

**Александр Дуванов  
Алексей Рудь  
Виктор Семенко**

# **АЗЫ ПРОГРАММИРОВАНИЯ**

**ФАКУЛЬТАТИВНЫЙ КУРС**

**ЗАДАЧНИК**

Санкт-Петербург

«БХВ-Петербург»

2005

УДК 681.3.06(076.1)  
ББК 32.973я721  
Д79

**Дуванов А. А., Рудь А. В., Семенко В. П.**

Д79 Азы программирования. Факультативный курс. Задачник. — СПб.: БХВ-Петербург, 2005. — 160 с.: ил.

ISBN 5-94157-288-3

В книге собраны задачи, которые предлагались на сетевых турнирах курса «Азы программирования» Роботландского университета. Представлены задачи различной тематики: поиск и сортировка, нахождение экстремума, разнообразные подсчёты, логические и геометрические задачи; традиционный наибольший общий делитель и Ханойская башня; нетрадиционные для школьников рекурсивные определения Бэкуса-Наура, запись выражения по-польски, стековые вычисления, синтаксический анализ и построение трансляторов. Задачи ориентированы на программирование в среде исполнителей Кукарача и Корректор, подробное описание которых содержит книга «Азы программирования. Книга для ученика».

*Для учащихся средних образовательных учреждений*

УДК 681.3.06(076.1)  
ББК 32.973я721

### **Группа подготовки издания:**

Главный редактор	<i>Екатерина Кондукова</i>
Зам. гл. редактора	<i>Людмила Еремеевская</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Елена Михальчук</i>
Компьютерная верстка	<i>Натальи Караваевой</i>
Корректор	<i>Виктория Пиотровская</i>
Дизайн обложки	<i>Инны Тачиной</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 27.06.05.

Формат 70×100<sup>1/16</sup>. Печать офсетная. Усл. печ. л. 10.

Тираж 3000 экз. Заказ №

“БХВ-Петербург”, 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию № 77.99.02.953.Д.006421.11.04 от 11.11.2004 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов  
в ГУП “Типография “Наука”  
199034, Санкт-Петербург, 9 линия, 12

ISBN 5-94157-288-3

© Дуванов А. А., Рудь А. В., Семенко В. П., 2005  
© Дуванов А. А., Русс А. А., иллюстрации, 2005  
© Оформление, издательство “БХВ-Петербург”, 2005

# Оглавление

<b>Состав комплекта .....</b>	<b>7</b>
<b>Вступление .....</b>	<b>9</b>
Исполнители и язык программирования .....	9
Исполнитель Кукарача.....	9
Исполнитель Корректор .....	11
<b>Организация обучения на курсе «Азы программирования» .....</b>	<b>15</b>
Сетевые турниры .....	15
Решение задач .....	16
Перекрестная проверка работ .....	16
Общие рекомендации по перекрёстной проверке .....	17
Этическая дисциплина проверки и её полнота .....	17
Общий критерий оценок .....	18
Апелляционный период .....	18
Подведение итогов.....	18
Говорят студенты курса «Азы программирования» .....	19
Маги и Магистры программирования .....	20
Список Бессмертных Магов (по состоянию на 2004 год) .....	20
История Кукарачи .....	21
История Корректора .....	22
<b>Глава 1. Олимпиада 1996/1997.....</b>	<b>23</b>
Турнир Кукарачи.....	23

<b>Глава 2. Турниры 1997/1998 .....</b>	<b>31</b>
2.1. Турнир Кукарачи .....	31
2.2. Турнир Корректора .....	38
<b>Глава 3. Турниры 1998/1999 .....</b>	<b>45</b>
3.1. Турнир Кукарачи .....	45
Пояснительная записка .....	45
Разминка .....	46
Синтаксический анализ записи .....	48
Кукарачья геометрия .....	51
3.2. Турнир Корректора .....	56
Синтаксический анализ выражений и трансляторы .....	56
Геометрия на ленте Корректора .....	61
<b>Глава 4. Турниры 1999/2000 .....</b>	<b>67</b>
4.1. Турнир Кукарачи .....	67
Разминка .....	67
Синтаксический анализ записи .....	70
Кукарачья геометрия .....	75
4.2. Турнир Корректора .....	78
Система команд Малыша .....	82
Алгоритм выполнения программы .....	83
Авост .....	83
Задания .....	84
<b>Глава 5. Турниры 2000/2001 .....</b>	<b>85</b>
5.1. Турнир Кукарачи .....	85
Построения на клетчатом поле .....	85
Анализаторы и трансляторы .....	90
5.2. Турнир Корректора .....	92
Соглашения .....	92
Рекуррентные последовательности .....	93
Лексический анализ выражений .....	96
Классические алгоритмы .....	96

---

<b>Глава 6. Турниры 2001/2002 .....</b>	<b>99</b>
6.1. Турнир Кукарачи .....	99
Вступительное слово куратора .....	99
6.2. Турнир Корректора .....	106
Вступительное слово куратора .....	106
<b>Глава 7. Турниры 2002/2003 .....</b>	<b>113</b>
7.1. Турнир Кукарачи .....	113
7.2. Турнир Корректора .....	117
Вступительное слово куратора .....	117
<b>Глава 8. Турниры 2003/2004 .....</b>	<b>125</b>
8.1. Турнир Кукарачи .....	125
8.2. Турнир Корректора .....	131
Определения .....	131
<b>Глава 9. Турниры 2004/2005 .....</b>	<b>139</b>
Олимпиада Кукарачи .....	139
<b>Ссылки на задачи.....</b>	<b>145</b>
Часть I. Кукарача.....	145
Часть II. Корректор.....	151
Часть III. Транслятор?.. Это очень просто .....	159

# Состав комплекта

Комплект «Азы программирования» содержит всё необходимое для построения факультатива, сопровождающего школьный курс «Азы информатики». Он включает в себя:

- книгу «Азы программирования. Магия для начинающих» — учебник;
- книгу «Азы программирования. Задачи роботландских турниров» — задачник;
- книгу «Азы программирования» — пособие для учителя;
- CD с программными средами и дополнительными материалами (сопровождает книгу учителя).

# Вступление

В этой книге собраны задачи, которые предлагались на сетевых турнирах курса «Азы программирования» Роботландского университета в течение девяти учебных лет, начиная с первой общей олимпиады 1996/1997 и далее до олимпиады первого семестра 2004/2005 учебного года.

Задачи ориентированы на программирование исполнителей Кукарача и Корректор.

Подробное описание этих исполнителей и языка программирования содержит книга «Азы программирования. Магия для начинающих». Далее приводится лишь краткое, справочное описание Кукарачи, Корректора и языка программирования этих исполнителей.

## Исполнители и язык программирования

### Исполнитель Кукарача

Кукарача — исполнитель, работающий на клетчатом поле (рис. 1).



Рис. 1. Клетчатое поле с исполнителем и кубиками

Кукарача может переползать из клетки в клетку вверх, вниз, влево, вправо, но не по диагонали. Выход исполнителя за пределы поля запрещён (отказ «Не могу!»). В каждой клетке может находиться кубик с нанесённым на его грань символом. Кукарача, перемещаясь по полю, может толкать один или несколько кубиков перед собой и «сбрасывать» кубики за пределы поля.

Некоторые кубики на поле могут иметь «скрытые» символы, обозначаемые знаком вопроса. Считается, что такие кубики расположены в клетке «символом вниз». Толкая скрытый кубик, Кукарача перемещает его в следующую клетку по ходу движения, переворачивает и видит надпись.

Исполнитель в состоянии прочитать символ на кубике независимо от того, была ли запись символа обычной или скрытой.

Система команд исполнителя (СКИ) включает пять команд (рис. 2, табл. 1).



Рис. 2. Система команд Кукарачи

Таблица 1

Команда Кукарачи	Как выполняется
ВПРАВО	Сместиться на одну клетку вправо
ВЛЕВО	Сместиться на одну клетку влево
ВВЕРХ	Сместиться на одну клетку вверх
ВНИЗ	Сместиться на одну клетку вниз
СТОЯТЬ	Пустая команда — исполнитель не выполняет никаких действий

Любая команда, кроме команды **стоять**, может привести к отказу «Не могу», если её выполнение уводит Кукарачу за пределы поля.

Кукарача способен выполнять в своей среде проверки, представленные в табл. 2 (запись <символ> обозначает шаблон, на месте которого может быть любой символ).

Таблица 2

Запись условия	Результат проверки
<символ>	<i>Истина</i> , если Кукарача толкнул кубик с указанным в условии символом, <i>ложь</i> в противном случае
НЕ <символ>	<i>Ложь</i> , если Кукарача толкнул кубик с указанным в условии символом, <i>истина</i> в противном случае
ПУСТО	<i>Истина</i> , если клетка, в которую сместился Кукарача, пуста, <i>ложь</i> в противном случае
НЕ ПУСТО	<i>Ложь</i> , если клетка, в которую сместился Кукарача, пуста, <i>истина</i> в противном случае
ЦИФРА	<i>Истина</i> , если Кукарача толкнул кубик с цифрой (один из символов 0, 1, 2, 3, 4, 5, 6, 7, 8, 9), <i>ложь</i> в противном случае
НЕ ЦИФРА	<i>Ложь</i> , если Кукарача толкнул кубик с цифрой (один из символов 0, 1, 2, 3, 4, 5, 6, 7, 8, 9), <i>истина</i> в противном случае

## Исполнитель Корректор

Корректор работает с длинной лентой, которая разбита на клетки (ячейки).

В ячейку может быть записан один символ. Символ — это буква, цифра, другие знаки, составляющие фиксированный алфавит исполнителя.

Считается, что лента спрятана в непрозрачный для Корректора футляр. Исполнитель видит только одну ячейку через специальное окно в футляре, и может записывать в «оконную» ячейку (окно ленты) любые символы из своего алфавита. Корректор может перемещать окно вправо и влево вдоль ленты и, таким образом, записывать символы в любые ячейки на ней.

В среде исполнителя есть ящик — специальная ячейка памяти. В ящик Корректор может снимать копию символа из окна и, наоборот, копировать символ из ящика в окно на ленту (рис. 3).



Рис. 3. Среда Корректора

На рис. 4 и в табл. 3 приводится полный набор команд Корректора и его алфавит.



Рис. 4. Система команд Корректора

Таблица 3

Команда Корректора	Как выполняется
ВПРАВО	Переместить окно на одну клетку вправо
ВЛЕВО	Переместить окно на одну клетку влево
ПИШИ <символ>	Записать указанный символ в клетку на ленте. В качестве символа можно указывать ключевые слова ПУСТО и ПРОБЕЛ
ЯЩИК+	Копировать символ с ленты в ящик
ЯЩИК-	Копировать символ из ящика на ленту
ОБМЕН	Поменять местами содержимое ящика и окна ленты
ПЛЮС	Заменить символ в окне символом, следующим по порядку в алфавите Корректора. Команда приводит к отказу (ситуация «Не могу»), когда в окне перед её выполнением записан последний символ алфавита
МИНУС	Заменить символ в окне символом, предыдущим по порядку в алфавите Корректора. Команда приводит к отказу (ситуация «Не могу»), когда в окне записан первый символ алфавита (специальный символ ПУСТО)
СТОЯТЬ	Пустая команда; её выполнение не вызывает никаких изменений в среде Корректора

Корректор способен выполнять в своей среде следующие проверки (табл. 4).

Таблица 4

Запись условия	Результат проверки
<символ>	<i>Истина</i> , если символ в окне совпадает с символом, указанным в условии, <i>ложь</i> в противном случае. В качестве символа можно указывать ключевые слова ПУСТО, ПРОБЕЛ и ЦИФРА
Я=Л	<i>Истина</i> , если символ в ящике совпадает с символом в окне, <i>ложь</i> в противном случае
Я#Л	<i>Истина</i> , если символ в ящике не совпадает с символом в окне, <i>ложь</i> в противном случае
Я>Л	<i>Истина</i> , если символ в ящике имеет больший порядковый номер в алфавите Корректора по сравнению с номером символа в окне, <i>ложь</i> в противном случае
Я<Л	<i>Истина</i> , если символ в ящике имеет меньший порядковый номер в алфавите Корректора по сравнению с номером символа в окне, <i>ложь</i> в противном случае

Перед любым условием может быть написан ключевой модификатор **НЕ**, который меняет смысл условия на противоположный.

## Язык программирования исполнителей

### Разделитель слов

Разделителем слов в языке служит один или несколько пробелов, а также конец строки.

### Программа

Программа представляет собой последовательность процедур, записанных в любом порядке. Имя процедуры может быть использовано наравне с командой из СКИ исполнителя.

