:

а) 100 раз;

б) 2—3 раза;

в) 10—15 раз;

г) не изменяется.

16. Разрешение изображения измеряется в:

    а) пикселах;

    б) точках на дюйм (dpi);

    в) мм, см, дюймах;

    г) количестве цветовых оттенков на дюйм (jpeg).

17. Разрядность представления цвета это:

    а) количество цветов, отображаемых на экране монитора;

    б) цифры, 8, 16, 24 и 32 (или 256, True Color);

    в) число битов, необходимых для кодирования одного пиксела в зависимости от количества цветов;

    г) совпадает с разрядностью системной шины.

18. При увеличении разрешения (количества пикселов на дюйм) и размера рисунка размер файла этого рисунка:

    а) уменьшается;

    б) возрастает;

    в) остается неизменным.

19. Минимальной единицей измерения на экране графического редактора является:

    а) мм;

    б) см;

    в) пиксел;

    г) дюйм.

20. Какая заливка называется градиентной?

    а) сплошная (одним цветом);

    б) с переходом (от одного цвета к другому);

    в) заливка с использованием внешней текстуры;

    г) заливка узором.

21. Даны две оцифрованные картинки. Первая — получена в режиме "Gray", вторая в режиме "Color". Какая из них будет занимать больше места на жестком диске, если при сканировании использовался один и тот же образец и одно и то же разрешение?

    а) первая;

    б) вторая;

    в) обе одинаковые.

И еще один маленький тест.

Тест 4. Сканирование

1. Какой стороной нужно помещать оригинал в сканер?

   а) изображением вверх;

   б) любой стороной;

   в) изображением вниз.

2. С каким разрешением вы будете сканировать большую фотографию, которую в дальнейшем собираетесь показывать на экране компьютера и посылать по E-mail?

   а) 600—1200 dpi;

   б) 75—100 dpi;

   в) 300—600 dpi;

   г) некритично.

3. Для сканирования только фрагмента большой фотографии в редакторе растровой графики следует:

   а) использовать повышенное разрешение;

   б) выбрать режим True Color;

   в) изменить область сканирования;

   г) уменьшить разрешение.

4. С каким разрешением вы будете сканировать небольшую фотографию, которую в дальнейшем собираетесь распечатать на листе формата А4?

   а) 600 и менее dpi;

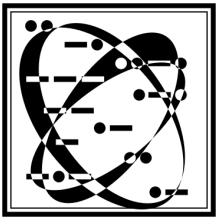
   б) максимально возможным оптическим разрешением;

   в) 150—300 dpi;

   г) с разрешением принтера;

   д) выбор разрешения зависит от качества фотографии.

Глава 6



Мультимедийные технологии

Эта глава посвящена, в основном, вспомогательным, но очень полезным компьютерным приложениям. Приводимые практикумы представлены, главным образом, на уровне идей, которые творческая личность (каковой обязательно должен быть любой учитель информатики) может воспринять или не воспринять, сделать, как предлагает автор или по-своему. Все эти идеи реализуемы практически.

6.1. Программы распознавания текста

Это называется — техника на службе человека. Зачем теперь переписывать реферат из книжек вручную? Особо продвинутые, конечно, скажут, что и из книжек-то незачем — все есть в Интернете, но будут правы лишь отчасти, поскольку на каждого хитрого ученика или студента найдется не менее хитрый преподаватель, который тоже об Интернете знает не понаслышке. Поэтому лучше рефераты все же делать самому, но при помощи современной компьютерной техники — сканера и программ распознавания печатного текста, из которых самой известной является ABBYY FineReader. Хотя есть много и других неплохих программ, например CuneiForm, Kodak Imaging Scanstation, ScanSoft OmniPage, Pecognita Plus. Некоторые программы обладают уникальной возможностью распознавания и рукописного текста (правда тех, кто пишет, как курица лапой, это пока не касается).

Практикум 1. Распознавание текста

Имеется набор учебников по информатике. Необходимо подготовить на уроке, в компьютерном классе, краткое сообщение (две-три страницы формата А4) на следующие, например, темы:

- ❒ история развития вычислительной техники;
- ❒ принтеры, их виды и свойства;
- ❒ мониторы, сравнительные характеристики;

- моделирование, классификация моделей по различным признакам;
- внешняя и внутренняя память компьютера.

Выберите необходимые фрагменты, отсканируйте, распознайте текст, экспортируйте в Word, скомпонуйте, оформите, проиллюстрируйте отсканированными же картинками из учебников. Оформите титульный лист, распечатайте документ, сдайте работу.

6.2. Программы-переводчики

Опять же — все для человека. Зачем мучиться, не понимая, что написано в инструкции или письме, в газете или на сайте. Облегчаем себе жизнь!

Из известных программ-переводчиков приведем в пример несколько — Promt, Socrat, Language Master, Magic Gooddy, WebTransSite.

Практикум 2. Перевод руководства по материнской плате (монитору) или, на худой конец, любого печатного текста

Имеющийся иностранный текст сканируем, распознаем, переводим при помощи программы-переводчика. Получаем *подстрочник*, т. е. необработанный русский текст. Экспортируем его в Word, придаем ему правильное русское звучание, оформляем, готовим титульный лист, иллюстрируем отсканированными картинками из текста, распечатываем и сдаем!

Тест 1. Распознавание с переводом

1. Какие типы блоков могут встречаться в программе FineReader?

   а) картинка и текст;

   б) текст, таблица;

   в) распознаваемые и нераспознаваемые;

   г) текст, таблица, рисунок.

2. Будет ли переведено слово, отсутствующее в словаре программы-переводчика?

   а) да;

   б) нет;

   в) наличие словаря необязательно.

3. Как изменить направление перевода текста?

   а) вводить текст на другом языке;

б) направление перевода меняется автоматически;

в) выбрать нужную команду из меню или через инструмент.

4. Как выполнить перевод текста, видимого на экране браузера (странички из Интернета)?

а) напечатать аналогичный текст в программе-переводчике;

б) скопировать в буфер обмена;

в) скопировать в буфер обмена и вставить в переводчик, либо правой кнопкой мыши выбрать нужное направление перевода;

г) перевод текста со страницы браузера невозможен;

д) сохранить страницу на диске, затем перевести в переводчике.

5. Выберите наиболее производительный способ перевода текста, находящегося в каком-либо файле:

а) открыть файл в текстовом редакторе и напечатать его заново, затем перевести;

б) открыть файл в переводчике и перевести;

в) открыть в текстовом редакторе, скопировать в буфер и вставить в переводчик, затем перевести.

6. Принцип действия программы-переводчика основан на:

а) использовании памяти компьютера;

б) использовании машинной системы перевода;

в) использовании словаря.

7. Требуется перевести на русский язык английский текст, отпечатанный на листе бумаги. Ваши действия:

а) перепечатать текст с листа (ввести в программу-переводчик) и перевести;

б) набрать в текстовом редакторе, скопировать в переводчик и перевести;

в) преобразовать в программе распознавания, скопировать в переводчик и перевести.

8. Возможен ли перевод текста из другого приложения Windows?

а) нет, невозможен;

б) да, с помощью буфера обмена;

в) возможен, но не из всех приложений.

9. Для выделения блока вручную в FineReader требуется:

а) нарисовать рамку левой кнопкой мыши вокруг нужного текста;

б) используя левую кнопку мыши, указать границы блока, уточнив их в "крупном плане";

в) блоки выделяются в автоматическом режиме.

6.3. Работа с цифровой камерой

В наше время бурно развивающихся технологий грех было бы ими не пользоваться. Если школа так богата, что купила себе цифровую камеру — хорошо, а, если нет, то неужели не найдется благодарный, да нет — благородный! — родитель, который не даст детям позаниматься со своей?

Практикум 3. Виртуальная галерея (живые картины или скульптуры)

Каждый учащийся вытаскивает заранее подготовленные карточки с репродукцией известной картины или скульптуры и порядковый номер, под которым он должен реализовать свой замысел. Когда подходит его черед, учащийся выбирает из группы (класса) тех, кто нужен ему для воплощения замысла, осуществляет постановочный кадр (можно использовать подручные средства для быстрого макияжа и костюмирования) и запечатлевает это на smart-card цифровой камеры. По окончании занятия кадры переносятся на винчестер, обрабатываются графическими редакторами (яркость, контрастность и т. д.), затем вставляются в виртуальные рамы и — "Ах, вернисаж, ах, вернисаж! Какой портрет, какой пейзаж!.." При возможности, из лучших работ организуется реальная фотовыставка, а на сайте школы — постоянно действующая экспозиция.

6.4. Компьютерные презентации

Входящая в состав Microsoft Office программа PowerPoint позволяет сделать настоящее мультимедийное слайд-шоу, которое, будучи грамотно выполнено, может быть размещено в Интернете в качестве полноценного сайта (особенно хорошо это получается в PowerPoint XP).

Кроме того, умело сделанные презентации можно использовать в качестве учебных пособий в преподавании различных школьных дисциплин.

Практикум 4. Компьютерные презентации

Вариант 1. Моя биография

Учащиеся готовят десятислайдовые шоу, в которых представляют свои короткие, но уже насыщенные событиями биографии. Каждый слайд должен содержать одну-две фотографии, короткие пояснения и сопровождаться любимыми мелодиями героя презентации (рис. 6.1).

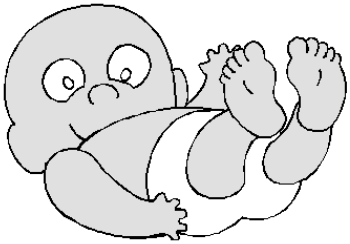


Рис. 6.1. Моя биография

Вариант 2. Любимая группа

Учащиеся готовят десятислайдовые шоу, в которых представляют свои любимые музыкальные группы. Каждый слайд должен содержать одну-две фотографии, короткие пояснения и сопровождаться любимыми мелодиями в исполнении любимой группы автора презентации (рис. 6.2).

Рис. 6.2. Любимая группа

Вариант 3. История развития вычислительной техники — от 10 пальцев до наших дней

Учащиеся готовят десятислайдовые шоу, в которых представляют один из этапов развития вычислительной техники (рис. 6.3). Каждый слайд должен содержать одну-две фотографии или картинки, короткие пояснения и сопровождаться техногенными мелодиями, подобранными автором.

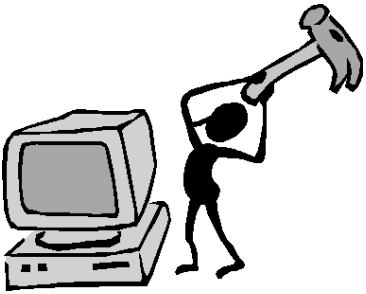


Рис. 6.3. История развития ВТ

Вариант 4. Славные страницы военной истории России

Учащиеся готовят десятислайдовые шоу, в которых представляют одно из прославивших Россию сражений (рис. 6.4). Каждый слайд должен содержать одну-две фотографии или картинки, планы баталий, короткие пояснения и сопровождаться патриотическими мелодиями, подобранными автором.

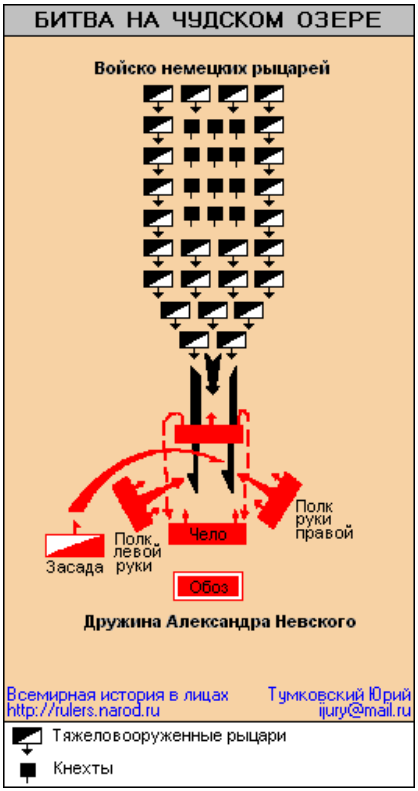


Рис. 6.4. Ледовое побоище

Вариант 5. Спорт

Учащиеся готовят десятислайдовые шоу, в которых представляют один из своих любимых видов спорта или команду (рис. 6.5). Каждый слайд должен содержать одну-две фотографии или картинки, короткие пояснения и сопровождаться песнями или мелодиями на спортивные темы, подобранными автором.

Рис. 6.5. Футбол

Вариант 6. Рекламная презентация родной и любимой школы

Учащиеся готовят десятислайдовые шоу, в которых представляют все лучшее, чем, на их взгляд, можно гордиться в родимой школе (рис. 6.6). Каждый слайд должен содержать одну-две фотографии или картинки, короткие пояснения и сопровождаться песнями или мелодиями, подобранными автором.

Рис. 6.6. Любимая школа

Вариант 7. Меньшие братья по разуму

Учащиеся готовят десятислайдовые шоу, в которых представляют своего (или не своего) любимца из числа домашних животных (рис. 6.7). Каждый слайд должен содержать одну-две фотографии или картинки, короткие пояснения и сопровождаться звуками, соответствующими животному, выбранному автором.