### Процедура

Описание процедуры имеет вид, представленный на рис. 5.

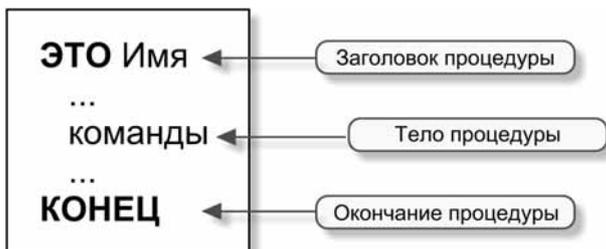


Рис. 5. Описание процедуры

## Имя процедуры

Имя процедуры должно начинаться с буквы. Оно не должно содержать пробелов и не может совпадать с ключевыми словами языка программирования.

## Комментарии

Комментарий имеет вид:

// текст комментария (до конца строки)

Комментарии могут располагаться как в процедурах, так и между ними.

## Команды

Таблица 5

Команда языка	Как выполняется
<команда из СКИ>	<b>Команда исполнителя</b> Выполнение определяется описанием команды в СКИ
<имя процедуры>	<b>Вызов процедуры</b> Выполняется процедура с указанным именем
ПОВТОРИ <число> <команда>	<b>Цикл ПОВТОРИ</b> Повторение выполнения команды указанное число раз
ПОКА <условие> <команда>	<b>Цикл ПОКА</b> Повторение выполнения команды, пока условие имеет значение <i>истина</i> . Проверка условия — перед выполнением команды
ЕСЛИ <условие> ТО <команда1> ИНАЧЕ <команда2>	<b>Команда ветвления</b> Проверяется условие и выполняется либо <команда1> (при значении условия: <i>истина</i> ), либо <команда2> (при значении условия: <i>ложь</i> ). Часть «ИНАЧЕ <команда2>» может быть опущена
{ <список команд> }	<b>Составная команда</b> Объединение нескольких команд в одну

# Организация обучения на курсе «Азы программирования»

Роботландский университет — это сетевая школа, в которой обучаются учителя и школьники индивидуально или совместно. Руководитель Роботландского университета — Дуванов Александр Александрович ([kurs@robotland.pereslavl.ru](mailto:kurs@robotland.pereslavl.ru)).

Во главе курса стоит куратор — сотрудник Роботландского университета.

В разное время кураторами «Азов программирования» были:

- Дуванов Александр Александрович, Переславль-Залесский (1996/1997 — 2000/2001);
- Рудь Алексей Владиславович, Снежинск (2001/2002, 2002/2003);
- Семенко Виктор Петрович, Рубцовск (2003/2004, 2004/2005).

Студенты курса образуют команды — группы детей, возглавляемые руководителем (как правило, школьным учителем).

В начале учебного года команды получают от куратора необходимые программно-методические материалы, календарные, тематические планы и приступают к работе.

## Сетевые турниры

В течение учебного года на курсе организуются два турнира — первый в конце первого семестра, второй — в конце второго.

Турнир состоит из четырёх этапов:

- решение задач;
- перекрёстная проверка;
- апелляционный период;
- подведение итогов.

## Решение задач

Условия задач (с оценочными баллами) публикуются на курсовой электронной конференции, и команды приступают к работе, на которую отводится две недели.

Задачи могут решаться коллективно, группами или индивидуально — турнирная тактика определяется командой (куратором никак не регламентируется).

В качестве результата команда высылает куратору одну общую работу, с обязательным указанием авторов решения каждой задачи.

## Перекрестная проверка работ

Проверка работ выполняется за одну неделю.

Все решения, присылаемые на конкурс, проверяются и рецензируются куратором, но кроме этого, сами ребята проверяют работы друг друга (по принципу команда из города А проверяет работы команд из городов В, С и D).

Перекрёстная проверка работ продолжает этап обучения. Она позволяет:

- Провести практические занятия с детьми по очень важному и ответственному этапу создания программного продукта — тестированию.

Ни одна компьютерная фирма не выпустит в свет свой программный продукт без тщательной и всесторонней проверки. Время, затраченное на тестирование продукта, заранее предусматривается, включается в план разработки и по количеству человеко-часов часто сравнимо со временем создания продукта. Тестируют программы специальные работники фирмы, как правило, не авторы программы. Как показывает опыт, автор программы не способен выявить в своём продукте большинство допущенных им ошибок. Объясняется это просто. Автор заставляет программу работать правильно и на это направляет свои усилия. Человек-тестер, наоборот, прилагает усилия для того, чтобы заставить программу работать неверно, «подсовывает» ей такие входные данные, на которых она начинает «хромать» — выдавать неверные результаты. Иными словами, у автора программы и проверяющего разные психологические установки: у одного — заставить программу работать, у другого — «сломать» её.

- Научить юных программистов читать чужую программу и понимать по её исходному тексту мысль автора.
- Познакомиться с решением других ребят и сравнить их со своими.

Перекрёстная проверка снабжается следующей инструкцией.

## Общие рекомендации по перекрёстной проверке

1. *Тестирование программы.* Первой, обязательной частью контроля, является проверка работоспособности программы на компьютере. Тестирование не должно сводиться к одному запуску программы с тем состоянием среды, которое предложено автором. Например, для Кукарачи запускайте программу с различными допустимыми (по условию задачи) положениями кубиков и исполнителя на поле. Обязательно рассмотрите критические (крайние) случаи и убедитесь, что программа обрабатывает их правильно.
2. *Проверка качества алгоритма.* Нужно внимательно прочитать объяснение алгоритма, приведённое автором, и проверить соответствие его программе при помощи запуска «по шагам». Если вы заметите в алгоритме погрешность, которая может привести к неверной работе программы, то придумайте тест (допустимое состояние среды), который обрабатывается программой неверно. Этот тест послужит доказательством наличия ошибки в программе.
3. *Оценка эффективности алгоритма и программы.* Оценивайте эффективность по трём признакам:
  - универсальность алгоритма (алгоритмы, которые работают не для всех возможных начальных состояний среды — ошибочные);
  - число строк программы с кодом (чем меньше, тем лучше);
  - время работы программы (чем меньше «движений» выполняет исполнитель в среде, тем лучше).
4. *Качество описаний алгоритма и внешний вид (рисунок) программы.* Изучая алгоритм автора, обратите внимание на качество описания, доступность и точность изложения, грамматическую правильность текста. Рассматривая программу, обращайтесь внимание на форматирование кода и оформление: комментарии, смысловые имена процедур, верно расставленные отступы и пустые строки улучшают внешний вид и способствуют лучшему пониманию программы.

## Этическая дисциплина проверки и её полнота

В инструкциях к перекрёстным проверкам куратор рекомендует командам помимо формальной оценки не скупиться на развернутые замечания и особенно на заслуженные комплименты.

Все замечания должны быть высказаны дружеским тоном, ведь оценивается не автор, а авторская работа. Работа может быть очень слабой (это разрешается!), но сам автор может быть очень талантливым, и его главный труд ещё впереди. Не надо считать недостатки работы недостатками автора.

Все оценки должны быть в обязательном порядке обоснованы (единым для всех сводом опубликованных критериев).

## Общий критерий оценок

1. Если задача не решалась командой, то за неё выставляется 0 баллов.
2. Если задача имеет несколько вариантов решения, то выставляется единственная оценка (балл) по лучшему варианту.
3. Программа не работает: выдается синтаксическая ошибка или сообщение «Не могу». Может, это недоразумение? Попробуйте обнаружить и исправить допущенную автором опisku.
  - 3.1. Описка. Если исправление помогло и программа заработала, снизьте оценку по рекомендациям для каждой отдельной задачи.
  - 3.2. Неверное решение. Если ошибки серьёзные и связаны с неверным алгоритмом или неверной программой, выставляйте за задачу 1 балл.
4. Программа работает, но в ней обнаружены грубые ошибки (например, программа работает только для того состояния среды, которое предложил автор) — выставляйте 2 балла.
5. Программа работает, но в ней есть ошибки (обнаружены состояния среды, в которых программа работает неправильно). Ставьте оценку в соответствии с приведёнными рекомендациями для каждой отдельной задачи.
6. В тех случаях, когда приведённые рекомендации оставляют неоднозначность, проверяющий принимает самостоятельное решение по выставлению оценки.

## Апелляционный период

Этот этап обучения длится одну неделю и совмещается с обсуждением решений, которое организует куратор.

Апелляции направляются на курсовую электронную конференцию и адресуются куратору или конкретным командам.

Студенты учатся вести дискуссию, публично отстаивая свою точку зрения. Ответ на апелляцию также имеет публичную форму.

В апелляции в корректной форме обосновывается претензия. Апелляции, основанные на эмоциях, не допускаются. Не допускаются упрёки к проверяющим, сравнение одних проверок с другими.

В апелляции указывается на конкретный пункт конкретной проверки и доказывается, что он противоречит конкурсным критериям оценок или что обозначенная в проверке ошибка отсутствует в работе.

## Подведение итогов

Куратор подводит итоги, публикуя в электронной конференции турнирную таблицу.

В этой таблице команды располагаются в порядке убывания итогового балла, вычисляемого как простая сумма набранных баллов по каждой задаче.

Балл за каждую задачу вычисляется по формуле:

$$\text{Балл} = ((\text{сумма } n \text{ оценок перекрестной проверки}) + 3 \cdot (\text{оценка куратора})) / (n + 3).$$

По результатам определяются победители: команды, занявшие I, II и III места.

В конце учебного года победителей ожидают:

- грамоты;
- дипломы Магов и Магистров;
- подарки от Роботландского университета.

## Говорят студенты курса «Азы программирования»

### Устич Дима

Мне понравился курс «Азы программирования». Сначала мы думали, что знаем всё о Кукараче, но оказалось, что знали совсем чуть-чуть. Но зато теперь мы можем решить задачу, используя рекурсивную пружину. А это совсем не просто. Но самое интересное, что нам было здорово вместе. Мы научились решать задачи сообща. Один предложит идею, а все вместе додумаем решение до конца.

### Галушко Марина

Мне понравилось дистанционное обучение. Но было очень трудно. Задачи были сложные, мы очень подробно их разбирали. Когда приходили конкурсные задачи, то забывали обо всём остальном. И решали, решали, решали. Дома, в школе, все вместе, по одному. Хотя и было трудно, но было интересно.

### Бердышева Юлия

Кажется, что эту задачу решить просто невозможно, а потом каждый говорит то, о чём думает, появляется интересная мысль и вырисовывается ход решения. Это очень здорово — решать трудные задачи вместе. А сколько у всех радости, когда задача уже решена, и Кукарача делает именно то, что мы хотим.

### Комлев Дима

Мне очень понравилось учиться в Роботландском университете, потому что мы не только научились решать задачи, но и увидели, как это делают

другие. Помню, нам очень понравились решения ребят из Снежинска. Мы очень тщательно тестировали их решения, а ошибок всё равно не было. Мы даже обнаружили ошибки в своих решениях, когда проверjali эту команду.

## Маги и Магистры программирования

**Маг** — это руководитель группы, которая когда-либо заняла в курсовом конкурсе первое место.

**Магистр** — это руководитель группы, которая в течение одного (любого) учебного года приняла участие в двух семестровых конкурсах и набрала на каждом не нулевую сумму баллов.

## Список Бессмертных Магов (по состоянию на 2004 год)

- Пинженина Софья Владимировна, Челябинск (Маг 1998/1999)
- Петров Владимир Анатольевич, Волгоград (Маг 1998/1999)
- Рудь Алексей Владиславович, Снежинск (Маг 1999/2000, 2000/2001)
- Семенова Наталья Евгеньевна, Тольятти (Маг 2001/2002)
- Есауленко Оксана Николаевна, Качканар (Маг 2001/2002)
- Вассина Роза Николаевна, Качканар (Маг 2001/2002)
- Пранис Сергей Александрович, Качканар (Маг 2001/2002)
- Семенко Виктор Петрович, Рубцовск (Маг 2002/2003)
- Шумилина Нина Дмитриевна, Тверь (Маг 2003/2004)
- Матыкин Вячеслав Юрьевич, Новохопёрск (Маг 2003/2004)

С 2003/2004 учебного года звание «Маг программирования» стало присуждаться и детям, которые показали особые успехи на роботландских сетевых турнирах.

Эти славные имена Магов от программирования навечно занесены в историю Роботландского университета.

- Мусин Марат Нариманович, Тверь, 9 класс, МОУ «Тверская гимназия № 6» (Маг 2003/2004)
- Громова Анна Валерьевна, Тверь, 9 класс, МОУ «Тверская гимназия № 6» (Маг 2003/2004)
- Погорелая Дарья Андреевна, Миасс, 6 класс, школа № 19 (Маг 2003/2004)
- Кротенко Владимир Николаевич, Алтайский край, г. Рубцовск, 8 класс, школа № 1 (Маг 2003/2004)

- **Югатов Владимир Анатольевич**, Алтайский край, г. Рубцовск, 8 класс, школа № 1 (Маг 2003/2004)
- **Носов Алексей Юрьевич**, Алтайский край, г. Рубцовск, 8 класс, школа № 1 (Маг 2003/2004)

## История Кукарачи

Исполнитель Кукарача (первое имя Таракан) родился на Дальнем Востоке в 1984 году. Роботландии тогда ещё не было, а был новосибирский Муравей, придуманный Г. А. Звенигородским вместе с группой исследователей (в неё входил и Ю. А. Первин, будущий директор Роботландии).

Второй участник будущей Роботландии, А. А. Дуванов в то время работал преподавателем в Благовещенском пединституте. Интерес благовещенцев к пионерским работам новосибирцев был велик. Настолько, что Дальний Восток приехал в Сибирь пообщаться с умными людьми, увидеть уникальную работу школьников на компьютерах своими глазами. По трагическому стечению обстоятельств как раз в это время от гриппа умер Г. А. Звенигородский. Ему было 32 года. Он даже не увидел издание своей книги, которая вышла уже после его смерти (Г. А. Звенигородский. Первые уроки программирования. Библиотечка Квант. Выпуск 41. Москва. Наука. 1985).

Разговор состоялся с Юрием Абрамовичем Первиным. Был показан школьный урок в младших классах за компьютерами «Агат». Была подробная содержательная беседа о новосибирском опыте, в частности, об исполнителе Муравей.

В Благовещенске закипела работа. В группу энтузиастов, которую возглавил А. А. Дуванов, вошли: О. Г. Какаулин, В. В. Немилостива, Ю. В. Прашкович, О. Д. Десятириков.

Появилась идея: взяв за основу среду Муравья, придумать такую её модификацию и такой учебный язык, которые, с одной стороны, были бы предельно просты и доступны самым маленьким, а с другой — позволили бы представить все основные управляющие структуры.

Результат первых опытов в этом направлении — исполнитель Тараканчик, работающий в командном режиме. Олег Какаулин спустя 18 лет реконструировал среду Тараканчика. Вы можете посмотреть, как выглядел Тараканчик на ЭВМ ИСКРА-1256 (и даже поработать с ним) на следующей ссылке:

**[www.botik.ru/~robot/history/chick.htm](http://www.botik.ru/~robot/history/chick.htm)**

Затем появился Таракан с возможностью программного управления. Позже, после поездки А. А. Дуванова на Кубу, Таракан получил второе имя — Кукарача.

Придумав исполнителя, мы даже и не подозревали обо всех его необычных возможностях. Первотолчком, закрутившим Кукарачу на новых оборотах, стали рекурсивные задачи Е. П. Лилитко, которые он придумал для своей дочки и любезно передал нам. Потом было много других задач, неожиданно

радостных находок. Особый успех Кукарача приобрёл на курсе 31 Роботландского университета ([www.botik.ru/~robot/ru](http://www.botik.ru/~robot/ru)).

Турнирные задания на этом курсе всегда неожиданны, а эмоциональный всплеск конкурсов имеет в университете повышенный накал. Каждый год куратор придумывает задачу для конкурса в надежде, что никто из ребят её не решит. Но такого ещё не случалось. Школьники дошли до того, что спокойно пишут трансляторы и даже решили для Кукарачи задачу о Ханойской башне (Первушин Данил, Снежинск)! И всё это притом, что в языке нет ни одной переменной!

## История Корректора

Корректор, как и Кукарача, переехал в Переславль-Залесский из Благовещенска (дальневосточный город, центр Амурской области, граница с Китаем). Этот исполнитель был придуман А. А. Дувановым двумя годами позже Кукарачи, в 1986 году. В этом же году была подготовлена рукопись книги «Введение в программирование. Корректор».

Однако эта рукопись так и не появилась в печати. На следующий год Дуванов переехал из Благовещенска в Переславль-Залесский и поступил на работу в Институт программных систем АН СССР в лабораторию школьной информатики, которую возглавил, приехав в Переславль из Новосибирска, Ю. А. Первин.

В Переславле закипела работа над новыми идеями. Рукопись сначала просто была положена в стол, а потом на время забыта.

Тем не менее, Корректор был одной из самых первых программ, реализованных на Ямахе и вошедших в самую раннюю версию курса «Роботландии». В дальнейшем, при переносе «Роботландии» сначала на УКНЦ, а затем на РС, Корректор остался на старой квартире. Мы не стали включать Корректор в новые версии «Роботландии», потому что его программирование выходило за рамки общеобразовательного курса информатики для младших школьников, носило факультативный характер и предназначалось для детей, которые хотели заниматься именно программированием. На пристройку к Роботландии кружка юных программистов не было сил и времени: нужно было разрабатывать и опробовать в школьных классах ставшие теперь традиционными, роботландские общеобразовательные темы.

Только в 1992 году, когда для издательства «Педагогика-Пресс» готовилась книга «Необычайные приключения Пети Кука в Роботландии» (авторы А. Дуванов, Ю. Первин), рукопись по Корректору была извлечена из картонной коробки и использована при подготовке четырёх глав приключений главного героя.

Роботландский университет дал новую жизнь Корректору. Рекурсивные опусы Кукарачи у Корректора получили логическое продолжение, более оправданное в его псевдотьюринговской среде, чем на клетчатом поле коллеги.

# Глава 1



## Олимпиада 1996/1997

### Турнир Кукарачи

— Здравствуй, Пух, — сказала она. — Какие новости?

— Грустные и ужасные, — сказал Пух, — потому что Иа-Иа, мой старый друг, потерял свой хвост, и он очень убивается о нём. Будь так добра, скажи мне, пожалуйста, как мне его найти?

— Ну, — сказала Сова, — обычная процедура в таких случаях нижеследующая...

*А. Милн*

#### 1. Деление на два (3 балла).

Автор: Е. П. Лилитко.

Кукарача расположен в верхнем левом углу поля. Где-то под ним в первом столбце в строке с чётным номером находится кубик с произвольным символом. Требуется разделить номер строки, в которой стоит кубик, на два и поставить Кукарачу в соответствующую строку. Поле считать бесконечным вниз. Пример начального и конечного состояний среды показан на рис. 1.1.

#### 2. Ишак (4 балла).

Автор: А. А. Дуванов.

На одном из перевёрнутых кубиков во втором столбце — буква Ш. Найдите её и составьте слово ИШАК. Букву Ф с поля надо убрать. Пример начального и конечного состояний среды показан на рис. 1.2.



## 3. Треугольник (5 баллов).

Автор: Е. П. Лилитко.

Первый столбец поля пуст. На остальном пространстве расположены два кубика, стоящие в одном столбце, но не в соседних строчках. Кукарача находится в первой позиции строки, содержащей первый кубик. Поставить Кукарачу так, чтобы он являлся третьей вершиной равнобедренного прямоугольного треугольника (справа от первого кубика). Поле считать бесконечным вправо и вниз. Пример начального и конечного состояний среды показан на рис. 1.3.

## 4. Прямоугольник (6 баллов).

Автор: Е. П. Лилитко.

Первый, предпоследний и последний столбцы поля  $10 \times 10$  пусты. Пусты также и две нижние строки. На остальном пространстве расположены буквы А, В и С так, что они образуют прямоугольный треугольник с размерами катетов, большими единицы. Кукарача находится в первой позиции строки, содержащей букву А. Требуется поставить Кукарачу так, чтобы он являлся четвёртой вершиной прямоугольника. Независимо от расположения букв (при точном соблюдении приведённых условий), сообщение «Не могу» не должно возникать никогда. Пример начального и конечного состояний среды показан на рис. 1.4.



Рис. 1.4

## 5. Мышь (4 балла).

Автор: А. А. Дуванов.

На одном из перевёрнутых кубиков в пятой строке поля Кукарачи — буква Ь. Найдите её и составьте слово МЫШЬ. В первом столбце третьей строки

расположена буква Ф — её надо спихнуть с поля. Пример начального и конечного состояний среды показан на рис. 1.5.



Рис. 1.5

6. Новый исполнитель (6 баллов).

Автор: Е. П. Лилитко.

Построить нового исполнителя, «управляемого данными», поведение которого будет полностью зависеть от расположения букв на поле. Изначально Кукарача движется вниз до тех пор, пока не встретит одну из букв В, Н, Л, П или С. Встретив одну из этих букв, он продолжает движение соответственно вверх, вниз, влево, вправо или немедленно останавливается. На дальнейшем пути Кукараче может вновь повстречаться одна из приведённых букв, и тогда он вновь изменяет направление вышеописанным способом. Для нового исполнителя создайте программу (установите в нужные места буквы-команды и Кукарачу), заставляющую его ходить по расширяющейся спирали.

7. Значение цифры (3 балла).

Автор: А. А. Дуванов.

Под Кукарачей во второй строке расположена цифра (её значение больше двух). Определить, какая это цифра, и поставить её в строку с соответствующим номером. Пример начального и конечного состояний среды показан на рис. 1.6.



Рис. 1.6

8. Количество букв А на поле (9 баллов).

Автор: Е. П. Лилитко.

Кукарача расположен в нижнем левом углу поля. Первая строка и первый столбец поля пусты за исключением буквы Г, расположенной точно в левом верхнем углу поля. На остальном пространстве поля разбросано некоторое количество букв А, не более, чем по одной на строку. Таким образом, некоторые строки содержат одну букву А, а некоторые — ни одной. Требуется очистить поле от букв, попутно подсчитывая их количество. В конце работы Кукарача должен остановиться в строке, номер которой равен количеству сброшенных с поля букв А. То есть, если изначально на поле была одна буква А, Кукарача останавливается в первой строке, если две — во второй и т. д. Сообщение «Не могу» допускается только в том случае, если ни одной буквы А на поле не оказалось.

9. Экран (8 баллов).

Автор: А. А. Дуванов.

Во второй строке злоумышленник перемешал буквы в слове ЭКРАН. Восстановить испорченное слово. После исправления слово может быть расположено в любой строке или даже столбце. В последнем случае его нужно читать сверху вниз. Пример начального и конечного состояний среды показан на рис. 1.7.

10. Кукарачья арифметика.

Автор: А. А. Дуванов.

Введём новые обозначения для команд исполнителя (рис. 1.8).

Программа для исполнителя теперь задаётся числом, состоящим из цифр 1, 2, 3, 4, например, короткий поход Кукарачи на рис. 1.9 запишется в виде программы так: 1234.



Рис. 1.7

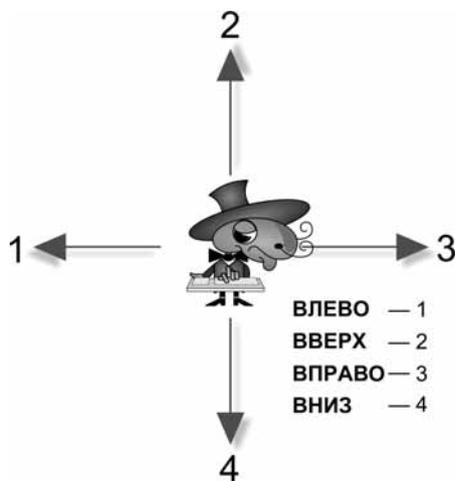


Рис. 1.8

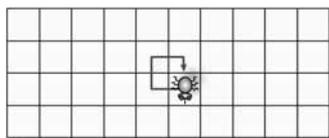


Рис. 1.9

Будем полагать также, что поле Кукарачи бесконечно (или, по крайней мере, очень большое).

- 1) (3 балла) Доказать, что если траектория движения приводит Кукарачу в клетку, с которой он начал движение, то сумма цифр числа, задающего программу исполнителя, — чётное число.
- 2) (2 балла) Верно ли обратное: если сумма цифр числа, задающего программу движения исполнителя — чётное число, то после выполнения программы Кукарача оказывается в исходной позиции.
- 3) (3 балла) Известно, что в программе исполнителя нет команд **ВВЕРХ**, а количество команд **ВЛЕВО** равно количеству команд **ВПРАВО**. Докажите, что в этом случае сумма цифр числа, изображающего программу, делится на 4, а если при этом условии ещё количество команд **ВЛЕВО** равно количеству команд **ВНИЗ**, то сумма цифр числа, изображающего программу, делится на 8.