Рис. 6.7. Брат по разуму

Вариант 8. Мой город – лучший в мире!

Учащиеся готовят десятислайдовые шоу, в которых представляют лучшие стороны своего родного города (рис. 6.8). Каждый слайд должен содержать одну-две фотографии или картинки, короткие пояснения и сопровождаться песнями или мелодиями, посвященными городу, подобранными автором.

Рис. 6.8. Мой город

Вариант 9. Что будет модно на выпускном вечере

Учащиеся готовят десятислайдовые шоу, в которых представляют разработанные ими наряды для выпускного вечера (рис. 6.9). Каждый слайд должен содержать одну-две фотографии или картинки, короткие пояснения и сопровождаться подходящими мелодиями, подобранными автором.

Рис. 6.9. Ветреная мода

Вариант 10. Один день школы: стоп-кадр

Учащиеся готовят десятислайдовые шоу, в которых представляют один из дней школы (рис. 6.10). Каждый слайд должен содержать одну-две фотографии или картинки, короткие пояснения и сопровождаться школьными звуками, любовно подобранными автором.

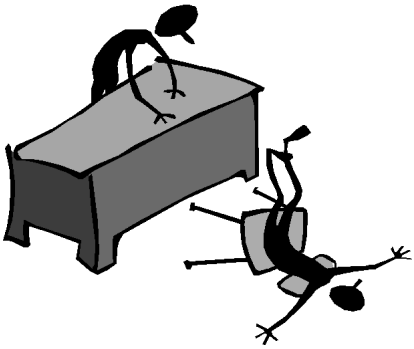


Рис. 6.10. Стоп-кадр

Тест 2. Знаете ли вы PowerPoint?

1. Презентация это:

   а) набор слайдов, показываемых в определенном порядке;

   б) иллюстрированный рассказ о товаре, услуге, объекте;

   в) способ передачи информации.

2. Для применения к объекту MS PowerPoint эффектов анимации необходимо:

   а) перейти в режим структуры;

   б) выбрать команду меню **Показ слайдов | Встроенная анимация**;

   в) выделить объект и перетащить его на новое место.

3. Для изменения шаблона дизайна в MS PowerPoint следует:

   а) перейти в режим сортировщика слайдов;

   б) выделить лист (слайд);

   в) выбрать команду меню **Формат | Применить шаблон оформления**.

4. Для изменения размера объекта в редакторах векторной графики и MS PowerPoint необходимо:

   а) использовать прием, аналогичный изменению свойств окна;

   б) сначала выделить объект, затем изменять размеры с помощью мыши или панели свойств;

   в) изменение размеров объекта (фигуры) возможно только через меню.

5. Для изменения порядка расположения объектов на листе (слайде) следует:

   а) перетащить объект относительно другого;

   б) выделить объект;

   в) использовать команду **Порядок | Впереди** или **Позади**;

   г) изменить прозрачность "верхнего" объекта.

6. Как вставить в слайд картинку (рисунок), ранее сохраненный на диске?

   а) скопировать через буфер обмена;

   б) использовать команду **Вставка | Рисунок** или **Картинка**;

   в) выбрать команду меню **Вставка | Рисунок из файла**.

7. Как применить эффект тени к объекту **Текст** в MS PowerPoint?

   а) создать копию объекта и поместить ее позади основного;

   б) выбрать команду меню **Формат | Шрифт**, установить метку в пункте **Тень** или использовать инструмент;

   в) использовать только инструмент **Тень** панели **Рисование**.

6.5. Создание маленького сайта

Конечно, в любой школе найдется несколько человек, одержимых не только компьютерными играми, но и интернет-технологиями. Главное — направить их кипучую энергию в правильное русло. А еще лучше использовать их в качестве консультантов при создании индивидуальных сайтов для остальных учащихся.

Для создания собственных сайтов можно рекомендовать любые инструменты, начиная от встроенных возможностей Word и непосредственного использования языка HTML или программ FrontPage Express, DreamViewer и заканчивая модными ныне Flash-технологиями (привет Масяне! — рис. 6.11). Кроме того, многие бесплатные серверы типа **Boom.ru**, **Narod.ru**, **Hotbox.ru** и др. представляют не только место под персональные странички, но и простые конструкторы для их создания.

Рис. 6.11. Масяня

Практикум 5. "Мой маленький сайтик — ты очень мне дорог..."

Каждому человеку хочется самовыразиться. Интернет предоставляет для этого отличные возможности. В глобальной Сети висят миллионы личных страничек самовыражающихся людей, но — увы! — в большинстве своем это однообразные рассказы о себе на казенный манер: "родился, учился, женился и от этого умер" с не менее однообразными фотографиями: "вот я на горшке, а вот я на Эйфелевой башне" — тоска смертная.

Так что вам и карты в руки — самовыразитесь так, чтобы Сеть ахнула. Проснитесь знаменитым! Все в ваших руках!

6.6. Поиск в Интернете

Существует несколько крупных поисковых систем, у каждой из которых есть как свои ярые приверженцы, так и явные или тайные недоброжелатели.

Вот несколько — **Rambler.ru**, **Yahoo.com**, **Yandex.ru**, **Altervista.ru**, **Google.ru**, **Aport.ru** и множество других. Искусство поиска в Сети весьма сложно. Недаром устраиваются чемпионаты по поиску.

Попробуйте и вы. Непременное условие — необходимо не только найти правильный ответ, но и обязательно подтвердить его ссылкой на интернет-ресурс, на котором этот ответ был найден.

Практикум 6. Интеллектуальное Интернет-Очко

Итак, мы вам 21 вопрос (табл. 6.1), вы нам 21 ответ со ссылками, да еще за минимальное время! (В скобках указано количество баллов за правильный ответ, подкрепленный ссылкой.)

Таблица 6.1. *Интеллектуальное Интернет-Очко*

№	Вопрос	Ответ	Ссылка на интернет-ресурсы
1.	Год создания первой ЭВМ ENIAC и расшифровка аббревиатуры ее названия (10)		
2.	От какой даты ведется летоисчисление в Исламе? (10)		
3.	Название малой планеты № 433 и ее диаметр (20)		
4.	Как называется, кем и когда открыт и в честь чего назван химический элемент № 71 (20)		
5.	Время отправления поезда Калининград—Москва (20)		
6.	Какая станция третья при поездке на электричке по маршруту Москва—Петушки? (10)		
7.	Кто был вторым мужем Джейн Фонды (20)		
8.	Стоимость входного билета в Лувр (30)		
9.	Основные гостиницы города Магадан (30)		
10.	Кто построил Парфенон? (30)		
11.	Размеры фрески "Тайная вечеря" Леонардо да Винчи (50)		

Таблица 6.1 *(окончание)*

№	Вопрос	Ответ	Ссылка на интернет-ресурсы
12.	Высота Кёльнского собора (20)		
13.	Как называлась первая газета, в которой работал Чарльз Диккенс? (30)		
14.	Кто открыл реку Замбези в Африке? (10)		
15.	Как называется самое высокогорное озеро на Земле? (20)		
16.	Сколько провинций в Канаде? (10)		
17.	Когда и за сколько США купили Аляску у России? (30)		
18.	Как звали и в виде кого изображался древнеегипетский бог, покровитель умерших? (20)		
19.	Имена трех сыновей Ноя, от которых после всемирного потопа "населилась вся земля" (10)		
20.	Как называется денежная единица и разменная монета Объединенных Арабских Эмиратов? (20)		
21.	Какая температура днем ожидается в Сочи завтра? (10)		

Практикум 7. 300-летию Санкт-Петербурга посвящается

Условия те же — мы вам 21 вопрос про любимый город (табл. 6.2), вы нам — 21 ответ. Стоимость вопросов можете придумать сами.

Таблица 6.2. *К 300-летию Санкт-Петербурга*

№	Вопрос	Ответ	Ссылка на интернет-ресурсы
1.	На какой широте расположен Санкт-Петербург?		
2.	Каков номер малой планеты "Ленинград"?		
3.	В Ленинградском зоопарке впервые в России и СССР родился жираф. В каком году это произошло?		

Таблица 6.2 *(продолжение)*

№	Вопрос	Ответ	Ссылка на интернет-ресурсы
4.	В Царском Селе в парке у пирамидальной беседки захоронены три любимые собаки императрицы Екатерины II. Перечислите их клички.		
5.	Какого числа был проведен первый в СПб. футбольный матч?		
6.	Какова высота самого высокого сооружения СПб.?		
7.	Какой мост в Санкт-Петербурге первым построен из чугуна?		
8.	Больше всего львов в СПб. у бывшей дачи Кушелева-Безбородко. Сколько их там?		
9.	Самая длинная улица СПб.?		
10.	Кто забил голы в решающем матче "Зенит"—"Металлист" 21 ноября 1984 года, после которого "Зенит" стал чемпионом СССР?		
11.	По чьему проекту было построено старейшее в городе лечебное учреждение общего пользования?		
12.	Сколько свай было забито в грунт для постройки Исаакиевского собора?		
13.	Какая школа (по номеру) в плане представляет собой стилизованные серп и молот?		
14.	Самый большой в мировой практике вес разводного крыла — 3225 тонн имеет один из мостов СПб. Какой?		
15.	Сколько фонтанов входит в Большой Каскад Нижнего парка Петергофа?		
16.	В саду дома № 12 по улице Академика Павлова сооружен в 1935 году памятник собаке как дань бескорыстному служению собак науке. Кто автор проекта памятника?		
17.	Из всех городов мира наибольшим числом конных памятников историческим лицам обладает Петербург. Сколько их и каким императорам?		
18.	Каково современное название Кунсткамеры?		
19.	Первый русский журнал для детей вышел в 1785 году. Как он назывался?		

Таблица 6.2 *(окончание)*

№	Вопрос	Ответ	Ссылка на интернет-ресурсы
20.	Старейший из вокзалов Петербурга — Московский. Кто был первым его начальником?		
21.	12 мая 1911 года состоялся первый выпуск Гатчинской школы авиации. Сколько летчиков было выпущено и их фамилии?		

Практикум 8. Пример задания на Кубке Yandex по поиску в Интернете

Правила проведения Кубка были очень просты — участники соревновались в нахождении в Интернете ответов на заданные вопросы. Никаких ограничений на использование инструментов поиска не было.

Необходимо было найти то, что записано в скобках курсивом.

Задание "Астрономия"

Близ {*название созвездия*}, над горизонтом расположен Эридан {*латинское название*}, цепочка звезд которого напоминает реку, текущую от {*часть тела*} {*имя*} (ее и называли Рекой — Потамос). По мифу, {*имя*}, сын Солнца, назывался сперва Эриданом, он же дал свое имя реке {*название реки*} (Падус), в которую упал. Это длинное созвездие оканчивается звездой {*название звезды*}, (по-арабски: ахир ан-нахр — {*перевод названия*}) — {*которой?*} по яркости звездой неба (звездная величина {*число*}, расстояние от Солнца — 30 {*чего?*}).

{*Специалисты*} высоко оценивают эту звезду. {*Фамилия*} называет ее "Ангел с мечом", а {*имя*} Глоба — звездой высшего покровительства ("{*образное выражение*}" защиты). Эбертин и {*фамилия*} находят у нее, помимо {*какого?*} влияния, примесь энергий Марса и Урана. Интуитивно она привлекает и писателей: именно так — "{*греческая буква*}* Эридана" — был назван известный рассказ {*имя*} {*фамилия*}, давший название одноименному сборнику фантастики издательства "{*Название*}" ({*год*} г.).

В древности она не была известна (ибо не видна в Европе и на Ближнем Востоке), поэтому последней звездой считали другую, менее яркую звезду того же созвездия — {*название звезды*} ({*греческая буква*}* Эридана). Именно смешение этих звезд порождает ошибки в псевдоисторических трудах {*имя*} Фоменко.

Многие другие звезды этого созвездия также интересны, например, {*греческая буква*}* Эридана Курса, Омикрон-1 Эридана {*название звезды*}. Эта

Эридана {*название звезды*} и {*греческая буква*}* (Тимим). Кстати, обозначать звезды буквами греческого алфавита предложил в своем атласе "{*Название*}" {*национальность*} астроном {*имя*} Байер в {*год*} г.; но букв всего {*число*}, так что применяются дополнительные индексы — в том же Эридане целых 9 звезд {*греческая буква*}*!

Но наиболее интересна практически звезда {*греческая буква*}* Эридана — это самая близкая из обнаруженных до сих пор звезд с планетами, она находится на расстоянии всего 10,5 {*чего?*}. Планета была обнаружена группой {*имя*} {*фамилия*} в обсерватории {*какого?*} университета.

Замечание

Названия греческих букв (в задании помечены знаком *) записывать русскими буквами.

6.7. Электронная почта

Очень удобная штука — электронная почта. Был такой случай. Поехал автор в отпуск — недалеко, на родину — в славный город Калининград (ex-Konigsberg). Решил домой письмишко написать и открыточку отправить с видами янтарных берегов. Написал в первый же день, отправил. Вернулся домой через две недели, а тут встреча неадекватная — где ты был, почему не писал?! Письмо это пришло еще через пару дней.