## Глава 2



# Турниры 1997/1998

## 2.1. Турнир Кукарачи

— Вперёд! Вперёд! — кричал Кристофер Робин.  
— Вперёд! — кричали Пух и Пятачок.  
— Вперёд! — кричала Сова.  
— Тронулись! — сказал Кролик. — Я должен бежать. — И он помчался в голову колонны к Кристоферу Робину.  
— Вот именно, — сказал Иа. — Все тронулись. Но я тут ни при чём.

*А. Мшин*

### 1. Прогулка (3 балла).

Автор: А. А. Дуванов.

Помогите Кукараче совершить прогулку по полю размером  $10 \times 10$  из клетки (2,1) в клетку (1,10), не сдвигая кубиков. Начальное и конечное состояния среды показаны на рис. 2.1.

Дано		Надо	
	1 2 3 4 5 6 7 8 9 10		1 2 3 4 5 6 7 8 9 10
1	Ж Ж Ж Ж Ж Ж Ж Ж Ж	Ж Ж Ж Ж Ж Ж Ж Ж Ж	
2	☀	Ж Ж	
3	Ж Ж Ж Ж Ж Ж Ж Ж Ж	Ж Ж Ж Ж Ж Ж Ж Ж Ж	
4		Ж Ж	
5	Ж Ж Ж Ж Ж Ж Ж Ж Ж	Ж Ж Ж Ж Ж Ж Ж Ж	
6		Ж Ж	
7	Ж Ж Ж Ж Ж Ж Ж Ж Ж	Ж Ж Ж Ж Ж Ж Ж Ж	
8		Ж Ж	
9	Ж Ж Ж Ж Ж Ж Ж Ж Ж	Ж Ж Ж Ж Ж Ж Ж Ж	
10			

Рис. 2.1

## 2. Расставь цифры (4 балла).

Автор: А. А. Дуванов.

Исполнитель стоит в левом верхнем углу поля. Во второй строке, начиная с первой клетки, находится плотный ряд кубиков с цифрами. Цифры в ряду расположены в произвольном порядке и значение каждой из них больше двух. Поставить кубики в строки с номерами, равными значениям цифр. Считается, что поле исполнителя имеет 10 строк, а длина строки не ограничена. Пример начального и конечного состояний среды показан на рис. 2.2.



Рис. 2.2

## 3. Сложение двух палочных чисел (5 баллов).

Автор: А. А. Дуванов.

Будем обозначать целые положительные числа палочками (символ «|»): через | — обозначим число 1, через || — число 2, через ||| — число 3 и т. д. Иными словами, целое положительное число  $n$  будем записывать как группу из  $n$  палочек, идущих подряд в  $n$  соседних клетках одной строки. На основе введённых обозначений рассмотрим операцию сложения двух чисел. Понятно, что число палочек, обозначающих результат, равно сумме палочек, изображающих слагаемые.

А теперь заставим поработать Кукарачу, который в начальный момент находится в клетке (3,1). Во второй строке, начиная с первой клетки, записан пример на сложение двух палочных чисел (на рис. 2.3 изображён один из возможных вариантов). В конце работы программы на поле должен остаться результат вычислений. Поле исполнителя считается бесконечным вправо. Пример начального и конечного состояний среды показан на рис. 2.3.



Рис. 2.3

## 4. Вычитание двух палочных чисел (10 баллов).

Автор: А. А. Дуванов.

В третьей строке, начиная со второй клетки, записан пример на вычитание двух палочных чисел. Кукарача находится в клетке (4,2). Вычислить результат, если известно, что вычитаемое не больше уменьшаемого. Поле исполнителя считается бесконечным вправо. Пример начального и конечного состояний среды показан на рис. 2.4.



Рис. 2.4

## 5. Деление палочного числа на 2 (5 баллов).

Автор: А. А. Дуванов.

В первой строке поля, начиная с первой позиции, записано чётное палочное число. Выполнить деление его на два. В начале работы программы

исполнитель находится в клетке (2,1). Поле исполнителя считается бесконечным вправо. Пример начального и конечного состояний среды показан на рис. 2.5.



Рис. 2.5

#### 6. Умножение цифры на 11 (4 балла).

Автор: А. А. Дуванов.

В клетке (1,1) расположен кубик с ненулевой цифрой. Исполнитель стоит под ним в клетке (2,1). Умножить цифру на 11 и придумать способ отображения результата. Пример начального состояния среды показан на рис. 2.6.



Рис. 2.6



Рис. 2.7

## 7. Подсчёт букв О (8 баллов).

Автор: А. А. Дуванов.

Кукарача располагается в клетке (3,1). Прямо над ним, во второй строке с первой клетки стоит плотный ряд кубиков. Длина ряда заранее неизвестна, но известно, что в нём не менее одной буквы О. Определить количество этих букв и поставить исполнителя в строку с соответствующим номером. Поле исполнителя считать бесконечным вправо и вниз. Пример начального и конечного состояний среды показан на рис. 2.7.

## 8. Баскетбол (4 балла).

Автор: Вячеслав Пахолкин, Снежинск.

Из кубиков с буквой Щ сделан щит, а из кубиков с буквой К — кольцо для игры в баскетбол. Исполнитель находится в любой точке поля справа от щита, но выше 8-й и ниже 2-ой строки. Мяч — кубик с буквой О — находится на строку выше исполнителя в одном с ним столбце. Исполнителю нужно забросить мяч в кольцо. Разрешается сдвигать кубики, образующие щит, и один кубик кольца, примыкающий к щиту. Поле считается бесконечным вправо. Пример начального и конечного состояний среды показан на рис. 2.8.

## 9. Умножение на 10 (4 балла).

Автор: И. Ю. Донская, Ангарск.

В четвёртой строке поля, начиная со второй клетки, записано дробное десятичное число. Запись числа содержит произвольное количество цифр, целая часть отделяется от дробной точкой (она не последняя в записи). Исполнитель расположен под первым из кубиков записи. Умножить число на 10. Поле исполнителя считается бесконечным вправо. Пример начального и конечного состояний среды показан на рис. 2.9.



Рис. 2.8



Рис. 2.9



Рис. 2.10

## 10. Разворот слова (6 баллов).

Автор: Павел Пинженин, Челябинск.

В одной из строк (не в первой), начиная с диагональной клетки, написано слово длиной более одной буквы. Развернуть слово по вертикали, начиная с той же клетки, в которой находится первая буква слова. Исполнитель в начальный момент находится в клетке (1,1). Поле считать бесконечным вправо и вниз. Пример начального и конечного состояний среды показан на рис. 2.10.

## 11. Количество согласных (8 баллов).

Авторы: Елисеева и Янулевич, Иркутск.

В начальный момент исполнитель стоит в клетке (1,1) перед словом, содержащим хотя бы одну согласную букву. Результат работы: чистое поле, а номер строки, в которой стоит Кукарача, равен количеству согласных букв слова. Поле исполнителя считать бесконечным вниз и вправо. Пример начального и конечного состояний среды показан на рис. 2.11.



Рис. 2.11

## 12. Середина отрезка (6 баллов).

Авторы: Игорь Калинин и Павел Сазонтов, Находка.

Во втором столбце поля — два кубика. Один в клетке (1,2), другой — где-то ниже, в нечётной строке. В начальный момент Кукарача расположен в клетке (2,2) под первым кубиком. Поставить исполнителя в середину отрезка между двумя кубиками. Кубики должны остаться на прежних местах. Поле исполнителя считать бесконечным вниз. Пример начального и конечного состояний среды показан на рис. 2.12.



Рис. 2.12

## 2.2. Турнир Корректора

— Пух, — сказал он, — где ты нашёл эту ось?

Пух посмотрел на палку, которую всё ещё продолжал держать.

— Ну, просто нашёл, — сказал он. — Разве это ось? Я думал это просто палка и она может пригодиться. Она там торчала в земле, а я её поднял.

— Пух, — сказал Кристофер Робин торжественно, — экспедиция окончена. Это — Земная Ось. Мы нашли Северный полюс.

*А. Милн*

Далее вводится определение понятий «обычного» и «палочного» чисел. Эти понятия используются в формулировках некоторых задач турнира.

**Обычное число.** Этот термин обозначает целое неотрицательное число, записанное арабскими цифрами в десятичной системе счисления. Примеры: 0, 25, 48754.

**Палочное число.** Это обычное число, записанное на ленту в виде цепочки символов «|» (палочек). Значение «палочного» числа равно числу изображающих его палочек. Например, число 3 в палочной записи изображается как |||, а число 5 — как |||||. Ноль изображается на ленте пустой клеткой (не пробелом!).

Кроме того, будем называть символы алфавита Корректора, когда они используются для вычислений, *символьными числами*.

Значением символьного числа будем считать номер соответствующего символа в алфавите Корректора по отношению к символу «0» (нуль), номер которого полагается равным числу 0.

Под арифметическими действиями над символьными числами будем понимать соответствующие действия над их алфавитными номерами.

1. Копия в обратном порядке (5 баллов).

Автор: А. А. Дуванов.

На ленте записан некоторый текст. Требуется записать на ленту его копию в обратном порядке символов. В начальный момент в окне виден первый символ текста. Исходное слово на ленте можно не сохранять. На рис. 2.13 показан пример начального и конечного состояний среды.



Рис. 2.13

2. Палиндром (10 баллов).

Автор: А. А. Дуванов.

Палиндромами называют слова, которые одинаково читаются как слева направо, так и справа налево. Примеры палиндромов: шалаш, потоп, казак. На ленте Корректора записано слово. Проверить, является ли оно палиндромом. В начальный момент в окне виден первый символ слова. Ответ записать в виде слов ДА или НЕТ. Исходное слово на ленте можно не сохранять. На рис. 2.14 показан пример начального и конечного состояний среды.



Рис. 2.14



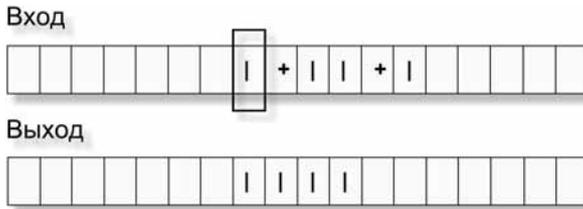


Рис. 2.17

6. Количество цифр в тексте (6 баллов).

Автор: А. А. Дуванов.

На ленте записан некоторый текст. Подсчитать количество цифр в нём. В начальный момент окно установлено на первый символ текста. На рис. 2.18 показан пример начального и конечного состояний среды.

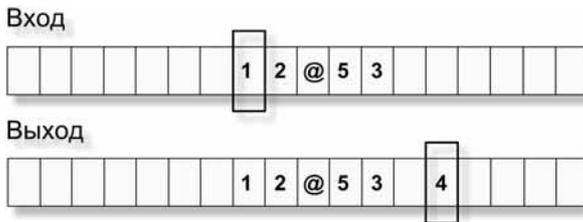


Рис. 2.18

7. Кратность трём (10 баллов).

Автор: А. А. Дуванов.

На ленте записано обычное число. Проверить, кратно ли оно трём. В начальный момент окно установлено на первый символ числа. На рис. 2.19 показан пример начального и конечного состояний среды.

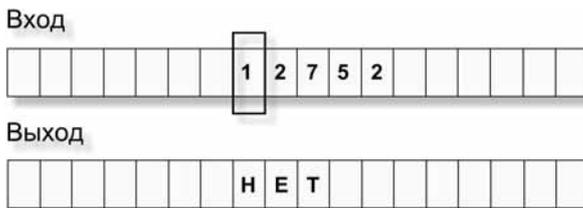


Рис. 2.19

8. Дополнение до 9 (4 балла).

Автор: А. А. Дуванов.

На ленте записано обычное число. Заменить каждую его цифру дополнением до девяти. Например, число 1456 должно быть преобразовано в число 8543. В начальный момент окно установлено на первый символ

записи. Удалить незначащие нули в ответе не надо. На рис. 2.20 показан пример начального и конечного состояний среды.

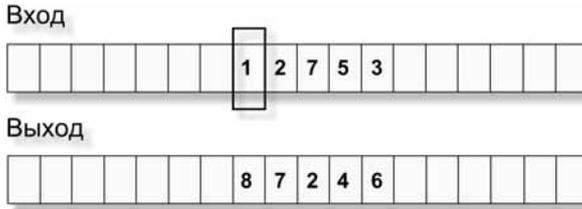


Рис. 2.20

9. Испорченное слово (6 баллов).

Автор: Роман Скочилов, Снежинск.

Злоумышленник испортил слово на ленте, вставив между буквами различные посторонние символы. Ни один из вставленных символов буквой не является. Восстановить исходное слово так, чтобы все его буквы прилегли плотно друг к другу. В начальный момент окошко установлено на первый символ записи. На рис. 2.21 показан пример начального и конечного состояний среды.

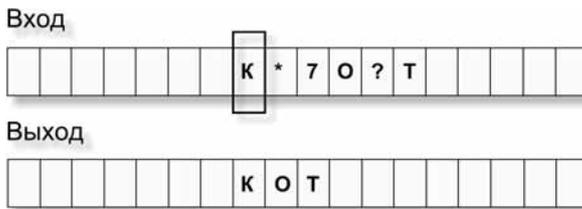


Рис. 2.21

10. Алфавит записи (6 баллов).

Авторы: Крашенинников, Метляев; Иркутск.

Построить алфавит записи на ленте — набор различных символов, из которых она состоит. В начальный момент окно установлено на первый символ записи. На рис. 2.22 показан пример начального и конечного состояний среды.



Рис. 2.22



## Глава 3



# Турниры 1998/1999

## 3.1. Турнир Кукарачи

Сквозь болота и туман  
Пробирается Фортран,  
Сквозь дубовый частокол  
Продирается Алгол,  
По чашобам куролеся,  
Семенит несчастный Бейсик,  
А по кочкам, скрывшим даль,  
Нервно прыгает Паскаль.  
Эти все болота, кочки,  
И деревья, и кусточки,  
И туман молочно-мглистый —  
Мозг бедняги программиста.

*Алексей Морозов*

## Пояснительная записка

Дополнительно к условиям задач используйте следующие разъяснения.

### Задача 3

- Расположение последнего кубика с символом «!» в записи заранее не известно.
- Горизонтальный размер поля считается настолько большим, насколько это необходимо для решения задачи.
- Кубики с символами могут располагаться в любой строке (даже первой) начиная с любого столбца (даже первого).

### Задачи 4–8

- Расположение записи в строке совершенно произвольное.
- Не разрешается смещать по горизонтали кубики, составляющие запись.

- Разрешается смещать запись на строку вверх, но структура её не должна меняться: после выполнения программы кубики должны следовать друг за другом без промежутков в прежнем порядке.
- Горизонтальный размер поля считается настолько большим, насколько это необходимо для решения задачи.

### Задачи 9–13

- Кубики с символами **a** и **b** могут располагаться в любой строке поля (даже самой первой).
- Кубик с символом **a** может располагаться в любом столбце, даже самом первом.
- После выполнения программы все кубики должны оставаться на прежних местах (кроме задачи 12, где кубик с символом **c** должен перемещаться вниз по условию задачи).
- Горизонтальный и вертикальный размер поля считаются настолько большими, насколько это необходимо для решения задачи.

### Разминка

1. Кубик в прямоугольнике (4 балла).

Автор: А. А. Дуванов.

Кукарача находится в верхнем левом углу поля. В прямоугольнике, образованном клетками (1,3), (1,7), (9,3), (9,7), расположен кубик. Найти этот кубик и поставить его в клетку (1,1). Пример начального и конечного состояний среды показан на рис. 3.1.



Рис. 3.1

## 2. Кубик с цифрой (5 баллов).

Автор: А. А. Дуванов.

В прямоугольнике, образованном клетками (3,3), (3,7), (9,3), (9,7), расположен кубик с ненулевой цифрой. Установить этот кубик в первом столбце в клетку, номер строки которой равен значению цифры на кубике. В начальный момент исполнитель расположен в клетке (3,1). Пример начального и конечного состояний среды показан на рис. 3.2.



Рис. 3.2

## 3. Уплотнение записи (7 баллов).

Автор: А. А. Дуванов.

В одной строке поля расположен неплотный ряд кубиков (между кубиками могут располагаться пустые клетки). Известно, что среди кубиков есть только один кубик с символом «!», и он — самый последний. Написать программу, которая уплотняет символы — сдвигает их в строке так, чтобы все они шли подряд без промежутков. В начальный момент исполнитель расположен перед первым кубиком в одной с ним строке. Пример начального и конечного состояний среды показан на рис. 3.3.



Рис. 3.3

## Синтаксический анализ записи

В программировании часто используют язык Бэкуса-Наура для описания определений.

В языке Бэкуса-Наура используются следующие символы:

- $\langle \rangle$  — скобки, выделяющие определяемое понятие;
- $::=$  — составной символ, который читается «по определению есть»;
- $|$  — символ, обозначающий альтернативу, читается «или».

Введём такое определение числа.

### Определение 1

$\langle \text{число} \rangle ::= \langle \text{цифра} \rangle | \langle \text{число} \rangle \langle \text{цифра} \rangle$  (1)

$\langle \text{цифра} \rangle ::= 0 | 1$  (2)

Прочитать это определение можно так:

«Число» по определению есть: или «цифра», или объект, про который мы знаем, что он — «число», вслед за которым записана «цифра».

«Цифра» — это либо символ «0», либо символ «1».

Как видите, первая строка в определении содержит рекурсию!

#### 4. Анализатор-1 (5 баллов).

Автор: А. А. Дуванов.

Напишите программу, которая проверяет, является ли запись на его поле числом в смысле определения 1. В начальный момент исполнитель расположен перед кубиками с записью во второй строке:

■ запись

Если проверяемая запись — число, установить исполнитель в конец записи:

запись ■

Если запись — не число, поставить исполнитель в третью строку поля.

Пример начального и конечного состояний среды показан на рис. 3.4.



Рис. 3.4

## 5. Анализатор-2 (7 баллов).

Автор: А. А. Дуванов.

Решить задачу 4, но, если запись не является числом, установить исполнитель под первым неверным символом в записи. Например:

1021

■

Если запись — число, установить исполнитель за последним ее символом. Например:

1011■

Пример начального и конечного состояний среды показан на рис. 3.5.



Рис. 3.5

## 6. Анализатор-3 (8 баллов).

Автор: А. А. Дуванов.

Введём следующее определение числа:

**Определение 2**

<число> ::= <целое> | <дробное>

<целое> ::= <цифра> | <целое><цифра>

<дробное> ::= .<целое> | <целое>. | <целое>.<целое>

<цифра> ::= 0 | 1

Напишите программу для Кукарачи, которая проверяет, является ли запись на его поле числом в смысле определения 2. В начальный момент исполнитель расположен перед кубиками с записью во второй строке:

■запись

Если проверяемая запись — число, установить исполнитель в конец записи:

запись■

Если запись — не число, поставить исполнитель под первым неверным символом:

запись

■

Пример начального и конечного состояний среды показан на рис. 3.6.

		Дано										Надо										
		1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	
1														1	A	0	2	6	6	6		
2	☀	1	A	0	2	6	6	6						☀								
3																						
4																						

Рис. 3.6

### 7. Анализатор-4 (9 баллов).

Автор: А. А. Дуванов.

Введем следующее определение числа:

#### Определение 3

<число> ::= <целое> | <дробь>

<целое> ::= <цифра> | <целое><цифра>

<дробь> ::= <простая> | <десятичная>

<простая> ::= <целое>/<целое>

<десятичная> ::= .<целое> | <целое>. | <целое>.<целое>

<цифра> ::= 0 | 1

Напишите программу для Кукарачи, которая проверяет, является ли запись на его поле числом в смысле определения 3. В начальный момент исполнитель расположен перед кубиками с записью во второй строке:

■ запись

Если проверяемая запись — число, установить исполнитель в конец записи:

запись ■

Если запись — не число, поставить исполнитель под первым неверным символом:

запись

■

Пример начального и конечного состояний среды показан на рис. 3.7.

		Дано										Надо										
		1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	
1														1	/	1	0	.	1	1		
2	☀	1	/	1	0	.	1	1										☀				
3																						
4																						

Рис. 3.7

## 8. Анализатор-5 (9 баллов).

Автор: А. А. Дуванов.

Введем следующее определение выражения:

**Определение 4** $\langle \text{выражение} \rangle ::= \langle \text{число} \rangle + \langle \text{число} \rangle \mid \langle \text{число} \rangle + \langle \text{выражение} \rangle$  $\langle \text{число} \rangle ::= \langle \text{цифра} \rangle \mid \langle \text{число} \rangle \langle \text{цифра} \rangle$  $\langle \text{цифра} \rangle ::= 0 \mid 1$ 

Напишите программу для Кукарачи, которая проверяет, является ли запись на его поле выражением в смысле определения 4. В начальный момент исполнитель расположен перед кубиками с записью во второй строке:

■ запись ■

Если проверяемая запись — выражение, установить исполнитель в конец записи:

запись ■

Если запись — не выражение, поставить исполнитель под первым неверным символом:

запись

■

Пример начального и конечного состояний среды показан на рис. 3.8.



Рис. 3.8

**Кукарачья геометрия****Определение 5**

Будем рассматривать каждую клетку на поле исполнителя как геометрическую точку. Координаты точки отсчитываются от левого верхнего угла поля и обозначаются как обычные точки на плоскости парой чисел  $(x, y)$ . Число  $x$  — номер строки, а  $y$  — номер столбца, на пересечении которых располо-

жена клетка-точка. Первое число  $x$  будем называть строкой точки, второе  $y$  — столбцом точки. Точки будем обозначать малыми латинскими буквами, приписывая к ним, в случае необходимости, координаты. Точка  $\mathbf{o}$ , соответствующая первой клетке в первой строке, имеет координаты  $(1,1)$ . Таким образом, она обозначается как  $\mathbf{o}(1,1)$ .

## Определение 6

Горизонтальным отрезком на поле Кукарачи назовём упорядоченную пару точек, имеющих одинаковую строку. При этом столбец первой точки пары меньше столбца второй. Отрезок будем обозначать двумя латинскими буквами — названиями точек, его образующих. При необходимости в скобках можно указывать координаты точек, отделяя их друг от друга запятой.

Например, горизонтальный отрезок, образуемый двумя точками  $\mathbf{a}(x,y_1)$  и  $\mathbf{b}(x,y_2)$ , можно обозначать как  $\mathbf{ab}$  или как  $\mathbf{ab}((x,y_1),(x,y_2))$ .

Длиной горизонтального отрезка  $\mathbf{ab}((x,y_1),(x,y_2))$  назовём число  $y_2 - y_1$ . Обозначать длины горизонтальных отрезков будем как  $\mathbf{Pab}$  или  $\mathbf{Pab}((x,y_1),(x,y_2))$ . Таким образом, можно записать:

$$\mathbf{Pab}((x,y_1),(x,y_2)) = y_2 - y_1$$

Будем считать, что точка  $\mathbf{t}(t_1,t_2)$  на поле исполнителя принадлежит горизонтальному отрезку  $\mathbf{ab}((x,y_1),(x,y_2))$ , если:

$$t_1 = x$$

$$y_1 \leq t_2 \leq y_2$$

## Определение 7

Вертикальным отрезком на поле Кукарачи назовём упорядоченную пару точек, имеющих одинаковый столбец. При этом строка первой точки пары меньше строки второй. Отрезок будем обозначать двумя латинскими буквами — названиями точек, его образующих. При необходимости в скобках можно указывать координаты точек, отделяя их друг от друга запятой.

Например, вертикальный отрезок, образуемый двумя точками  $\mathbf{a}(x_1,y)$  и  $\mathbf{b}(x_2,y)$ , можно обозначать как  $\mathbf{ab}$  или как  $\mathbf{ab}((x_1,y),(x_2,y))$ .

Длиной вертикального отрезка  $\mathbf{ab}((x_1,y),(x_2,y))$  назовём число  $x_2 - x_1$ . Обозначать длины горизонтальных отрезков будем как  $\mathbf{Pab}$  или  $\mathbf{Pab}((x_1,y),(x_2,y))$ . Таким образом, можно записать:

$$\mathbf{Pab}((x_1,y),(x_2,y)) = x_2 - x_1$$

Будем считать, что точка  $\mathbf{t}(t_1,t_2)$  на поле исполнителя принадлежит вертикальному отрезку  $\mathbf{ab}((x_1,y),(x_2,y))$ , если:

$$x_1 \leq t_1 \leq x_2$$

$$t_2 = y$$

## 9. Квадрат (5 баллов).

Автор: А. А. Дуванов.

Кубики **a** и **b** на поле исполнителя образуют горизонтальный отрезок **ab**. В начальный момент Кукарача расположен на этом отрезке сразу за кубиком **a**. Написать программу, которая поставит Кукарачу в нижний правый угол квадрата, построенного на отрезке **ab** как на верхней его стороне. Пример начального и конечного состояний среды показан на рис. 3.9.