На следующий год пошел автор в Калининграде на почтамт, да и отправлял с него, с временного электронного почтового ящика, письма домой каждый день, а там в тот же день все получали, и встретили его по возвращении добрые все и довольные.

Отсюда мораль, электронная почта — не роскошь, а средство поддержания хорошего здоровья (рис. 6.12)!

А чтобы это здоровье вас не покидало — подготовимся заранее.

Рис. 6.12. Почта

Практикум 9. Регистрация почтового ящика. Настройка почтовой программы

Подключимся к Интернету, войдем на бесплатный сервер, скажем **Mail.ru**, и выберем опцию **Регистрация**. Далее все весьма просто, знай, следуй инструкциям. После создания ящиков (лучше было бы, если бы имена пользователей — логины — были однотипными, тогда учителю будет легче их идентифицировать), можно попереписываться, затем, воспользовавшись инструкцией с сервера, где вы зарегистрировались, настройте почтовую программу Outlook Express или Bat!

Пошлите отчет о проделанных действиях в ящик учителю.

В дальнейшем эти ящики можно использовать для обмена, например, домашними заданиями.

Тест 3. Компьютерные телекоммуникации-1

1. Какой из способов подключения к Интернету обеспечивает наибольшие возможности для доступа к информационным ресурсам?

   а) постоянное соединение по оптоволоконному каналу;

   б) удаленный доступ по коммутируемому телефонному каналу;

   в) постоянное соединение по выделенному телефонному каналу;

   г) терминальное соединение по коммутируемому телефонному каналу.

2. Модем — это:

   а) почтовая программа;

   б) сетевой протокол;

   в) сервер Интернета;

   г) техническое устройство.

3. Модем, передающий информацию со скоростью 56 600 бит/с, может передать песню в формате MP3 (3,6 Мбайт) в течение приблизительно:

   а) 9 минут;

   б) получаса;

   в) 40 секунд;

   г) 1 дня.

4. Электронная почта (E-mail) позволяет передавать:

   а) только сообщения;

   б) только файлы;

   в) сообщения и приложенные файлы;

   г) видеотрансляцию.

5. Какой протокол является базовым в Интернете?
   а) HTML;
   б) TCP;
   в) HTTP;
   г) TCP/IP.
6. Компьютер, подключенный к Интернету, обязательно имеет:
   а) IP-адрес;
   б) Web-сервер;
   в) домашнюю Web-страницу;
   г) доменное имя.
7. Гиперссылки на Web-странице могут обеспечить переход:
   а) только в пределах данной Web-страницы;
   б) только на Web-страницы данного сервера;
   в) на любую Web-страницу данного региона;
   г) на любую Web-страницу любого сервера Интернета.
8. Задан адрес электронной почты в сети Интернет: **klero@rambler.spb.ru**. Каково имя владельца электронного адреса?
   а) rambler.spb.ru;
   б) klero;
   в) rambler;
   г) ru;
9. Браузеры (например, Microsoft Internet Explorer) являются:
   а) серверами Интернета;
   б) антивирусными программами;
   в) трансляторами языка программирования;
   г) средством просмотра Web-страниц.
10. Web-страницы имеют формат (расширение):
   а) txt;
   б) htm;
   в) doc;
   г) exe.

Тест 4. Компьютерные телекоммуникации-2

1. Какие виды сетей вы знаете?

   а) внутренние и внешние;

   б) локальные и глобальные;

   в) встроенные и подключаемые;

   г) коммутируемые и постоянные.

2. Для открытия файла, находящегося на другом компьютере локальной сети необходимо:

   а) запустить поиск файла (командой **Пуск | Найти**);

   б) открыть **Сетевое окружение**, указать компьютер и путь к файлу;

   в) переписать файл с другого компьютера локальной сети на дискету.

3. Модем служит для:

   а) приема и передачи информации;

   б) передачи и приема информации;

   в) преобразования сигналов АТС в цифровой вид и обратно;

   г) только для подключения к сети Интернет.

4. Из каких частей состоит адрес электронной почты?

   а) из имени пользователя;

   б) из имени пользователя и имени почтового сервера;

   в) из имени пользователя, знака @ и имени почтового сервера;

   г) из имени пользователя и списка доменов.

5. Как отправить письмо сразу нескольким корреспондентам?

   а) повторить данное сообщение столько раз, сколько получателей;

   б) в поле **Копия** указать адреса получателей;

   в) указать адреса получателей копии сообщения в поле **Тема**.

6. Что такое домен?

   а) буквы RU или COM в адресе электронной почты;

   б) буквы RU или COM в адресе WWW-сервера;

   в) часть адреса, определяющая страну, организацию (фирму) до имени компьютера (сервера);

   г) имя сервера, включая страну и фирму.

7. Можно ли приложить к вашему сообщению или письму файл, содержащий, например рисунок?

   а) нет;

   б) да;

   в) не знаю.

8. Браузер (агент пользователя) предназначен для:

   а) приема информации из сети Интернет;

   б) просмотра информации, навигации по ссылкам и т. д.;

   в) для навигации при работе в Интернете.

9. В виде каких объектов может находиться информация на страничке Интернета?

   а) только в виде текста;

   б) в виде текста и рисунков (картинок);

   в) в виде текста, картинок, видео- и аудиофрагментов, ссылок;

   г) в виде ссылок на другие ресурсы и текста.

10. Выберите из предлагаемого списка наиболее точное определение понятия "ссылка":

    а) текст или картинка, а также другие элементы Web-страницы;

    б) любой объект на Web-странице, нажатие (щелчок) на котором вызовет переход к другому объекту;

    в) текст или картинка, при приближении к которым указатель меняет свою форму на палец.

11. Как осуществить поиск информации в Интернете, если вы не знаете адреса сервера, где она находится?

    а) послать запрос владельцу ресурса по электронной почте;

    б) задать критерии поиска в поисковой машине или каталоге;

    в) ввести нужные слова в строке поиска каталога ресурсов и нажать <Enter>.

12. Как будет выглядеть адрес электронной почты для абонента audit, подключенного к провайдеру (поставщику услуг) cityline, находящемуся в России — ru?

    а) ru@cityline.audit;

    б) audit.cityline.ru;

    в) audit@cityline.ru;

    г) cityline@audit.ru.

13. Из каких частей состоит адрес Web-сервера?
    а) из имени сервера и домена;
    б) из имени компьютера и сервера;
    в) из домена, имени сервера и букв WWW.
14. С помощью какой программы возможен обмен информацией посредством модема без подключения к сети Интернет?
    а) Блокнот;
    б) WordPad;
    в) Terminal;
    г) Internet Explorer;
    д) NetMeeting;
    е) Outlook Express.
15. Назовите наиболее распространенные типы соединений с провайдером (поставщиком услуг):
    а) с помощью интернет-карты;
    б) внутренние и внешние;
    в) локальные и глобальные;
    г) коммутируемое и постоянное;
    д) листовые и планшетные.
16. Как осуществляется переход к следующей порции информации при работе с Web-страницей?
    а) по разному на различных сайтах;
    б) с помощью ссылок;
    в) методы навигации определяет создатель страницы.

6.8. Мультимедийные проекты

В общем то, что мы делали в *разд. "Компьютерные презентации"* тоже можно назвать проектами, но можно еще глобальней. Предложим несколько вариантов.

Практикум 10. Мультимедийные проекты

Вариант 1. Настольно-издательская деятельность

Идея отнюдь не нова, но всякий раз очень привлекательна для учащихся, особенно если их не одергивать, излишне не цензурировать и не ограничивать в действиях.

Например, каждый класс может выпустить журнал либо на свободную, либо на заданную тему. Заранее оговариваются критерии оценки конкурса и инструменты для работы. Жюри оценивает как техническую сторону дела — верстку, дизайн, так и содержательную часть (для этого желательно привлечь учителей гуманитарного цикла). Должно быть и независимое ученическое жюри. Презентация журнала должна быть театрализована и сопровождаться слайд-шоу.

Вариант 2. Генеалогическое древо

Кому ж не хочется знать историю своего рода, не быть эдаким Иваном Непомнящимродства? Проект может реализовываться в течение учебного года. В результате должен быть получен гипертекстовый документ, в котором рассказывается о семье его автора, проиллюстрированный фотографиями из семейного архива и сопровождающийся любимыми семейными мелодиями.

Есть на эту тему хороший сайт "Всероссийское генеалогическое древо" (**www.vgd.ru**) — может, и ваша фамилия там имеется!

Пример такого древа приведен ниже (рис. 6.13).

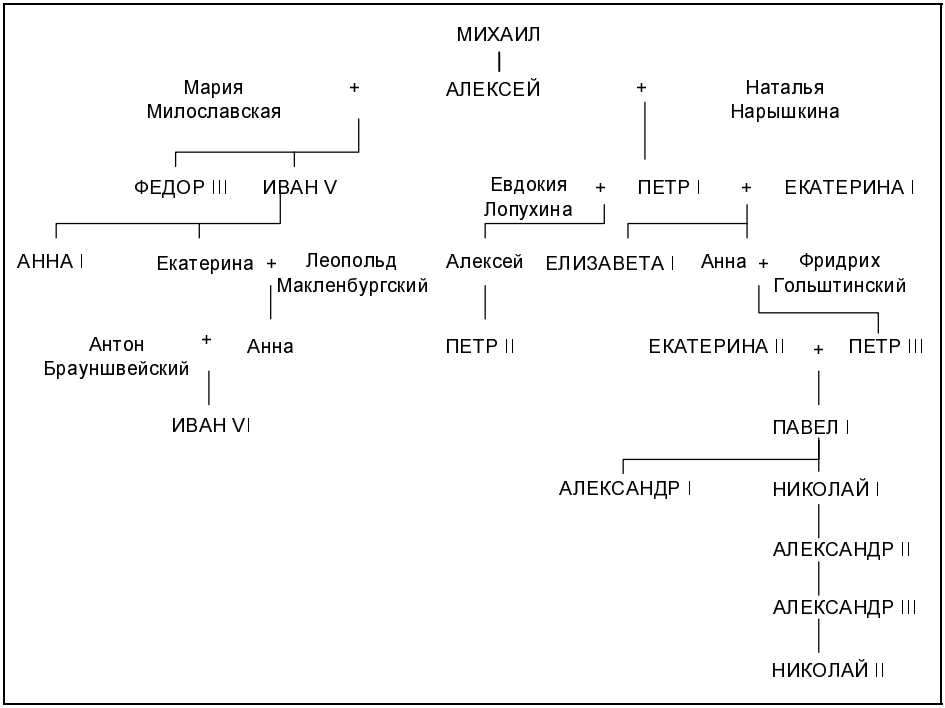


Рис. 6.13. Генеалогическое древо Дома Романовых

6.9. Сетевые игры

Про компьютерные игры говорено-переговорено много. И о ненужности их, и о вредности, и о том, что не место им в учебном процессе. Процитируем, однако, одного известного человека — "Что наша жизнь? Игра!" Игра — это модель жизни. Поэтому автор разрешает своим ученикам иногда играть в компьютерные игры по сети.

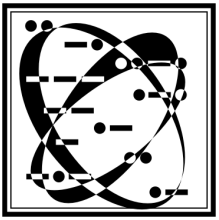
Устраиваем мы и школьные чемпионаты по разным играм. Особой популярностью у ребят пользуются NFS, Quake, Counter Strike.

Организацию такого чемпионата можно смело возложить на самих школьников и сделать это тоже проектом. Удачи!

ЧАСТЬ II

Решения
ИЗБРАННЫХ ЗАДАЧ

Глава 7



Ответы и решения

В этой главе приведены ответы к заданиям, по образцам которых вы сможете выполнить остальные упражнения.

Ответов не очень много — во-первых, по всем вопросам можно обращаться к автору, во-вторых, как-то хочется, чтобы читатель сам подумал, в-третьих, не все бывает однозначно и, особенно в тестировании, если учащийся докажет мне, что он прав, я буду только рад, а вот если он просто угадал правильный ответ и не может ничего пояснить — вот это уже беда.

Глава 1

Вопрос 1

Глаза получают *визуальную (видео)* информацию, уши — *слуховую (аудиальную)*, кожа — *осязательную (тактильную)*, нос — *запаховую (обонятельную)*, *язык* — *вкусовую*. А ведь есть и еще чувства. Если изолировать все прочие органы чувств, человек каким-то образом (через вестибулярный аппарат) получает информацию о положении в пространстве — сидит или стоит, а может быть висит вверх ногами. И это тоже информация. Кроме того, многие всерьез говорят о чувстве интуиции, которое на основе некой подсознательной информации помогает человеку принять правильное решение в критической ситуации.

Вопрос 2

Свойства информации:

- ❒ достоверность (неискажение истинного положения дел);
- ❒ полнота (достаточность для понимания и принятия решений);
- ❒ ценность, актуальность;
- ❒ ясность (понимаемость).

Вопрос 3

- Некомпьютерные — бумага, кинопленка, аудиокассета, пергамент, лазерный диск и т. д.
- Компьютерные — перфокарта, перфолента, магнитные барабаны и диски, лазерные компакт-диски, стримеры и т. д.