Рис. 3.9

## 10. Срединный перпендикуляр (6 баллов).

Автор: А. А. Дуванов.

**Определение 8**

Горизонтальный отрезок, концевая точка которого принадлежит вертикальному отрезку, называется перпендикуляром к вертикальному отрезку.

Соответственно вертикальный отрезок, концевая точка которого совпадает с одной из точек горизонтального отрезка, называется перпендикуляром к горизонтальному отрезку.

Перпендикуляр к отрезку называется срединным, если его концевая точка совпадает с точкой, расположенной точно в середине отрезка.

Понятно, что срединный перпендикуляр можно построить только к отрезку с длиной, выражающейся чётным числом.

Задание. Кубики **a** и **b** на поле исполнителя образуют горизонтальный отрезок **ab**. Известно, что длина отрезка **ab** выражается чётным числом. В начальный момент Кукарача расположен на этом отрезке, сразу за кубиком **a**. Установить исполнителя в точку на срединном перпендикуляре, построен-

ном к отрезку **ab** вниз так, чтобы длина вертикального отрезка, изображающего перпендикуляр, равнялась половине длины отрезка **ab**. Пример начального и конечного состояний среды показан на рис. 3.10.



Рис. 3.10

11. Вниз на **Рab** (8 баллов).

Автор: А. А. Дуванов.

Исполнитель расположен где-то внутри горизонтального отрезка **ab**. Сместить его вниз из исходного положения на **Рab**. Известно, что столбец точки **a** больше 2. Пример начального и конечного состояний среды показан на рис. 3.11.



Рис. 3.11

## 12. Прямоугольник (10 баллов).

Автор: А. А. Дуванов.

Кубики **a** и **b** на поле исполнителя образуют горизонтальный отрезок **ab**. В начальный момент Кукарача расположен на этом отрезке сразу за кубиком **a**. Кубик **c** установлен на поле так, что образуется вертикальный отрезок **bc** и  $Pbc = 2$ . Построить на поле из кубиков **a**, **b**, **c** и исполнителя прямоугольник **abc**. Длина одной стороны прямоугольника должна равняться длине исходного отрезка **ab**, длина другой (сторона **bc**) — в два раза больше. Пример начального и конечного состояний среды показан на рис. 3.12.



Рис. 3.12

## 13. Деление отрезка (6 баллов).

Автор: Павел Пинженин, Челябинск.

Кубики **a**, **b** и **c** последовательно расположены во 2-й строке таким образом, что длина отрезка **ab** больше длины отрезка **bc**. Исполнитель расположен где-то между кубиками **a** и **b**. Поставить исполнителя так, чтобы длина отрезка **ac** стала равна длине отрезка **bc** ( $Pa_c = Pbc$ ). Не разрешается смещать кубики по горизонтали. Кубики можно смещать на строку вверх, сохраняя заданные длины отрезков **ab** и **bc**. Пример начального и конечного состояний среды показан на рис. 3.13.



Рис. 3.13

## 3.2. Турнир Корректора

А вы  
ноктюрн сыграть  
могли бы  
на флейте водосточных труб?

*В. В. Маяковский*

### Синтаксический анализ выражений и трансляторы

1. Анализатор-1 (5 баллов).

Автор: А. А. Дуванов.

Введём следующее определение:

#### Определение

$\langle \text{число} \rangle ::= \langle \text{целое} \rangle \mid \langle \text{дробь} \rangle$

$\langle \text{целое} \rangle ::= \langle \text{цифра} \rangle \mid \langle \text{целое} \rangle \langle \text{цифра} \rangle$

$\langle \text{дробь} \rangle ::= \langle \text{простая} \rangle \mid \langle \text{десятичная} \rangle$

$\langle \text{простая} \rangle ::= \langle \text{целое} \rangle / \langle \text{целое} \rangle$

$\langle \text{десятичная} \rangle ::= . \langle \text{целое} \rangle \mid \langle \text{целое} \rangle . \mid \langle \text{целое} \rangle . \langle \text{целое} \rangle$

$\langle \text{цифра} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

Напишите программу, которая проверяет запись на ленте и помещает результат проверки справа от неё в виде двух букв:

- ОШ — запись числом не является;
- ЦЛ — запись есть «целое»;
- ДС — запись есть «десятичная»;
- ПР — запись есть «простая».

В начальный момент окошко расположено перед записью, а в конце работы — сразу за ней, если в записи нет ошибок. Если запись неверная, окошко должно указывать на первый ошибочный символ. Пример начального и конечного состояний среды показан на рис. 3.14.

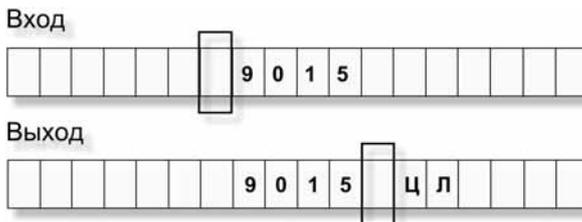


Рис. 3.14

## 2. Анализатор-2 (8 баллов).

Автор: А. А. Дуванов.

Введём следующее определение:

**Определение** $\langle \text{выражение} \rangle ::= \langle \text{число} \rangle + \langle \text{число} \rangle \mid \langle \text{выражение} \rangle + \langle \text{число} \rangle$  $\langle \text{число} \rangle ::= \langle \text{целое} \rangle \mid \langle \text{дробь} \rangle$  $\langle \text{целое} \rangle ::= \langle \text{цифра} \rangle \mid \langle \text{целое} \rangle \langle \text{цифра} \rangle$  $\langle \text{дробь} \rangle ::= . \langle \text{целое} \rangle \mid \langle \text{целое} \rangle . \mid \langle \text{целое} \rangle . \langle \text{целое} \rangle$  $\langle \text{цифра} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$ 

Напишите программу, которая проверяет запись на ленте и помещает результат проверки справа от неё в виде двух букв:

- ОШ — запись не является ни числом, ни выражением;
- ЧС — запись есть число;
- ВР — запись есть выражение.

В начальный момент окошко расположено перед записью, а в конце работы — сразу за ней, если в записи нет ошибок. Если запись содержит ошибки, окошко должно указывать на первый ошибочный символ. Пример начального и конечного состояний среды показан на рис. 3.15.

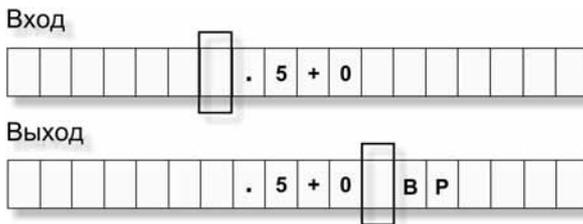


Рис. 3.15

## 3. Транслятор-1 (9 баллов).

Автор: А. А. Дуванов.

Введём следующее определение:

**Определение** $\langle \text{выражение} \rangle ::= \langle \text{число} \rangle \mid \langle \text{выражение} \rangle + \langle \text{число} \rangle$  $\langle \text{число} \rangle ::= 0 \mid 1 \mid 2$ 

Напишите программу, которая вычисляет выражение, если оно правильное, или записывает на ленту сообщение ОШ, если запись содержит

ошибку. Если запись ошибочная, то дополнительно к сообщению ОШ окошко должно указывать на первый неверный символ.

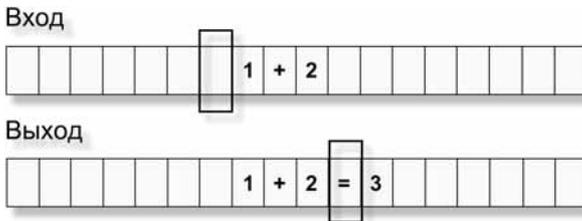
### **Замечание 1**

Значение выражения должно быть записано на ленте в виде обычного целого десятичного числа.

### **Замечание 2**

Предполагается, что результат вычисления выражения не превышает длины алфавита Корректора.

В начальный момент окошко расположено перед записью. Пример начального и конечного состояний среды показан на рис. 3.16.



**Рис. 3.16**

## 4. Транслятор-2 (9 баллов).

Автор: А. А. Дуванов.

Введём следующее определение:

### **Определение**

$\langle \text{выражение} \rangle ::= \langle \text{число} \rangle \mid \langle \text{выражение} \rangle + \langle \text{число} \rangle \mid \langle \text{выражение} \rangle - \langle \text{число} \rangle$   
 $\langle \text{число} \rangle ::= 0 \mid 1 \mid 2$

Напишите программу, которая вычисляет выражение, если оно правильное, или записывает на ленту сообщение ОШ, если запись содержит ошибку. Если запись ошибочная, то дополнительно к сообщению ОШ окошко должно указывать на первый неверный символ.

### **Замечание 1**

Значение выражения должно быть записано на ленте в виде обычного целого десятичного числа.

### **Замечание 2**

Предполагается, что конечный и промежуточный результаты вычислений не превышают числа, равного длине алфавита Корректора.

**Замечание 3**

Если конечный или промежуточный результат вычисления выражения окажется меньше нуля, записать на ленту сообщение АВ (авария) и установить окошко на знак операции, которая приводит к отрицательному результату.

В начальный момент окошко расположено перед записью. Пример начального и конечного состояний среды показан на рис. 3.17.

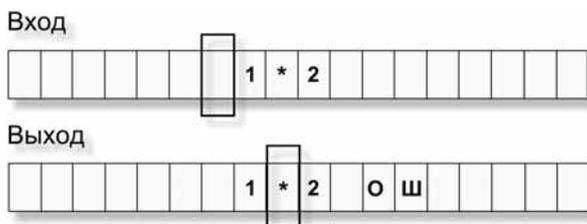


Рис. 3.17

## 5. Анализатор-3 (10 баллов).

Автор: А. А. Дуванов.

Введём следующее определение:

**Определение**

$\langle \text{выражение} \rangle ::= \langle \text{число} \rangle \mid$

$\langle \text{выражение} \rangle + \langle \text{выражение} \rangle \mid$

$\langle \text{выражение} \rangle - \langle \text{выражение} \rangle \mid$

$(\langle \text{выражение} \rangle)$

$\langle \text{число} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

Напишите программу, которая проверяет правильность записи выражения. Если запись правильная, то нужно за ней на ленте записать ОК. Если запись ошибочная, то нужно записать ОШ и дополнительно установить окошко на первый неверный символ или на место обнаружения ошибки в случае ошибок со скобками. В начальный момент окошко расположено перед записью. Пример начального и конечного состояний среды показан на рис. 3.18.

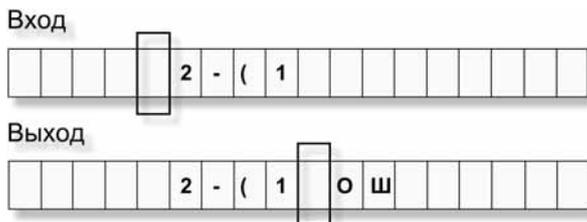


Рис. 3.18



## Геометрия на ленте Корректора

Введём следующие определения.

### Определение 1

Под точкой в среде Корректора будем понимать любую клетку на ленте. При письме будем обозначать точки маленькими латинскими буквами.

### Определение 2

Расстоянием между двумя точками **a** и **b** назовём неотрицательное целое число **Pab**, на единицу большее минимального числа клеток между этими точками на ленте.

Например, если **a** и **b** — соседние клетки, то расстояние между ними равно единице (**a** не **L-1**, если за **L** обозначить длину склеенной в кольцо ленты Корректора).

Очевидно, что расстояние между точками **a** и **b** равно минимальному числу команд **ВПРАВО** (или **ВЛЕВО**), необходимых для перемещения окна из точки **a** в точку **b**.

### Определение 3

Точка **b** расположена на ленте правее точки **a**, если потребуется **Pab** команд **ВПРАВО** для перемещения окна из точки **a** в точку **b**. Соответственно, точка **a** расположена левее точки **b**, если потребуется **Pab** команд **ВЛЕВО** для перемещения окна из точки **b** в точку **a**.

Обозначение операций отношений в этом тексте:

- $a = b$  — точки совпадают;
- $a \neq b$  — точки не совпадают;
- $a < b$  — точка **a** левее точки **b**;
- $a > b$  — точка **a** правее точки **b**;
- $a \leq b$  — точка **a** левее точки **b** или совпадает с ней;
- $a \geq b$  — точка **a** правее точки **b** или совпадает с ней.

*Вопрос 1.* Может ли точка **a** быть одновременно левее и правее точки **b**?

*Ответ.* Да, если точки совпадают (т. е. изображаются одной клеткой). В этом случае записывают  $a = b$ .