Вопрос 4

Ввод, хранение, обработка, передача, вывод.

Вопрос 5

- Появление осмысленной речи.
- Появление письменности.
- Изобретение книгопечатания.
- Изобретение радио.
- Появление персонального компьютера и сети Интернет.

Тест 1

1б, 2в, 3а, 4а, 5б, 6в, 7г, 8бд, 9а — визуальная, 9б — обонятельная, 9в — визуальная, 9г — аудиальная, 9д — визуально-аудиальная, 10а — бумага, 10б — магнитный диск, 10в — магнитная лента, 10г — лазерный диск, 10д — кинопленка, 11.1 — передача, обработка и вывод, 11.2 — ввод, 11.3 — хранение, 11.4 — передача, 11.5 — ввод, 11.6 — хранение, 11.7 — хранение, 11.8 — хранение и ввод, 11.9 — ввод и хранение, 12а — вы — приемник, учебник — источник, 12б: вы — приемник, будильник — источник, 12в: вы — приемник, телевизор — источник, 12г: ученый — приемник, звезда — источник.

Практикум 2

МАХРОСЫЙ КЛОТ

Практикум 5

"Жену свою я не хаю,

И никогда не брошу ее,

Это со мной она стала плохая,

А брал-то я ее хорошую!"

Практикум 7

Составим возможные варианты переданных букв (рис. 7.1).

ГЪЙ	**АЭЕ**	**БПРК**	**ЕЖЩЮ**	**НМЬЧ**	**СЫЗЛ**	**ШДУ**	**ЦХОТ**	**ЯФВИ**
БШЗ	**АЫВ**	**АНОИ**	**ГЕЧЬ**	**ЛКЪХ**	**ПЩЕЙ**	**ЦВС**	**ФУМР**	**ЭТАЖ**
ВЩИ	**БЬЕ**	**БОПЙ**	**ДЁШЭ**	**МЛЫЦ**	**РЪЖК**	**ЧГТ**	**ХВНС**	**ЮУБЗ**
ГЪЙ	**ВЭЁ**	**ВПРК**	**ЕЖЩЮ**	**НМЬЧ**	**СЪЗЛ**	**ШДУ**	**ЦХОТ**	**ЯФВИ**
ДЫК	**ЮЖ**	**ГРСЛ**	**ЁЗЪЯ**	**ОНЭШ**	**ТЬИМ**	**ЩЕФ**	**ЧЦПУ**	**ХГЙ**
ЕЬЛ	**ЯЗ**	**СТМ**	**ЖИЫ**	**ПОЮЩ**	**УЭЙН**	**ЪЁХ**	**ШЧРФ**	**ЦДК**

Рис. 7.1

Выбирая вторую и последнюю группу букв (где есть короткие колонки букв), определяем слова, им соответствующие: ВЯЗ, ЭТАЖ. В исходных словах 33 буквы, поэтому буквы В, Я, З, Э, Т, А, Ж уже использованы, и их можно вычеркнуть из всех колонок (рис. 7.2).

ГЪЙ	**АЭЕ**	**БПРК**	**ЕЖЩЮ**	**НМЬЧ**	**СЫЗЛ**	**ШДУ**	**ЦХОТ**	**ЯФВИ**
БШ		**НОИ**	**ГЕЧЬ**	**ЛКЪХ**	**ПЩЕЙ**	**Ц С**	**ФУМР**	**ЭТАЖ**
ЩИ		**БОПЙ**	**ДЁШ**	**МЛЫЦ**	**РЪ К**	**ЧГ**	**ХВНС**	
ГЪЙ	**В**	**ПРК**	**Е Щ**	**НМЬЧ**	**СЬ Л**	**ШДУ**	**ЦХО**	
		ГРСЛ	**Ё Ъ**	**ОН**	**ЬИМ**	**ЩЕФ**	**ЧЦПУ**	
ЕЬЛ	**ЯЗ**	**С М**	**ИЫ**	**ПО**	**У ЙН**	**ЪЁХ**	**ШЧРФ**	

Рис. 7.2

Из нескольких вариантов, например, в третьей группе: ГНОЙ, ГНОМ, ГРОМ выбираем варианты так, чтобы каждая буква использовалась один раз. Продолжая таким образом, получим ответ.

Ответ: БЫК ВЯЗ ГНОЙ ДИЧЬ ПЛЮЩ СЪЕМ ЦЕХ ШУРФ ЭТАЖ

Практикум 8

Вариант 1

1. COOL NAME ZERO
2. 8.
3. 0,53 Мбайт, 1 дискета.
4. 73,73 Кбайт.

Вариант 2

1. CQRD TREE CAKE
2. 16.
3. 0,8 Мбайт, 1 дискета.
4. 150 Кбайт.

Вариант 3

1. DRUG FOUR OPEN
2. 32.
3. 4,68 Мбайт, 4 дискеты.
4. 11,66 Кбайт.

Вариант 4

1. FIVE PURE VIEW
2. 64.
3. 3,55 Мбайт, 3 дискеты.
4. 54 Кбайт.

Вариант 5

1. NINE FILE MARK
2. 128.
3. 0,68 Мбайт, 1 дискета.
4. 38,88 Кбайт.

Вариант 6

1. BLUE GIRL MOON
2. 256.
3. 1,94 Мбайт, 2 дискеты.
4. 15,36 Кбайт.

Вариант 7

1. WHEN ROCK BULL
2. 512.
3. 3,97 Мбайт, 3 дискеты.
4. 18,14 Кбайт.

Практикум 13

Ответы практикума сведены в табл. 7.1.

Таблица 7.1

Номер варианта	а)	б)	в)
1	10111100110	5364	12220
2	1100100100	6251	11534
3	1000111111	10060	7890
4	10011001010	10773	123CB
5	1110010101	10554	17B7C
6	101111100100	6106	1030B
7	10111110111	11346	15CA3
8	101111110111	11173	F679
9	10010000100	12337	17142
10	110011100100	11024	FE99
11	10111100110	5364	12220
12	1100100100	6251	11534
13	1000111111	10060	7890

Практикум 19

Задача 1. Пять офицеров

Ясно, что каждый офицер имеет двух родственников. Один из них — брат жены, а другой — муж сестры. Обозначим для удобства каждого офицера буквой и расположим их так, чтобы соседом каждого были его родственники (рис. 7.3).

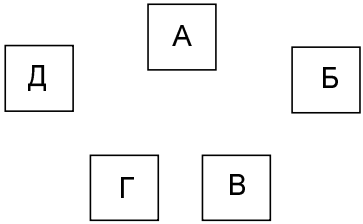


Рис. 7.3

Пусть пехотинец будет обозначен буквой А. Поскольку трое из офицеров служили в Мурманске, а двое там не были, то танкисту должна соответствовать либо буква В, либо Г. Допустим, что танкист — В. Отсюда следует (с учетом условия задачи), что А и В не служили в Мурманске и что Б — генерал. Продолжая рассуждать, приходим к выводу, что Б, В и Г служили в Твери. Поэтому лейтенантом должен быть А или Д и букве Д должен соответствовать артиллерист. Далее, либо В и Г, либо В и Б не служили в Свердловске. Следовательно, А, Д и Б либо А, Д и Г служили в Свердловске. А поскольку мы знаем, что Б не служил в Свердловске, это значит, что там служили А, Д и Г и что Д — полковник. Таким образом, А — лейтенант.

Переходим к следующему этапу решения. А, Б и В служили в Хабаровске, а Д там не служил. Нам известно, что капитан в Хабаровске не служил. Поскольку капитан не может быть Д, следовательно, ему соответствует буква Г. Далее приходим к заключению, что В — майор. Известно, что по меньшей мере один офицер должен быть старше по званию, чем связист. Следовательно, связист не может быть Б и должен быть Г, а саперу соответствует буква Б.

Таким образом, в итоге получается, что лейтенант — пехотинец, генерал — сапер, майор — танкист, капитан — связист, полковник — артиллерист.

Задача 2. Разоблачение оракула

Слева — бог Дипломатии, в центре — бог Лжи, справа — Правды.

Смельчак рассуждал следующим образом. Ясно, что бог, стоящий слева, не может быть богом Правды (в противном случае его заявление было бы ложным, а это противоречит условию), стоящий в центре, судя по его ответу, тоже не может быть богом Правды. Следовательно, бог Правды — крайний справа. А если так, то в центре — бог Лжи, а слева — бог Дипломатии.

Задача 3. Находчивый комендант

В комнате должны жить Дима, Федя, Гриша и Юра.

Задача 4. Четыре "если"

По условию получается, что если Левин бухгалтер, то Набатов счетовод, а если Набатов счетовод, то Митерев бухгалтер. Получается явное противоречие. Следовательно, ни Левин, ни Митерев — не бухгалтеры. Тогда бухгалтер Набатов. Из первого утверждения следует, что Митерев — не счетовод. Следовательно, он кассир, а Левин счетовод.

Задача 5. Любители птиц

Птица, являющаяся "тезкой" владельца грача, должна иметь светлое оперение. Поэтому хозяином грача может быть либо Голубев, либо Канарейкин,

либо Чайкин, либо Попугаев. Первые двое из них холостяки, а грачом владеет муж сестры жены Чайкина. Следовательно, хозяин грача — Попугаев.

У Воронова должна быть птица со светлым оперением, но в то же время надо учесть, что "тезка" птицы, которой владеет Воронов, женат. Следовательно, у Воронова либо попугай, либо чайка. Но попугай у Воронова не может быть, потому что птица, являющаяся "тезкой" хозяина попугая, принадлежит "тезке" той птицы, владельцем которой является Воронов, а у Попугаева, как мы знаем, дома живет грач. Поэтому Воронов — хозяин чайки.

У Грачева живет либо попугай, либо чайка, либо голубь. Но чайка принадлежит Воронову, а если попугай принадлежал бы Грачеву, тогда у Попугаева было бы две птицы — грач и канарейка. Поэтому голубь принадлежит Грачеву, а Голубеву принадлежит канарейка.

Владелец ворона не женат. Следовательно, ворон может принадлежать только Канарейкину. Тогда хозяин скворца — Чайкин, а хозяин попугая — Скворцов.

Задача 6. Финальный забег

Для того чтобы решить задачу, необходимо, прежде всего, выяснить, какое из девяти приведенных утверждений ложно.

Итак, проанализируем данные утверждения. Если утверждение (9) заведомо верно (по условию задачи), то нетрудно установить, что утверждения (4), (5), и (6) не могут быть истинными одновременно.

Действительно, если истинно утверждение (6), то 3 первые места разделили между собой 2 африканские и 1 американская команды, либо 1 африканская и 2 американские команды. Но по утверждению (5) 2 африканские команды не могли быть среди тех, кто вышел на первые три места, а по утверждению (4) 2 американские команды могли занять лишь первое и третье места. Кроме того, из этого же утверждения (4) следует, что на второе место вышла европейская команда и, следовательно, среди обладателей трех первых мест не было ни одной африканской команды.

Таким образом, ложные сведения должны содержаться в каком-то из утверждений (4), (5) и (6), а остальные утверждения истинны, т. к. по условию задачи ложным является только одно утверждение.

Воспользуемся сначала заведомо истинными утверждениями.

Объединим утверждения (1), (3) и (7) в одно, т. к. они взаимосвязаны между собой. Прочитав их в последовательности (1)—(7)—(3), приходим к следующему выводу: если исключить команды C и E, то представители остальных команд могли прийти к финишу лишь в последовательности A, F, B, D. Следовательно, среди команд, занявших 3 первые места, заведомо должна быть европейская команда A. В худшем случае, она могла выйти на третье место, но оказалась среди призеров. Это означает, что утверждение (6) ложно. Чтобы определить, какое место заняла каждая из шести команд, распо-

ложим истинные утверждения в следующем порядке: (2), (4), (5), (8), (1)—(7)—(3), (9).

Как видно из утверждений (2) и (1)—(7)—(3), первое место могла занять лишь команда C, поскольку команда D заведомо не вышла на первое место. По утверждению (5), команда D могла занять лишь последнее, шестое место.

Утверждения (8) и (1)—(7)—(3) позволяют схематически изобразить распределение мест между четырьмя остальными командами так, как показано на рис. 7.4 (острие стрелки направлено к команде, показавшей лучшее время, конец — к команде, занявшей последнее место).

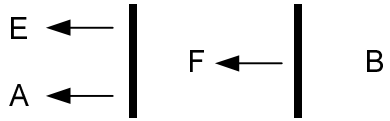


Рис. 7.4

Остается невыясненным, какая из команд: E или A — показала лучший результат. Эту неопределенность помогает разрешить утверждение (4). Согласно схеме, между двумя американскими командами E и F могла "вклиниться" только европейская команда A.

Следовательно, представители четырех команд, о которых идет речь, могли пересечь линию финиша лишь в следующей последовательности: E, A, F, B. Это означает, что команда C заняла первое место, E — второе, A — третье, F — четвертое, B — пятое и команда D — шестое место.

Задача 7

Зебра принадлежит итальянцу. А вот все остальное... Разберитесь.