*Вопрос 2.* Может ли точка **a** быть одновременно левее и правее точки **b**, если она с ней не совпадает?

*Ответ.* Да. Например, при длине ленты равной 6, точка **a** на следующей схеме одновременно и левее и правее точки **b**:

$\square a \square \square b \square$

В наших рассуждениях такие случаи учитывать не будем, считая, что длина ленты всегда существенно больше длины отрезка **ab**.

### Определение 4

Пусть  $a \leq b$ . Отрезком **ab** на ленте назовём множество точек **c**, таких, что  $a \leq c$  и одновременно  $c \leq b$ . Точку **a** назовём началом отрезка, точку **b** — его концом.

*Упражнение.* Закрасьте клетки отрезка **ab** для 10-клеточной ленты:

□□□a□□b□□□

### Определение 5

Длиной отрезка **ab** назовём число **Pab**, равное расстоянию между точками **a** и **b**.

Заметим, что в силу определений, если точка **c** принадлежит отрезку **ab**, то  $Pac + Pcb = Pab$ .

*Вопрос.* Рассмотрим определение 4 без ограничения «пусть  $a \leq b$ ». Верно ли при этом, что для любой точки **c** из отрезка **ab**:  $Pac + Pcb = Pab$ ?

### Определение 6

Окружностью на ленте назовём любой отрезок, у которого длина выражается чётным числом. Длина отрезка в этом случае называется диаметром окружности, а половина длины — радиусом. Центром окружности **ab** назовём точку **c** такую, что  $Pac = Pcb$ .

*Упражнение.* Докажите, что любая окружность имеет центр.

*Вопрос 1.* Может ли окружность иметь несколько центров?

*Вопрос 2.* Могут ли несколько окружностей иметь один центр?

### Определение 7

Две окружности пересекаются, если отрезки, их изображающие, имеют общие точки.

*Вопрос.* Могут ли не пересекаться окружности, имеющие общий центр?

Далее предложен небольшой набор «геометрических» задач Корректора. Построенные определения дополняют условия этих заданий, являясь их составной частью, вынесенной «за скобки».

При решении задач нужно пользоваться именно этими определениями, а не собственными представлениями о точках, отрезках и окружностях на ленте Корректора.

Дополнительно будем считать, что символ «0» в алфавите Корректора имеет порядковый номер 0, символ «1» — 1, символ «A» — 10 и так далее (символьные числа Корректора).

## 7. Закраска окружности (3 балла).

Автор: А. А. Дуванов.

Окно установлено в центр окружности. Закрасить пробелом все точки окружности (кроме центра). Значение радиуса окружности помещено в клетку-центр в виде символического числа Корректора. Пример начального и конечного состояний среды показан на рис. 3.20.

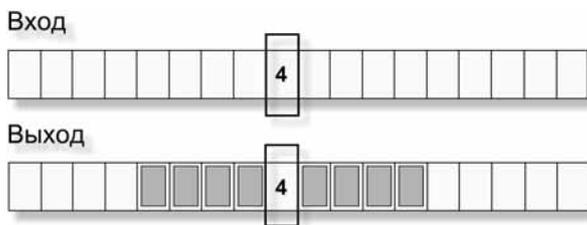


Рис. 3.20

## 8. Закраска пересечения окружностей (8 баллов).

Автор: А. А. Дуванов.

На ленте Корректора заданы две окружности с центрами в разных точках  $A$  и  $B$ . Известно, что  $A < B$  и радиусы обеих окружностей меньше 50. В клетках, изображающих центры окружностей, записаны символы, задающие радиусы окружностей (как в условии задачи 7). Написать программу, которая закрасит пробелами все клетки пересечения окружностей (кроме точек  $A$  и  $B$ , даже если они попадают в пересечение). В начальный момент окошко установлено в точке  $A$ . Пример начального и конечного состояний среды показан на рис. 3.21.

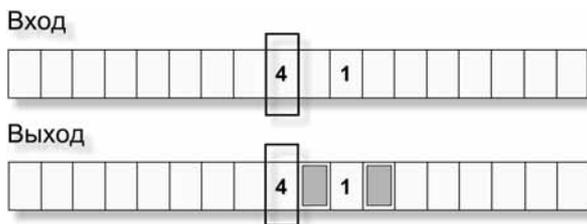


Рис. 3.21

## 9. Последовательность Фибоначчи (6 баллов).

Автор: А. А. Дуванов.

Последовательность чисел:

1, 1, 2, 3, 5, 8, 13, ...

называется последовательностью Фибоначчи. Первые два члена этой последовательности равны единице, а каждый следующий — сумме двух предыдущих.

Построить на ленте 8 точек так, чтобы расстояния между первой и второй, второй и третьей, третьей и четвёртой и так далее образовывали последовательность Фибоначчи.

В клетки, изображающие на ленте последовательность, поместить символы, порядковый номер которых в алфавите исполнителя равен численному значению соответствующего числа Фибоначчи.

Пример начального и конечного состояний среды показан на рис. 3.22.

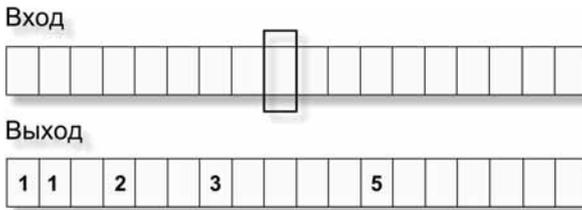


Рис. 3.22

10. Последовательность Фибоначчи-2 (8 баллов).

Автор: А. А. Дуванов.

В условиях предыдущей задачи построить на ленте  $n$  первых членов последовательности Фибоначчи ( $n > 2$ ). В начальный момент окно смотрит на клетку, символ в которой задаёт своим порядковым номером в алфавите Корректора число  $n$ . Пример начального и конечного состояний среды показан на рис. 3.23.

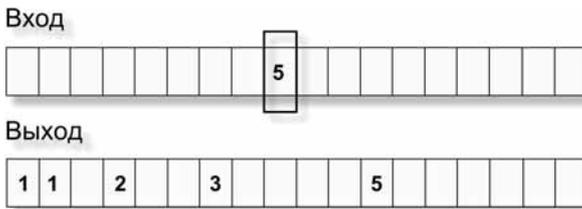


Рис. 3.23

11. Разметка отрезка (3 балла).

Автор: А. А. Дуванов.

Отрезок  $ab$  ( $a < b$ ) имеет ненулевую длину, кратную 3.

Разделить его на три равные части, в клетках-точках деления поставить символ «#». Начальная и конечная точки отрезка обозначены на ленте непустыми (заранее не известными) символами. В начальный момент окошко установлено на левый конец отрезка. Пример начального и конечного состояний среды показан на рис. 3.24.

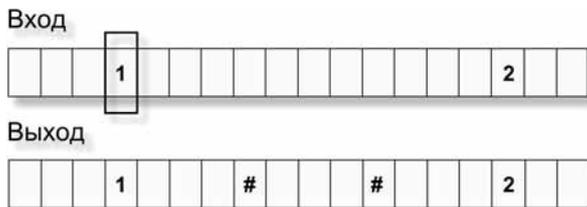


Рис. 3.24

## 12. Построение отрезка (5 баллов).

Автор: А. А. Дуванов.

Окошко смотрит на ячейку, в которой записан символ, порядковый номер которого (в алфавите Корректора) задаёт длину отрезка  $ab$ . Известно, что  $Pab > 0$ .

Построить отрезок на ленте, используя символ «.» для обозначения его концов, и поделить его пополам, если это возможно. В качестве обозначения средней точки использовать символ «#». Пример начального и конечного состояний среды показан на рис. 3.25.

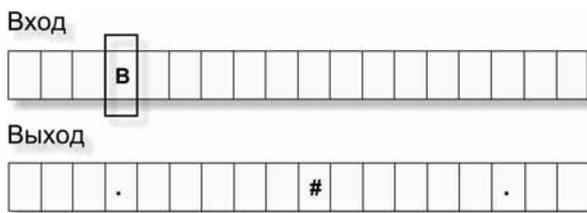


Рис. 3.25

## Глава 4



# Турниры 1999/2000

## 4.1. Турнир Кукарачи

Таил в молчанье он глубоко  
Движенья сердца своего,  
И на челе его высоком  
Не изменялось ничего.

*А. С. Пушкин*

Слова Александра Сергеевича — про нашего молчуна Кукарачу Залесского. Хотя на самом деле Кукарача родился не в Переславле, а на Дальнем Востоке, на пограничной реке Амур. В том самом месте, где в Амур впадает Зея. И случилось это в 1984 году.

Сначала Кукарача (тогда его называли попросту — Таракан) трудился на ЭВМ Искра-1256. Трудно теперь представить себе такой персональный компьютер. Объем его оперативной памяти составлял 4 Кбайта! Устройством ввода служил обычный кассетный магнитофон. Графического вывода на экран, конечно, не было. Вот в каких условиях пришлось работать нашему герою. Потом были: Искра-226, Ямаха, УК-НЦ, РС.

## Разминка

Разминка включает в себя три простых задания. Во всех этих задачах считается, что справа поле велико настолько, насколько это необходимо для решения.

1. Сложение строк (5 баллов).

Автор: А. А. Дуванов.

Введем несколько определений.

### Определение 1

Строка — это последовательность из нескольких символов.

## Определение 2

Пустая строка — это пустое место, отсутствие каких-либо символов (кубиков).

## Определение 3

Операция соединения двух строк в одну строку называется операцией конкатенации. Обычно эта операция в языках программирования обозначается так же, как и операция сложения — знаком «+».

Примеры выполнения операции конкатенации.

- каша + лот = кашалот;  
соединяются две непустые строки.
- каша ++ лот = кашалот;  
соединяются две непустые строки и одна пустая — она располагается между двумя знаками «+».
- + =;  
соединяются две пустые строки. Результат — пустая строка.
- + по + беда + = победа;  
соединяются 4 строки. Первая и последняя соединяемые строки — пустые.

### Замечание

Предполагается, что среди символов, входящих в состав строк, знака «+» нет.

Будем считать, что на поле Кукарачи пустая строка не занимает клетку на поле, а располагается на стыке двух клеток.

### Задание

Во второй строке клетчатого поля исполнителя записано выражение, состоящее из строк, соединённых между собой операцией конкатенации «+». Некоторые из строк или даже все могут быть пустыми. Произвести указанные операции и получить результат. В начальный момент исполнитель расположен слева от записи. Пример начального и конечного состояний среды показан на рис. 4.1.



Рис. 4.1

## 2. Удаление символов (5 баллов).

Автор: А. А. Дуванов.

Введём операцию удаления символа, которую будем обозначать знаком «-». Эта операция будет удалять из строки символ, который предшествует знаку операции.

Например,

крош-кодил = крокодил

**Замечание 1**

Каждый знак «-» в строке рассматривается как операция, а не как символ строки.

**Замечание 2**

Знак операции «-» не может быть самым первым в строке.

**Замечание 3**

Операции «-» не могут следовать в строке подряд, т. е. записи вида «по--беда» исключаются.

*Задание*

Во второй строке клетчатого поля исполнителя из кубиков составлена запись, в которой, возможно, присутствуют знаки операции «удаление символа». Выполнить присутствующие операции и получить результат. В начальный момент исполнитель расположен слева от записи. Пример начального и конечного состояний среды показан на рис. 4.2.



Рис. 4.2

## 3. Удаление символов-2 (10 баллов).

Автор: А. А. Дуванов.

Решить задачу 2, допуская следование нескольких знаков операций «-» подряд. Операции выполняются последовательно слева направо.

Предполагается, что для каждой операции «-» в записи действительно существует символ, подлежащий удалению, кроме знаков «-», начинающих запись.



Пусть, например, «1», «2», «3», «4» — это команды **ВВЕРХ**, **ВНИЗ**, **ВЛЕВО**, **ВПРАВО** исполнителя на клетчатом поле. Пусть условие «К» истинно, если в клетке растёт капуста, а условие «М» истинно, если в клетке растёт морковь. Тогда программа «11ЕК31» в более привычной записи выглядит так:

**ВВЕРХ**

**ВВЕРХ**

**ЕСЛИ КАПУСТА ТО ВЛЕВО**

**ВВЕРХ**

Исполнитель, выполняя эту программу, перемещается на две клетки вверх, затем проверяет, растёт ли в клетке капуста. Если растёт, он перемещается влево, а затем вверх. Если капуста не растёт, исполнитель поднимается вверх без смещения влево.