Глава 2

Тест 2

1в, 2в, 3а, 4а, 5г, 6бвг, 7г, 8а, 9в,10а

Тест 3

1г, 2а, 3в, 4а, 5г, 6а, 7а, 8в, 9в, 10а, 11в, 12б, 13в, 14а, 15б, 16в, 17а, 18в, 19г, 20г

Тест 5

1а, 2а, 3г, 4в, 5а, 6г, 7б, 8а, 9г, 10а, 11в, 12в, 13в, 14б, 15а

Глава 4

Свойства алгоритмов

Вопросы

6. Продолжите ряд:

- л (буквы русского алфавита идут через одну);
- 64 (каждое предыдущее число умножается на 2);
- 36 (квадраты натуральных чисел);
- 55 (так называемые числа Фибоначчи, где каждое последующее число образуется суммой двух предыдущих);
- 3 (перечисляется последовательность чисел, начиная с 1, но двузначные числа представляются в виде составляющих их цифр: 1, 0 — 10; 1, 1, — 11; 1, 2 — 12, и т. д.);
- да (каждое последующее слово образуется отсечением первой буквы от предыдущего);
- о (первые буквы числительных, начиная с одного: о — один, д — два, т — три и т. д.).

7 и 8. Ответы на задания на нестандартное мышление приводить не хочется, если уж совсем никак — пишите автору.

Задачи

Черные ящики

1. Умножает входное число на 2 и прибавляет 1.

3. Подсчитывает количество букв "О".

5. Подсчитывает количество удвоенных букв.

6. Заменяет каждую букву исходного слова на следующую по алфавиту.

8. Делит четные числа на 2 и вычитает 1.

9. В словах с четным количеством букв делит это количество на 2.

11. Указывает букву в слове, дальше всех стоящую в алфавите из букв этого слова.

12. Находит разность между наибольшей и наименьшей цифрами числа.

13. Находит сумму цифр введенного числа.

14. Делит четные числа последовательно на 2, пока делится.

15. Определяет номер каждой буквы в алфавите и находит общую сумму этих номеров.

16. Четные числа делит на 2, а нечетные умножает на 2.

17. Заменяет каждую цифру введенного числа следующей (а цифру 9 — цифрой 0)

18. Находит целочисленный остаток от деления на 3.

20. Переводит десятичные числа в восьмеричную систему.

21. Если число делится на 3 без остатка, то оставляет его без изменений, иначе выводит число, меньшее и ближайшее к введенному, которое делится на 3 без остатка.

23. Выводит 2 цифры, первая из которых обозначает количество гласных букв в слове, а вторая — количество всех остальных.

25. А вот над этим нестандартным заданием подумайте сами. А если не додумаетесь, пишите автору.

Алгоритмы

26. Ну, про перевозчика обычно в младшей школе рассказывают, так что постарайтесь вспомнить.

27. Мужья и жены.
Вначале переправляются две жены. Затем одна из жен возвращается и перевозит на другой берег третью жену. После этого одна из трех жен возвращается к своему мужу и с ним остается на первом берегу, два других мужа отправляются к своим женам. Затем один из мужей возвращается со своей женой, оставляет ее, а с собой забирает мужа, оставшегося на этом берегу. Теперь остававшаяся на другом берегу жена переезжает и забирает с собой одну из двух жен, а следующим рейсом забирает последнюю жену. И начинается веселый пикник!

28. Переливашка.
Решение первого задания (оставить 1 литр в сосуде А) входит в решение второго задания (деление пополам), для которого приводим два варианта решения (табл. 7.2, 7.3).

Таблица 7.2. *Вариант 1*

	Сосуды		
	А (8 л)	**В (5 л)**	**С (3 л)**
Начало	8	0	0
1-ый ход	3	5	0
2-ой ход	3	2	3
3-ий ход	6	2	2
4-ый ход	6	0	2
5-ый ход	1	5	2

Таблица 7.2 *(окончание)*

	Сосуды		
	А (8 л)	**В (5 л)**	**С (3 л)**
6-ой ход	1	4	3
7-ой ход	4	4	0

Таблица 7.3. *Вариант 2*

	Сосуды		
	А (8 л)	**В (5 л)**	**С (3л)**
Начало	8	0	0
1-й ход	5	0	3
2-й ход	5	3	0
3-й ход	2	3	3
4-й ход	2	5	1
5-й ход	7	0	1
6-й ход	7	1	0
7-й ход	4	1	3
8-й ход	4	4	0

29. Ханойские башни.

Алгоритм решения для трех колец

- ❒ 1-й ход 1—3
- ❒ 2-й ход 1—2
- ❒ 3-й ход 3—2
- ❒ 4-й ход 1—3
- ❒ 5-й ход 2—1
- ❒ 6-й ход 2—3
- ❒ 7-й ход 1—3

Алгоритм решения для четырех колец.

Задача сводится к предыдущей следующим образом. Переносить пирамидку из трех колец мы уже умеем. Значит, делаем так. Переносим пирамидку из трех верхних колец по уже данному выше алгоритму, только не на третий стержень, а на второй. Это займет 7 ходов. Потом переносим освободившееся нижнее, самое большое кольцо, на третий стержень. И снова задача, уже нами решенная, перенос пирамидки из трех колец на третий стержень со второго за те же 7 ходов. Итого ходов — **15**. Такой способ решения, основанный на возврате, называется *рекурсией*.

Теперь можно вывести и формулу минимального количества шагов для любого количества колец.

Пусть N — минимальное количество ходов, а K — количество колец, тогда

$$N = 2K - 1.$$

30. Сумматор.

Конечно, можно просто прибавить к исходному нулю единицу сто раз и задача будет решена. Но перед началом решения задачи стоило бы подумать, наверное, с конца — как вообще можно представить число **100**. Например $50 \times 2 = 100$ или $25 \times 4 =$ **100**. Тогда вот один из вариантов решения:

1-й ход	$0 + 1 = 1$
2-й ход	$1 + 1 = 2$
3-й ход	$2 + 1 = 3$
4-й ход	$3 \times 2 = 6$
5-й ход	$6 \times 2 = 12$
6-й ход	$12 \times 2 = 24$
7-й ход	$24 + 1 = 25$
8-й ход	$25 \times 2 = 50$
9-й ход	$50 \times 2 = 100$

31. Шахматные кони 1

1-й ход	A1-C2
2-й ход	B3-A1
3-й ход	C1-B3
4-й ход	C3-A2
5-й ход	B1-C2
6-й ход	A2-C1
7-й ход	A3-B1
8-й ход	C3-A3

Шахматные кони 2.

А вот здесь — порешайте-ка сами, возьмите шахматную доску, четырех коней и вперед! Решается задачка за 15 ходов.

33. Взвешивание монет.

Для семи монет, одна из которых легче.

Берем две и две монеты и кладем их на чаши весов. Если одна из чашек оказалась легче, то понятно, что фальшивая монета на этой чашке и есть, и за следующее взвешивание мы ее выявляем. Если же весы после первого взвешивания в равновесии, то значит фальшивая — среди оставшихся трех. Берем две из них и кладем на чашки весов, которая легче — та и фальшивая, если же чашки в равновесии, значит фальшивая — оставшаяся.

И для девяти монет, если заранее известно, что фальшивая монета легче, достаточно двух взвешиваний — разделить надо на три группы по три монеты, а далее как и в предыдущем примере.

А вот для девяти монет с фальшивой монетой, отличающейся по весу, но неизвестно в какую сторону — легче она или тяжелее, понадобится 3 взвешивания.

Первое — разбиваем монеты на группы по 3 и кладем две из них на чашки весов. Для последующих взвешиваний возможны два варианта.

- Вариант 1. Весы находятся в равновесии. Значит на весах в данный момент все шесть монет настоящие, а фальшивая находится в третьей группе. Берем две настоящих монеты и две из третьей группы и кладем на чашки весов. Если весы в равновесии, то фальшивая монета оставшаяся, и за оставшееся третье взвешивание мы просто определим, легче она или тяжелее настоящей. Если же весы не в равновесии, то, во-первых, мы сразу определяем, легче или тяжелее фальшивая монета, т. к. на одной из чашек у нас настоящие монеты, а, во-вторых, за последнее взвешивание легко находим фальшивую из двух возможных.
- Вариант 2. Весы не в равновесии уже после первого взвешивания. Значит, оставшиеся три монеты настоящие. Берем эти три монеты и три монеты, допустим, с левой чашки весов. Кладем их на весы. Если весы в равновесии, то, значит, фальшивая монета среди тех, что при первом взвешивании были на правой чашке. Кроме того, мы теперь можем понять, легче или тяжелее фальшивая монета настоящих, и за третье взвешивание определяем ее из трех оставшихся. Если же весы не в равновесии, то мы тоже определяем группу с фальшивой монетой, и находим ее.

А вот для двенадцати монет не все так тривиально. Пока оставляем вас в неведении. Пишите. А лучше все-таки решите!

34. Угадай-ка.

Конечно, при хорошо развитой интуиции можно угадать и с первого раза, но есть алгоритм гарантированного угадывания за минимальное количество ходов. Он довольно прост — надо каждый раз брать середину числового интервала. Первый раз (если загадано число до 100), надо называть **50**. Если загаданное число больше, то 75, а если меньше, то 25 и т. д. Таким образом, при загаданном числе от 1 до 100 — гарантированное отгадывание возможно за 7 ходов, от 1 до 1000 — за 10, а от 1 до 1 000 000 — всего лишь за 20 (потому что 2^{20}=1 048 576).

35. Игра Баше.

Для выигрышной стратегии второй игрок своим первым ходом должен взять столько же, сколько и первый, а каждый последующий ход брать столько, чтобы в сумме с первым игроком получалось 4. Вот пример партии для 15 предметов (табл. 7.4).

Таблица 7.4. *Игра Баше*

Первый игрок	Второй игрок
Взял 3 (осталось 12)	Взял 3 (осталось 9)
Взял 1 (осталось 8)	Взял 3 (1 + 3 = 4, осталось 5)
Взял 2 (осталось 3)	Взял 2 (2 + 2 = 4, осталась 1)
Взял последнюю и проиграл...	

36. Бочонок.

Алгоритм решения этой парадоксальной задачи изображен на рис. 7.5. Рекомендуется проанализировать его.

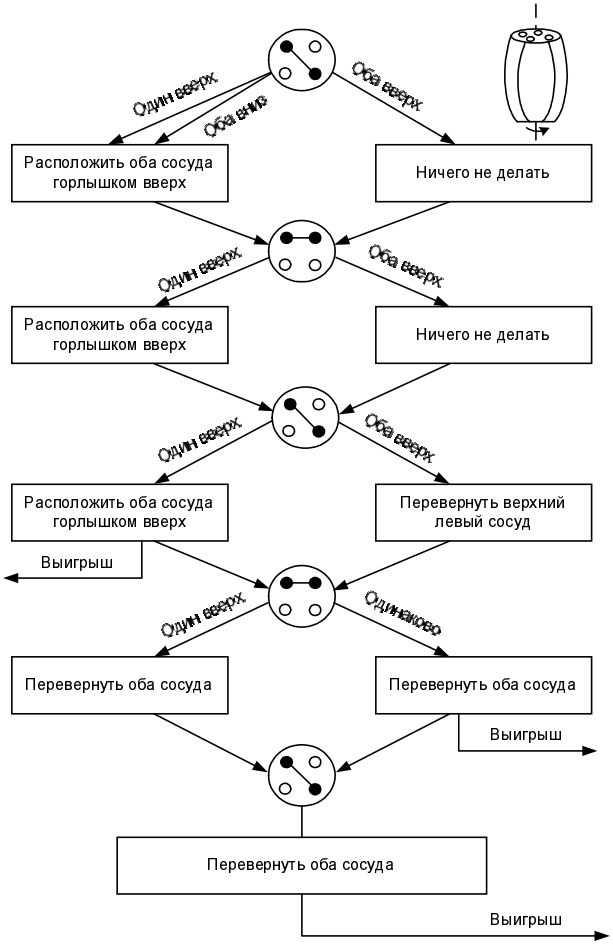


Рис. 7.5. Бочонок

Арифметика в программировании

42. `1/(2+3/(4+5/(6+7/8)))`

46. `sqr(3+sqr(3+sqr(3+sqr(3))))`

54. 3

57. 2

60. 1

61. 0

Оператор присваивания

67. Нет. Нет знака равенства.

68. Да.

70. Нет. Переменные связаны двумя знаками равенства.

71. Нет. Слева от знака равенства стоит арифметическое выражение.

77. `X=6`, `Y=6`

78. Необходимо ввести вспомогательную переменную `A`.

```
A=X
X=Y
Y=A
```

79. `W = K\100 + (K MOD 100)\10 + (K MOD 100) MOD 10`

80. `S=3*10^5*365*24*60*60`

Стандартные функции Бейсика

85. `COS(X^3)^2`

88. `(X^7+7^X)^(1/7)`

89. `(A+SIN(B^2)^3)/(COS(25)+ABS(1/TAN(60)))`

96. `B*TAN(X)^2-A/SIN(X/A)^2+A*EXP(-SQR(A))/COS(B*X/A)`

105. $$\frac{e^{-bt}\sin(at+b)-\sqrt{|bt+a|}}{b\sin^3 t^2-1}$$

Линейные алгоритмы

111.

```
' Полет к ближайшей звезде
CLS
' Длина светового года
S = 3*10^5*365*24*60*60
' Количество секунд полета
NSEC = S/100
' Количество часов полета
NHOUR = NSEC/3600
' Количество суток полета
NDAYS = NHOUR/24
' Количество лет полета
NYEARS = NDAYS/365
' Вывод результата
LOCATE 25, 12
?"С такой скоростью звездолет долетит до звезды за"; NYEARS; "лет"
END
```

115.

```
' Информационный объем книги
CLS
' Количество символов в книге
N = 450*60*80
' Перевод в Мегабайты
M = N/1024/1024
' Количество дискет
DISCS = INT(M/1.44)+1
' Вывод результата
?"Для записи книги понадобится"; DISCS; "дискет"
END
```

120.

```
'Деньги в банке
CLS
M=22000
N1=M+M*.01            'Первый месяц
N2=N1+N1*.01          'Второй месяц
```

```
N3=N2+N2*.01          'Третий месяц
N4=N3+N3*.01          'Четвертый месяц
N5=N4+N4*.01          'Пятый месяц
N6=N5+N5*.01          'Шестой месяц
N7=N6+N6*.01          'Седьмой месяц
N8=N7+N7*.01          'Восьмой месяц
N9=N8+N8*.01          'Девятый месяц
N10=N9+N9*.01         'Десятый месяц
N11=N10+N10*.01       'Одиннадцатый месяц
N12=N11+N11*.01       'Двенадцатый месяц
'Вывод результата
PRINT "Через год у меня на счету будет"; N12; "$"
END
```

Конечно, красивее было бы решение с циклами. Когда их изучите, решите эту задачу снова.

Графика в Бейсике

140.

```
'Российский флаг
CLS
' Белая полоса
LINE (100, 50)-(500, 130), 15, BF
' Синяя полоса
LINE (100, 130)-(500, 210), 1, BF
' Красная полоса
LINE (100, 210)-(500, 290), 4, BF
END
```

147.

```
' Летающая тарелка
SCREEN 9
' Основа тарелки
CIRCLE (320, 175), 200, 2, , , .3
' Рубка
CIRCLE (320, 70), 100, 2, , , .2
LINE (198, 125)-(220, 70), 2
LINE (420, 70)-(440, 125), 2
```

```
' Иллюминаторы
CIRCLE (150, 175), 10, 4
CIRCLE (490, 175), 10, 4
CIRCLE (200, 200), 10, 4
CIRCLE (440, 200), 10, 4
CIRCLE (320, 220), 10, 4
' Ноги
LINE (320, 235)-(320, 280), 2
LINE (197, 223)-(165, 310), 2
LINE (443, 223)-(475, 310), 2
END
```

152.

```
'Гистерезис
SCREEN 9
' Горизонтальные составляющие
LINE (300, 126)-(500, 126), 13
LINE (100, 274)-(300, 274), 13
' Левая верхняя четверть дуги
CIRCLE (300, 200), 100, 13, 1.57, 3.14
' Левая нижняя четверть дуги
CIRCLE (100, 200), 100, 13, 4.71, 0
' Правая верхняя четверть дуги
CIRCLE (500, 200), 100, 13, 1.57, 3.14
' Правая нижняя четверть дуги
CIRCLE (300, 200), 100, 13, 4.71, 0
' Закраска
PAINT (300, 200), 14, 13
END
```

160.

```
'Елочка
SCREEN 9
PRESET (320, 300)
DRAW "C2 L60 E26 L25 E26 L25 E55 F55 L25 F26 L25 F26 L60"
'Закраска
PAINT (320, 250), 2
' Ствол
LINE (300, 300)-(340, 340), 6, BF
END
```

Разветвляющийся алгоритм

163.

```
'Бегущий узор
1 : PRINT "***"
PRINT "   *******"
PRINT "          *"
PRINT "          *"
PRINT "          *"
PRINT "   *******"
PRINT "***"
GOTO 1
END
```

Условный переход

168.

```
'Чет-нечет
CLS
INPUT "Введите целое число"; N
IF N MOD 2 = 0 THEN LOCATE 12, 35: PRINT "Это число четное" ELSE
LOCATE 12, 35: PRINT "Это число нечетное"
END
```

171.

Переменная `X` примет следующие значения:

- `X=1`
- `X=2`
- `X=0`

174. `S=6`

188.

```
' Старояпонский календарь
CLS
' Запрос года
INPUT "Введите год"; YEAR
' Вычисление номера цвета и животного
C =((9910-YEAR) MOD 60)\12
K =(YEAR-3) MOD 12
```

```
' Определение по номеру цвета года
SELECT CASE C
  CASE 0
    PRINT "ЗЕЛЕНЫЙ";
  CASE 1
    PRINT "КРАСНЫЙ";
  CASE 2
    PRINT "ЖЕЛТЫЙ";
  CASE 3
    PRINT "БЕЛЫЙ";
  CASE 4
    PRINT "ЧЕРНЫЙ";
END SELECT
' Определение по номеру животного года
SELECT CASE K
  CASE 0
    PRINT " КАБАН"
  CASE 1
    PRINT " КРЫСА"
  CASE 2
    PRINT " БЫК"
  CASE 3
    PRINT " ТИГР"
  CASE 4
    PRINT " КОТ"
  CASE 5
    PRINT " ДРАКОН"
  CASE 6
    PRINT " ЗМЕЯ"
  CASE 7
    PRINT " ЛОШАДЬ"
  CASE 8
    PRINT " ОВЦА"
  CASE 9
    PRINT " ОБЕЗЬЯНА"
  CASE 10
    PRINT " ПЕТУХ"
```

```
  CASE 11
    PRINT " СОБАКА"
END SELECT
PRINT "УДАЧИ ВАМ!"
END
```

Циклический алгоритм

192.

```
' Пешеходный переход "зебра"
FOR Y = 10 TO 320 STEP 30
LINE (200, Y) - (400, Y+20), 15, BF
NEXT Y
```

196.

```
'Воздушный шарик
SCREEN 9
' Изменение цвета фона
COLOR , 1
FOR I = 1 TO 201
  ' Рисование окружности изменяющегося радиуса
  CIRCLE (320, 175), I, 14
' Пауза
FOR T = 1 TO 2000: NEXT T
  ' Стирание окружности изменяющегося радиуса
  CIRCLE (320, 175), I, 1
NEXT I
```

199.

```
F=1
```

Цикл вообще не будет ни разу исполняться, т. к. начальное значение параметра больше конечного, а шаг по умолчанию принят +1.

200.

```
' Да ты крут!
CLS
INPUT "Введи свой возраст"; N
FOR i = 1 TO N
  PRINT "ДА ТЫ КРУТ!"
  FOR T = 1 TO 5000: NEXT T
NEXT i
END
```

201.

```
' Таблица умножения
1:INPUT "Введите целое число от 2 до 9"; N
IF N < 2 OR N > 9 THEN ?"От 2 ДО 9!": GOTO 1
? "Таблица умножения на"; N
FOR I = 2 TO N
? I; "*"; N; "="; I*N
NEXT I
END
```

204.

```
' Электронный секундомер
FOR T = 1 TO 3600
LOCATE 12, 38:? T
NEXT T
END
```

205.

```
' Орнамент с полукольцами
SCREEN 9
' Задание радиусов окружностей
R1 = 20: R2 = 10
FOR X = 50 TO 600 STEP 4 * R1
' Первое полукольцо
CIRCLE (X, 175), R1, 14, 0, 3.14
CIRCLE (X, 175), R2, 14, 0, 3.14
LINE (X - R1, 175)-(X - R2, 175), 14
LINE (X + R1, 175)-(X + R2, 175), 14
PAINT (X, 162), 2, 14
' Второе полукольцо
CIRCLE (X + 2 * R1, 175), R1, 14, 3.14, 0
CIRCLE (X + 2 * R1, 175), R2, 14, 3.14, 0
LINE (X + R1, 175)-(X + R1 + R2, 175), 14
LINE (X + 2 * R1 + R2, 175)-(X + 3 * R1, 175), 14
PAINT (X + 2 * R1, 188), 4, 14
NEXT X
END
```

209.

а) Правильно, если переменные `R`, `S`, `H` ранее получили значения.

б) Правильно.

в) Правильно.

г) Неправильно. Отсутствует ключевое слово TO.

д) Правильно.

е) Неправильно. Цикл исполняться не будет, т. к. начальное значение параметра больше конечного, а шаг неотрицателен.

ж) Неправильно, т. к. `WEND` не должен содержать параметр.

210.

```
WAR

WAR

WAR

SUNDAY
PEACE
```

Так как после окончания цикла параметр имеет значение большее конечного значения параметра, то проверяемое условие `R=w` ложно, следовательно выполняется следующий оператор печати слова `SUNDAY`, а затем `PEACE`.

222. Будут бесконечно печататься нули. Значение `S` не меняется, поэтому цикл будет длиться бесконечно.

226. `X = INT(RND(1)*11) + 10`

228. `X = -INT(RND(1)*101) + 65`

232.

```
' Маленький взрыв
RANDOMIZE TIMER
SCREEN 9
FOR I = 1 TO 100
X = INT (RND(1)* 200) + 220
Y = INT (RND(1)* 200) + 75
C = INT (RND(1)*15)+1
LINE (X, Y) - (320, 175), C
NEXT I
END
```

241.

```
' Построение графика Y = SQR (X)
SCREEN 9
LINE (0, 175) — (640, 175), 15        Rem ось абсцисс
LINE (320, 0) — (320, 350), 15        Rem ось ординат
```

```
DEF FNY(X) = SQR (X)
FOR X = 320 TO 640
  Y = 175 — 30*FNY((X — 320)/30)
  PSET (X, Y), 15
NEXT X
END
```

Пояснение: корень квадратный извлекается только из неотрицательных чисел, следовательно, аргумент функции `(X-320)/30)` не должен быть отрицателен, что и достигается его начальным значением **320**.

243.

```
' Поверхность вращения функции y = 1/(1+x^2)
SCREEN 9
LINE (0, 175) — (640, 175), 15        Rem ось абсцисс
LINE (320, 0) — (320, 350), 15        Rem ось ординат
DEF FNY(X) = 1/(1+x^2)
FOR X = 320 TO 640 STEP 5
  Y = 175 — 30*FNY((X — 320)/30)
  CIRCLE (X, 175), 175 - Y,15,,,3
NEXT X
END
```

249.

```
' Лестница 1
SCREEN 9
FOR X = 0 TO 640 STEP 40
LINE(X, X*35/64) - (X + 40, (X + 40)*35/64),15,B
NEXT X
END
```

Так как экран представляет собой не квадрат, а прямоугольник, то чтобы серия прямоугольников располагалась вдоль диагонали, приходится вводить коэффициент соотношения сторон экрана 350/640 или 35/64.

250.

```
' Пирамида. Вид сверху
SCREEN 9
FOR X = 20 TO 220 STEP 10
LINE (X, X) - (220-X, 220-X),15, B
NEXT X
END
```

Задача 253

```
' Лоскутный ковер
SCREEN 9
RANDOMIZE TIMER
FOR X = 0 TO 640 STEP 10
  FOR Y = 0 TO 350 STEP 10
   ' Выбор случайного цвета для закраски кв. лоскута
   C = INT(RND(1) * 15) + 1
   ' Рисование закрашенного квадратика
   LINE (X, Y)-(X + 10, Y + 10), C, BF
   ' Обводка квадратика белым контуром
   LINE (X, Y)-(X + 10, Y + 10), 15, B
  NEXT Y
NEXT X
END
```

258.

```
' Хронометр
CLS
' Цикл для часов
FOR HOURS = 0 to 23
' Цикл для минут
  FOR MINUTES = 0 to 59
' Цикл для секунд
    FOR SECONDS = 0 to 59
    LOCATE 12, 35
    ? HOURS; ":"; MINUTES; ":"; SECONDS
    SLEEP 1
    NEXT SECONDS
  NEXT MINUTES
NEXT HOURS
```

261.

```
' Вычисление числа Пи методом Монте-Карло
CLS : RANDOMIZE TIMER
LOCATE 3, 20: PRINT " Вычисление числа Пи методом Монте-
Карло"
SCREEN 9
' Рисование квадрата со стороной 200LINE (100, 100)-(300,
300), 14, B' Рисование вписанной в квадрат окружности радиу-
сом 100CIRCLE (200, 200), 100, 0, , , 1.01' Задание общего
количества песчинок N' и обнуление счетчика попавших в круг
KN = 5000: K = 0
```

```
FOR I = 1 TO N
X = INT(RND(1) * 200) + 100
Y = INT(RND(1) * 200) + 100' Вычисление расстояния от выпав-
шей точки до центра
S = SQR((X - 200) ^ 2 + (Y - 200) ^ 2)' Проверка условия, по-
пала ли точка в круг
' и изображение ее красным или белым цветамиIF S <= 100 THEN
K = K + 1: PSET (X, Y), 4 ELSE PSET (X, Y), 15' Вывод на эк-
ран счетчиков песчинокLOCATE 5, 20: PRINT "Выпала песчинка
№"; I
LOCATE 6, 20: PRINT "В круг уже попало"; K; "песчинок"
NEXT I
' Вычисление и вывод на экран значения числа Пи
LOCATE 23, 15: PRINT "Число ПИ для"; N; "точек ="; 4 * K / N
END
```

265.

```
' Сумматор
CLS
' Запрос с клавиатуры количества суммируемых чисел
INPUT "Сколько чисел будем складывать"; N
' Обнуление переменной для накапливания суммы
S = 0
FOR I = 1 TO 5
? "Введите"; I; "число";
INPUT A
S = S + A
NEXT I
' Вывод результата на экран
?"Сумма ваших"; N; "чисел равна"; S
END
```

268.

```
' Нахождение суммы ряда
CLS
' Запрос значений переменных A и B
INPUT "Введите значения A и B"; A, B
' Обнуление переменной для накапливания суммы
S = 0
I = 0
WHILE (A + I) <> B
S = S +1 / (1 + (A + I)^2)
I = I + 1
WEND
' Вывод результата на экран
?"Сумма ряда равна"; S
END
```

269.

```
'Соприкасающиеся окружности
'Занесение значений радиусов
DATA 50, 20, 10, 25, 40, 15, 60
SCREEN 9
'Задание начальной координаты X
```

```
X = 0
FOR I = 1 TO 7
READ R
CIRCLE (X + R, 175), R, 14
X = X + 2 * R
NEXT I
END
```

273.

```
' Разноцветные окружности. Вариант 1
' Программа с DO ... LOOP WHILE
SCREEN 9
' Задание значения первого цвета
C = 1
DO
' Определение координаты X центра окружности
X = 10 + 40 * C
CIRCLE (X, X), 30, 15
PAINT (X, X), C, 15
C = C + 1
LOOP WHILE C < 7
END

' Разноцветные окружности. Вариант 2
' Программа с DO ... LOOP UNTIL
SCREEN 9
' Задание значения первого цвета
C = 1
DO
' Определение координаты X центра окружности
X = 10 + 40 * C
CIRCLE (X, X), 30, 15
PAINT (X, X), C, 15
C = C + 1
LOOP UNTIL C = 7
```

Символы и строки

275.

```
' Шифровка
CLS
' Запрос количества букв в шифруемом слове
INPUT "Сколько букв в слове"; N
' Объявление массива для зашифрованного слова
DIM D$(N)
FOR I = 1 TO N
' Побуквенный запрос исходного слова
PRINT "Введите"; I; "букву";
```

```
INPUT C$
' Побуквенная шифровка
D$(I) = CHR$(ASC(C$) + 1)
NEXT I
' Вывод на экран зашифрованного слова
FOR I = 1 TO N
PRINT D$(I);
NEXT I
END
```

Обратите внимание, что при введении буквы `Z` в результате получается не буква, а символ квадратной скобки. Как избежать этого? Подумайте сами.

282.

```
' Палиндром
CLS
' Запрос слова с клавиатуры
INPUT "Введите ваше слово"; W$
' Определение длины введенного слова
N = LEN(W$): W1$ = ""
' Цикл переворачивания исходного слова задом наперед
FOR I = N TO 1 STEP -1
W1$ = W1$ + MID$(W$, I, 1)
NEXT I
' Вывод исходного и получившегося слов
PRINT "Исходное слово - "; W$
PRINT "Исходное слово наоборот - "; W1$
' вывод на экран результата
IF W$ = W1$ THEN PRINT "Да, это палиндром" ELSE PRINT "Нет, это не
палиндром"
END
```

Обратите внимание, что программа для фразы "А РОЗА УПАЛА НА ЛАПУ АЗОРА" не сработает. Разберитесь, почему и внесите необходимые поправки.

284.

```
' 10 слов
DATA STUFF, CAR, SKI, QUIKE, SUN
DATA CARD, PRESS, POOH, GULI, COP
CLS
```

```
PRINT "Все слова исходного списка"
FOR I = 1 TO 10
READ W$
PRINT W$; " ";
NEXT I
PRINT : PRINT
RESTORE
PRINT "Слова списка, отличные от слова SUN"
FOR I = 1 TO 10
READ W$
IF W$ <> "SUN" THEN PRINT W$; " ";
NEXT I
PRINT : PRINT
RESTORE
' Определение слова, ближайшего к началу алфавита
READ W$
ALFA$ = W$
FOR I = 1 TO 9
READ W$
IF W$ < ALFA$ THEN ALFA$ = W$
NEXT I
PRINT "Слово в списке, ближайшее к началу алфавита - "; ALFA$
PRINT
RESTORE
' Определение слова, составленного из последних букв всех слов
списка
R$ = ""
FOR I = 1 TO 10
READ W$
R$ = R$ + RIGHT$(W$, 1)
NEXT I
PRINT "Слово, составленное из последних букв всех слов списка - ";
R$
PRINT
RESTORE
PRINT "Все трехбуквенные слова из списка"
FOR I = 1 TO 10
READ W$
```

```
IF LEN(W$) = 3 THEN PRINT W$; " ";
NEXT I
END
```

289.

```
' Замена буквосочетаний
DATA PHOTO, GRAPH, PHILOPHON, COPHE
CLS
FOR I = 1 TO 4
READ W$
  FOR J = 1 TO LEN(W$)
  P$ = MID$(W$, J, 1): H$ = MID$(W$, J + 1, 1)
  IF P$ <> "P" THEN PRINT P$;  ELSE IF H$ <> "H" THEN PRINT H$;
ELSE
  PRINT "F"; : J = J + 1
  NEXT J
PRINT
NEXT I
END
```

Массивы

290.

```
' Столбиковая интерпретация массива
CLS : RANDOMIZE TIMER
INPUT "Сколько чисел будет в массиве"; N
DIM S(N)
' Формирование массива
FOR I = 1 TO N
S(I) = INT(RND(1) * 150) + 50
NEXT I
' Графическая интерпретация массива
SCREEN 9
X = 100
FOR I = 1 TO 10
' Столбик, интерпретирующий элемент массива
LINE (X, 300)-(X + 10, 300 - S(I)), 2, BF
' Белая рамка, обрамляющая столбик
LINE (X, 300)-(X + 10, 300 - S(I)), 15, B
X = X + 10
```

```
NEXT I
' Вывод элементов массива на экран
FOR I = 1 TO N
PRINT S(I);
NEXT I
END
```

292.

```
' Вычисление среднего квадратичного элементов заданного массива
DATA 31, 19, 52, 65, 6, 8, 13, 16, 97, 33
CLS
DIM S(10)
' Формирование массива и вывод его элементов на экран
FOR I = 1 TO 10
READ S(I)
? S(I);
NEXT I
' Вычисление среднего арифметического элементов массива
T = 0
FOR I = 1 TO N
T = T + S(I)
NEXT I
TR = T/N
' Вычисление среднего квадратичного элементов массива
T = 0
FOR I = 1 TO N
T = T +(S(I)-TR)^2
NEXT I
SRKV = S/N
? "Среднее арифметическое элементов массива = "; TR
? "Среднее квадратичное элементов массива = "; SRKV
END
```

294.

```
' Вывод на экран суммы элементов массива по заданному критерию
CLS: RANDOMIZE TIMER
N = 100
DIM S(N)
' Формирование массива и вывод его на экран
FOR I = 1 TO N
S(I) = INT(RND(1) * 21) + 2
```

```
? S(I);
NEXT I
?
SUM = 0
FOR I = 1 TO INT(SQR(N))-1
SUM = SUM + S(I^2)
NEXT I
? "Сумма заданных элементов массива равна "; sum
END
```

296.

```
' Вывод массива в строки по 6 элементов, начиная с последнего
CLS: RANDOMIZE TIMER
INPUT "Сколько чисел будет в массиве"; N
DIM S(N)
' Формирование массива и вывод его на экран
FOR I = 1 TO N
S(I) = INT(RND(1) * 90) + 10
? S(I);
NEXT I
?: ?
' Вывод массива в строки по 6 элементов, начиная с последнего
I = N
WHILE I <> 6
FOR I = N TO N - 5 STEP -1
? S(I);
NEXT I
N = N - 6
?
WEND
END
```

299.

```
' Столбиковая интерпретация массива, его максимума и минимума
CLS : RANDOMIZE TIMER
' Объявление и формирование массива
INPUT "Сколько чисел будет в массиве"; N
DIM S(N)
FOR I = 1 TO N
S(I) = INT(RND(1) * 150) + 50
NEXT I
```

```
'Поиск максимального элемента массива и его индекса
MAX = S(1): NMAX = 1
FOR I = 2 TO N
IF S(I) > MAX THEN MAX = S(I): NMAX = I
NEXT I
'Поиск минимального элемента массива и его индекса
MIN = S(1): NMIN = 1
FOR I = 2 TO N
IF S(I) < MIN THEN MIN = S(I): NMIN = I
NEXT I
' Графическая интерпретация массива
SCREEN 9
X = 100
FOR I = 1 TO 10
LINE (X, 300)-(X + 10, 300 - S(I)), 14, BF
IF I = NMAX THEN LINE (X, 300)-(X + 10, 300 - S(I)), 4, BF
IF I = NMIN THEN LINE (X, 300)-(X + 10, 300 - S(I)), 2, BF
LINE (X, 300)-(X + 10, 300 - S(I)), 15, B
X = X + 10
NEXT I
' Вывод на экран элементов массива, его максимума и минимума
FOR I = 1 TO N
PRINT S(I);
NEXT I
PRINT
PRINT "Максимальный элемент массива - "; MAX; "Его индекс - ";
NMAX
PRINT "Минимальный элемент массива - "; MIN; "Его индекс - "; NMIN
END
```

301.

```
' Формула Герона
DATA 13, 48, 35, 90, 62
DATA 35, 40, 20, 71, 55
DATA 27, 68, 41, 54, 29
CLS
DIM A(5), B(5), C(5), P(5), S(5)
FOR I = 1 TO 5: READ A(I): NEXT I
FOR I = 1 TO 5: READ B(I): NEXT I
```

```
FOR I = 1 TO 5: READ C(I): NEXT I
FOR I = 1 TO 5
P(I) = A(I) + B(I) + C(I)
PP = P(I)/2
S(I) = SQR (PP *(PP - A(I)) * (PP - B(I)) * (PP - C(I)))
? "Периметр"; I; "треугольника равен"; P(I); "см"
? "Площадь"; I; "треугольника равна"; S(I); "кв.см"
?
NEXT I
END
```

306.

```
' Расстояния между точками
DATA 120, 58, 280, 440, 157, 99, 350, 290, 500,159
DATA 271, 279, 35, 100, 160, 234, 45, 170, 200, 66
DATA 271, 279, 35, 100, 360, 234, 545, 170, 200, 266
DATA 120, 58, 280, 240, 157, 99, 150, 290, 50, 159
SCREEN 9
' Объявление и формирование массивов
DIM X1(10), Y1(10), X2(10), Y2(10)
FOR I = 1 TO 10: READ X1(I): NEXT I
FOR I = 1 TO 10: READ Y1(I): NEXT I
FOR I = 1 TO 10: READ X2(I): NEXT I
FOR I = 1 TO 10: READ Y2(I): NEXT I
' Формирование массива расстояний между точками
FOR I = 1 TO 10
S(I) = SQR((X1(I) - X2(I)) ^ 2 + (Y1(I) - Y2(I)) ^ 2)
NEXT I
' Поиск максимальной и минимальной длины
MAX = S(1): NMAX = 1
FOR I = 2 TO 10
IF S(I) > MAX THEN MAX = S(I): NMAX = I
NEXT I
MIN = S(1): NMIN = 1
FOR I = 2 TO 10
IF S(I) < MIN THEN MIN = S(I): NMIN = I
NEXT I
' Графическая интерпретация
FOR I = 1 TO 10
LINE (X1(I), Y1(I))-(X2(I), Y2(I)), 15
```

```
IF I = NMAX THEN LINE (X1(I), Y1(I))-(X2(I), Y2(I)), 4
IF I = NMIN THEN LINE (X1(I), Y1(I))-(X2(I), Y2(I)), 2
NEXT I
END
```

311.

```
' Сборщики компьютеров
'Понедельник
DATA 37, 14, 48, 24, 80, 60, 56
'Вторник
DATA 77, 34, 45, 23, 45, 39, 51
'Среда
DATA 58, 65, 49, 49, 56, 45, 38
'Четверг
DATA 61, 57, 55, 89, 33, 52, 60
'Пятница
DATA 80, 67, 77, 70, 55, 76, 81
CLS
' Объявление и формирование массивов для каждого сборщика
DIM A(5), B(5), C(5), D(5), E(5), F(5), G(5), S(7), SR(5)
FOR I = 1 TO 5
READ A(I), B(I), C(I), D(I), E(I), F(I), G(I)
NEXT I
' Общее количество компьютеров за неделю для каждого сборщика
FOR J = 1 TO 7: S(J) = 0: NEXT J
J = 1: FOR I = 1 TO 5: S(J) = S(J) + A(I): NEXT I
J = 2: FOR I = 1 TO 5: S(J) = S(J) + B(I): NEXT I
J = 3: FOR I = 1 TO 5: S(J) = S(J) + C(I): NEXT I
J = 4: FOR I = 1 TO 5: S(J) = S(J) + D(I): NEXT I
J = 5: FOR I = 1 TO 5: S(J) = S(J) + E(I): NEXT I
J = 6: FOR I = 1 TO 5: S(J) = S(J) + F(I): NEXT I
J = 7: FOR I = 1 TO 5: S(J) = S(J) + G(I): NEXT I
' Максимальное количество за неделю одним сборщиком
MAX = S(1): K = 1
FOR J = 2 TO 7
IF S(J) > MAX THEN MAX = S(J): K = J
NEXT J
PRINT "Максимальное количество компьютеров за неделю - "; MAX
PRINT "собрал рабочий №"; K
PRINT
```

```
' Среднее количество за каждый день недели
FOR I = 1 TO 5
SR(I) = (A(I) + B(I) + C(I) + D(I) + E(I) + F(I) + G(I)) / 7
PRINT "В"; I; "день в среднем собрано"; SR(I); "компьютеров"
NEXT I
PRINT
' Лучший результат за один день, номер сборщика и день недели
RESTORE
READ DAY
MX = DAY: K = 1
FOR I = 2 TO 35
READ DAY
IF DAY > MX THEN MX = DAY: K = I
NEXT I
PRINT "Максимальное количество компьютеров за один день -"; MX
N = K MOD 7
N1 = K / 7
IF N1 <= 1 AND N1 > 0 THEN D$ = "Понедельник"
IF N1 <= 2 AND N1 > 1 THEN D$ = "Вторник"
IF N1 <= 3 AND N1 > 2 THEN D$ = "Среда"
IF N1 <= 4 AND N1 > 3 THEN D$ = "Четверг"
IF N1 <= 5 AND N1 > 4 THEN D$ = "Пятница"
PRINT "собрал рабочий №"; N
PRINT "Это был день недели "; D$
END
```

Простейшие сортировки

325.

```
' Сортировка выбором
CLS : RANDOMIZE TIMER
INPUT "Сколько чисел будет в массиве"; N
' Объявление, формирование массива и вывод его на экран
DIM S(N)
FOR I = 1 TO N
S(I) = INT(RND(1) * 150) + 50
PRINT S(I);
NEXT I
PRINT
```

```
' Сортировка с использованием вложенных циклов
FOR I = 1 TO N - 1
MIN = S(N)
K = N
FOR J = N TO I STEP -1
IF S(J) < MIN THEN MIN = S(J): K = J
NEXT J
SWAP S(K), S(I)
NEXT I
' Вывод отсортированного массива
FOR I = 1 TO N
PRINT S(I);
NEXT I
END
```

328.

```
' Сортировка методом "пузырька"
CLS
DATA Q,W,E,R,T,Y,U,I,O,P
DATA A,S,D,F,G,H,J,K,L,Z
DATA X,C,V,B,N,M
' Объявление, формирование и вывод исходного массива на экран
N = 26: DIM ENG$(N)
PRINT "Исходный массив"
FOR I = 1 TO N
READ ENG$(I)
PRINT ENG$(I); " ";
NEXT I
PRINT
FOR I = 1 TO N - 1
FOR J = 1 TO N - 1
IF ENG$(J) > ENG$(J + 1) THEN SWAP ENG$(J), ENG$(J + 1)
NEXT J
NEXT I
PRINT
' Вывод отсортированного массива
PRINT "Отсортированный массив"
FOR I = 1 TO N
PRINT ENG$(I); " ";
NEXT I
END
```

Глава 6

Тест 2

1б, 2б, 3в, 4б, 5в, 6в, 7б

Практикум 7

Таблица 7.5. *К 300-летию Санкт-Петербурга*

№	Вопрос	Ответ
1.	На какой широте расположен Санкт-Петербург?	59°57'
2.	Каков номер малой планеты "Ленинград"?	2046
3.	В Ленинградском зоопарке впервые в России и СССР родился жираф. В каком году это произошло?	1959
4.	В Царском селе в парке у пирамидальной беседки захоронены три любимые собаки императрицы Екатерины II. Перечислите их клички	Земира, Дюшес, Сир Том-Андерсон
5.	Какого числа был проведен первый в СПб. футбольный матч?	25 октября 1897 года
6.	Какова высота самого высокого сооружения СПб.?	316 м 27 см (телебашня)
7.	Какой мост в Санкт-Петербурге первым построен из чугуна?	Полицейский (ныне Народный), в 1806 году
8.	Больше всего львов в СПб. у бывшей дачи Кушелева-Безбородко. Сколько их там?	29
9.	Самая длинная улица СПб.?	Садовая
10.	Кто забил голы в решающем матче "Зенит"—"Металлист" 21 ноября 1984 года, после которого "Зенит" стал чемпионом СССР?	
11.	По чьему проекту было построено старейшее в городе лечебное учреждение общего пользования?	Дж. Кваренги (Мариинская больница)
12.	Сколько свай было забито в грунт для постройки Исаакиевского собора?	10 762
13.	Какая школа (по номеру) в плане представляет собой стилизованные серп и молот?	№ 384
14.	Самый большой в мировой практике вес разводного крыла – 3225 тонн имеет один из мостов СПб. Какой?	Литейный

Таблица 7.5 *(окончание)*

№	Вопрос	Ответ
15.	Сколько фонтанов входит в Большой Каскад Нижнего парка Петергофа?	64
16.	В саду дома № 12 по улице Академика Павлова сооружен в 1935 году памятник собаке как дань бескорыстному служению собак науке. Кто автор проекта памятника?	Скульптор И. Ф. Безпалов
17.	Из всех городов мира наибольшим числом конных памятников историческим лицам обладает Петербург. Сколько их и каким императорам?	4 памятника: 2 — Петру I, 1 — Николаю I и 1 — Александру III
18.	Каково современное название Кунсткамеры?	Музей антропологии и этнографии им. Петра Великого Российской академии наук
19.	Первый русский журнал для детей вышел в 1785 году. Как он назывался?	"Детское чтение для сердца и разума"
20.	Старейший из вокзалов Петербурга — Московский. Кто был первым начальником его?	Инженер-капитан Н. И. Миклухо-Маклай
21.	12 мая 1911 года состоялся первый выпуск Гатчинской школы авиации. Сколько летчиков было выпущено и их фамилии?	7. Бродович, Вегенер, Головин, Дацкевич, Данилевский, Карпов, Фирсов

Практикум 8

Близ {*название созвездия* — Ориона}, над горизонтом расположен Эридан {*латинское название* — Eridanus}, цепочка звезд которого напоминает реку, текущую от {*часть тела* — ног} {*имя* — Ориона } (ее и называли Рекой — Потамос). По мифу, {*имя* — Фаэтон}, сын Солнца, назывался сперва Эриданом, он же дал свое имя реке {*название реки* — По (Падус), в которую упал. Это длинное созвездие оканчивается звездой {*название звезды* — Ахернар}, (по-арабски: ахир ан-нахр — {*перевод названия* — конец реки}) — {*которой?* — девятой} по яркости звездой неба (звездная величина {*число* — 0,46}, расстояние от Солнца — 30 {*чего?* — парсеков}).

{*Специалисты* Астрологи} высоко оценивают эту звезду. {*Фамилия* — Девор} называет ее "Ангел с мечом", а {*имя* — Павел} Глоба — звездой высшего по-

кровительства ("{*образное выражение* — огненный меч}" защиты). Эбертин и {*фамилия* — Хофман} находят у нее, помимо {*какого?* — юпитерианского} влияния, примесь энергий Марса и Урана. Интуитивно она привлекает и писателей: именно так — "{*греческая буква* — Альфа}* Эридана" — был назван известный рассказ {*имя* — Александра} {*фамилия* — Колпакова}, давший название одноименному сборнику фантастики издательства "{*Название* — Молодая гвардия}" ({*год* — 1960} г.).

В древности она не была известна (ибо не видна в Европе и на Ближнем Востоке), поэтому последней звездой считали другую, менее яркую звезду того же созвездия — {*название звезды* — Акамар} ({*греческая буква* — тэта}* Эридана). Именно смешение этих звезд порождает ошибки в псевдоисторических трудах {*имя* — Анатолия} Фоменко.

Многие другие звезды этого созвездия также интересны, например, {*греческая буква* — бета}* Эридана Курса, Омикрон-1 Эридана {*название звезды* — Беид}, Эта Эридана {*название звезды* — Аза} и {*греческая буква* — Ипсилон}* (Тимим). Кстати, обозначать звезды буквами греческого алфавита предложил в своем атласе "{*Название* — Уранометрия}" {*национальность* — немецкий} астроном {*имя* — Иоганн} Байер в {*год* — 1603} г.; но букв всего {*число* — 24}, так что применяются дополнительные индексы — в том же Эридане целых 9 звезд {*греческая буква* — Тау }*!

Но наиболее интересна практически звезда {*греческая буква* — Эпсилон}* Эридана — это самая близкая из обнаруженных до сих пор звезд с планетами, она находится на расстоянии всего 10,5 {*чего?* — световых лет}. Планета была обнаружена группой {*имя* — Уильяма {*фамилия* — Кочрана} в обсерватории {*какого?* — Техасского} университета.

Тест 3

1а, 2г, 3а, 4в, 5г, 6а, 7г, 8б, 9г, 10б

Тест 4

1б, 2б, 3в, 4в, 5б, 6в, 7б, 8б, 9в, 10б, 11б, 12в, 13б, 14в, 15в, 16б

Заключение

После всего вышесказанного хочется еще раз сказать большое спасибо всем авторам перечитанных автором книг по информатике и просмотренных сайтов в Интернете, его любимым учителям и ученикам!

Эта книга не претендует на академичность, это не догма и даже не руководство к действию. Автор просто будет рад, если хоть кому-то в чем-то она помогла, а также рад любым замечаниям.

Кроме того, он готов ответить на все вопросы. И еще раз приводит свой E-mail: **old_matros@mail.ru**. (old_matros — через символ подчеркивания).

Всем удачи, счастья и здоровья!

Список дополнительной литературы

1. Персональный компьютер в играх и задачах / Под ред. Н. Н. Моисеева. — М.: Наука, 1988. — 191 с.
2. Элементы информатики / Под ред. Л. Н. Королева. — М.: Наука, 1988. — 318 с.
3. Абрамов С. А., Зима Е. В. Начала информатики. М.: Наука, 1989. — 256 с.
4. Задачи и упражнения по программированию / Под ред. А. Я. Савельева. — М.: Высшая школа, 1989. — 110 с.
5. Пильщиков В. Н. Сборник задач по языку Паскаль. — М.: Наука, 1989. — 154 с.
6. Арсак Ж. Программирование игр и головоломок. — М.: Наука, 1990. — 221 с.
7. Кушниренко А. Г. Основы информатики и вычислительной техники. — М.: Просвещение, 1990. — 223 с.
8. Пярнпуу А. А. Программирование на современных алгоритмических языках. — М.: Наука, 1990. — 380 с.
9. Чернов Б. И. Программирование на алгоритмических языках Бейсик, Фортран, Паскаль. — М.: Просвещение, 1991. — 190 с.
10. Моррилл Г. Бейсик для ПК ИБМ. — М.: Финансы и статистика, 1993. — 207 с.
11. Практикум по программированию / Под ред. А. Я. Савельева. — М.: Высшая школа, 1993. — 205 с.
12. Сборник задач по базовой компьютерной подготовке / Под общ. ред. И. Н. Котаровой. — М.: МЭИ, 1998. — 177 с.
13. Горячев А., Шафрин Ю. Практикум по информационным технологиям. — М.: Лаборатория базовых знаний, 1999. — 272 с.
14. Гусева В. А. Учимся информатике: задачи и методы их решения. — М.: ДИАЛОГ-МИФИ, 1999. — 320 с.

15. Задачник-практикум (в 2-х томах) / Под ред. И. Семакина, Е. Хеннера. — М.: Лаборатория базовых знаний, 1999. — 304 с.
16. Радченко Н. П., Козлов О. А. Школьная информатика. Экзаменационные вопросы и ответы. — М.: Финансы и статистика, 1999. — 160 с.
17. Филичев С. Ф. Информатика — это просто! — М.: ЭКОМ, 1999. — 343 с.
18. В. Байков. Интернет. Поиск информации. Продвижение сайтов. — СПб.: БХВ — Санкт-Петербург, 2000. — 288 с.
19. Мельникова О. И., Бонюшкина А. Ю. Начала программирования на языке QBasic. — М.: ЭКОМ, 2000. — 303 с.
20. Симонович С., Евсеев Г. Занимательный компьютер. — М.: АСТ-Пресс, 2001. — 368 с.
21. Тимофеев Г., Тимофеева Е. Графический дизайн. — Ростов н/Д: Феникс, 2002. — 320 с.