Программа «1ЕКЕМ23» переводится так:

**ВВЕРХ**

**ЕСЛИ КАПУСТА ТО ЕСЛИ МОРКОВЬ ТО ВНИЗ**

**ВЛЕВО**

Исполнитель смещается вверх и проверяет, растут ли в клетке капуста и морковь одновременно. Если это так, он смещается вниз. В любом случае исполнитель затем перемещается на одну клетку влево.

Формальное определение допускает программы типа «ЕКЕКЕК1», что в обозначенной ранее среде означает то же самое, что «ЕК1».

На самом деле, в некоторых средах программы «ЕКЕКЕК1» и «ЕК1» могут отличаться по смыслу.

Предположим, что проверка меняет среду. Например, каждые две проверки «К», совершённые над клеткой с капустой, губят её (исполнитель затаптывает капусту ногами), и капуста в клетке погибает.

Тогда программа «ЕК1» смещает исполнитель вверх из клетки с капустой (капуста в клетке остаётся), а программа «ЕКЕКЕК1» оставляет исполнитель на месте и ухудшает экологию клетки (капуста гибнет от затаптывания).

Таким образом, показано, что введённое выше определение программы может иметь смысл. Какой? Самый разный. Всё зависит от среды, в которой предполагается выполнение программы и алгоритмов работы команд и проверок.

Нас сейчас интересует чисто формальная сторона дела. Нужно написать лексический анализатор, при помощи которого Кукарача сможет проверить соответствие записи на своём поле введённому ранее определению 4.

### Задание

Напишите программу, которая проверяет, является ли запись на поле программой в смысле определения 4. В начальный момент исполнитель расположен перед кубиками с записью во второй строке:

■ запись

Если проверяемая запись — программа, установить исполнитель в конец записи:

запись ■

Если запись — не программа, поставить исполнитель под первым неверным символом:

запись

■

Пример начального и конечного состояний среды показан на рис. 4.4.



Рис. 4.4

## 5. Морковь и капуста-2 (8 баллов).

Автор: А. А. Дуванов.

### Определение 5

<программа> := <команда> | <программа> <команда>

<команда> := <простая> | <условная>

<простая> := 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

<условная> := <сокращенная> | <полная>

<сокращенная> := E <условие> <простая>

<полная> := <сокращенная> И <простая>

<условие> := K | M

Напишите программу, которая проверяет, является ли запись на поле программой в смысле определения 5. В начальный момент исполнитель расположен перед кубиками с записью во второй строке:

■ запись

Если проверяемая запись — программа, установить исполнитель в конец записи:

запись ■

Если запись — не программа, поставить исполнитель под первым неверным символом:

запись

■

Дополнительно к разработке лексического анализатора придумать среду, исполнитель, описать алгоритмы работы команд и проверок, ввести более содержательные обозначения лексем языка и записать несколько программ-примеров в формальной нотации определения 5 и новых эквивалентных обозначениях.

Пример начального и конечного состояний среды показан на рис. 4.5.



Рис. 4.5

## 6. Странные тексты (6 баллов).

Автор: А. А. Дуванов.

### Определение 6

<текст> := <слово>. |<слово><разделитель><текст>

<разделитель> := \_ | ,

<слово> := <буква> | <слово><буква>

<буква> := А | М

Напишите программу, которая проверяет, является ли запись на поле текстом в смысле определения 6. В начальный момент исполнитель расположен перед кубиками с записью во второй строке:

■ запись

Если проверяемая запись — текст, установить исполнитель в конец записи:

запись ■

Если запись — не текст, поставить исполнитель под первым неверным символом:

запись

■

Пример начального и конечного состояний среды показан на рис. 4.6.



Рис. 4.6

### 7. Странные тексты-2 (7 баллов).

Автор: А. А. Дуванов.

Измените определение задачи 6 таким образом, чтобы в новом определении:

- 1) текст всегда начинался словом, первая буква которого заглавная, т. е. текст обязан начинаться с буквы «М» или «А»;
- 2) внутри текста заглавных букв быть не должно;
- 3) слово не должно состоять из одной буквы.

Напишите программу, которая проверяет, является ли запись на поле текстом в смысле нового определения. В начальный момент исполнитель расположен перед кубиками с записью во второй строке:

■ запись

Если проверяемая запись — текст, установить исполнитель в конец записи:

запись ■

Если запись — не текст, поставить исполнитель под первым неверным символом:

запись

■

Пример начального и конечного состояний среды показан на рис. 4.7.



Рис. 4.7

## Кукарачья геометрия

Расстояние на поле Кукарача измеряет своими шагами. Так, расстояние между двумя соседними клетками по вертикали или по горизонтали равно 1. Ведь Кукараче нужно сделать ровно один шаг, чтобы пройти из одной такой клетки-точки в другую. А вот расстояние между двумя соседними клетками по диагонали равно 2! Такая странная геометрия.

Вообще говоря, из одной клетки в другую Кукарача может попасть самым причудливым образом. Когда мы говорим, что измеряем расстояние шагами исполнителя, то имеем в виду наименьшее число шагов из всех возможных, которое должен сделать исполнитель, чтобы попасть из одной клетки в другую.

### 8. Геометрическое место точек (4 балла).

Автор: А. А. Дуванов.

Изобразите на поле исполнителя при помощи кубиков с буквой O:

- 1) окружность радиуса 1;
- 2) круг радиуса 1;
- 3) окружность радиуса 2;
- 4) круг радиуса 2;
- 5) геометрическое место точек, удалённых на 1 от диагональной цепочки клеток, протянутой из клетки (1,1);
- 6) геометрическое место точек, удалённых на 2 от диагональной цепочки клеток, протянутой из клетки (1,1);
- 7) равносторонний треугольник со стороной, равной 2;
- 8) равносторонний треугольник со стороной, равной 3;
- 9) равносторонний треугольник со стороной, равной 4.

В следующих пяти задачах считается, что размер поля настолько большой, насколько это необходимо для решения, а край поля находится так далеко от места работы исполнителя, что его присутствие можно не учитывать.

### 9. Построение окружности (8 баллов).

Автор: А. А. Дуванов.

Кубики **a**, **b** и **c** образуют на поле равнобедренный треугольник (**bc = ba**, сторона **ca** вытянута по вертикали, точка **b** слева от неё). Исполнитель располагается на перпендикуляре, проведённом из вершины **b** на сторону **ac**, сразу за вершиной **b**. Точка **a** отстоит от точки **c** на два кукарачьих шага. На рис. 4.8 приводятся примеры возможных начальных установок.

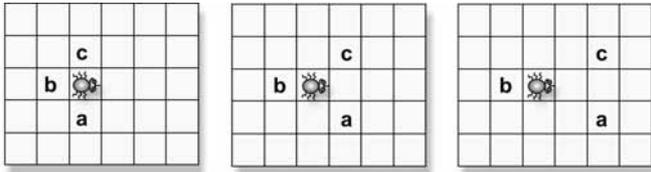


Рис. 4.8

Напишите программу, которая поместит точки **a** и **c** на окружность с центром в середине исходного отрезка **ac** и радиусом, равным длине перпендикуляра, опущенного из вершины треугольника **b** на сторону **ac**. После выполнения программы исполнитель должен располагаться в центре окружности (рис. 4.9).

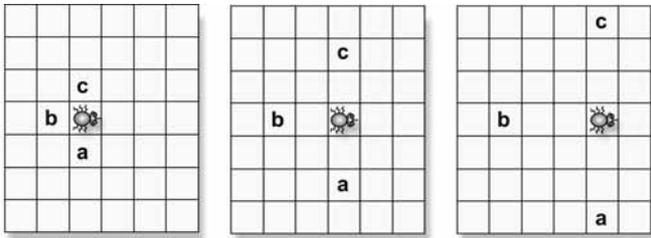


Рис. 4.9

### 10. Поворот треугольника (8 баллов).

Автор: А. А. Дуванов.

Три кубика на поле образуют равнобедренный треугольник с длиной основания, равной двум шагам исполнителя. Основание вытянуто по вертикали, третья вершина слева. Сам исполнитель располагается на высоте, проведённой к основанию, сразу за вершиной, из которой опущена высота. Повернуть треугольник на 90 градусов по часовой стрелке вокруг серединной точки основания и установить исполнителя так, чтобы он и три вершины повернутого треугольника образовывали ромб. Пример начального и конечного состояний среды показан на рис. 4.10.



Рис. 4.10

## 11. Смещение линейки (8 баллов).

Автор: А. А. Дуванов.

Справа от исполнителя кубики выстроились в линейку. Сместить линейку вниз на столько строк, сколько кубиков в линейке. Пример начального и конечного состояний среды показан на рис. 4.11.

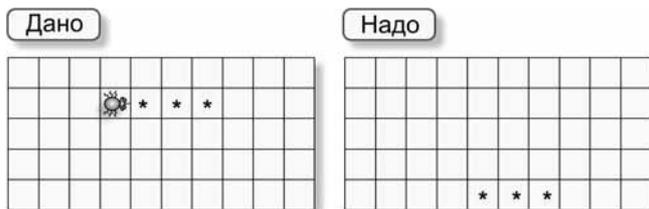


Рис. 4.11

## 12. Построение лесенки (8 баллов).

Автор: А. А. Дуванов.

Справа от исполнителя находится плотный ряд из нескольких кубиков. Построить из кубиков лесенку с высотой ступеньки в два шага и шириной — в один шаг Кукарачи. Сам исполнитель в конце работы изображает ещё одну, дополнительную ступеньку. Пример начального и конечного состояний среды показан на рис. 4.12.

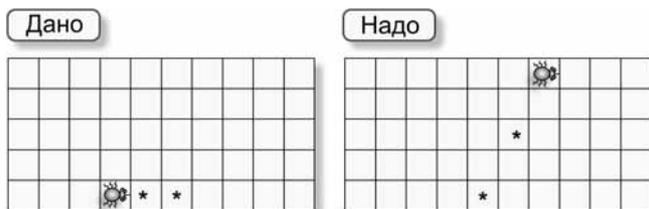


Рис. 4.12

## 13. Построение лесенки-2 (8 баллов).

Автор: А. А. Дуванов.

Решить предыдущую задачу при условии, что лесенка должна иметь ступени размером в один шаг исполнителя как по высоте, так и по ширине. Пример начального и конечного состояний среды показан на рис. 4.13.



Рис. 4.13

## 4.2. Турнир Корректора

И вот опять компьютеры скворчат,  
И стены сотрясаются от гула,  
И программисты пьют горячий чай  
С приятною улыбкой Вельзевула.

*Алексей Морозов*

Будем считать, что лента Корректора достаточно длинная и краевые эффекты в решениях можно не учитывать.

Для вычислений рекомендуется использовать ящик исполнителя, а в качестве чисел — алфавитные символы, считая, что символ изображает число, равное порядковому номеру символа в алфавите Корректора по отношению к символу «0». Порядковый номер символа «0» полагается равным 0.

Можно считать, что данные на ленте устроены так, что при вычислениях не возникает аварийных ситуаций: результат или промежуточные значения не превышают по величине порядковый номер последнего алфавитного символа и не отрицательны.

### 1. Популярная цифра (5 баллов).

Автор: А. А. Дуванов.

На ленте записана последовательность нулей и единиц. Выяснить, какая цифра встречается чаще. Разместить её справа от записи числа через пустую клетку. Если число нулей и единиц одинаково, в качестве ответа записать символ «=». В начальный момент окошко расположено перед записью числа. Пример начального и конечного состояний среды показан на рис. 4.14.

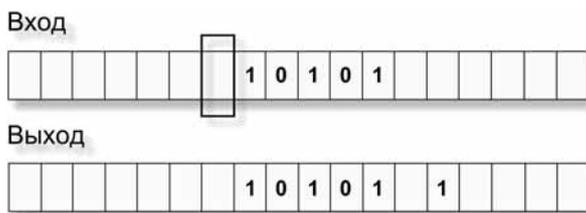


Рис. 4.14

### 2. Пересечение записей (8 баллов).

Автор: А. А. Дуванов.

Две записи на ленте разделяет пустая ячейка. Сформировать третью запись из символов, которые одновременно входят и в первую, и во вторую записи. В начальный момент окошко расположено перед первой записью.

Пример начального и конечного состояний среды показан на рис. 4.15.



Рис. 4.15

### 3. Сортировка записи (5 баллов).

Автор: А. А. Дуванов.

На ленте записана последовательность из цифр 0, 1 и 2. Переставить члены последовательности так, чтобы сначала располагались все нули, затем все единицы и, наконец, все двойки. В начальный момент окошко расположено перед записью. Пример начального и конечного состояний среды показан на рис. 4.16.

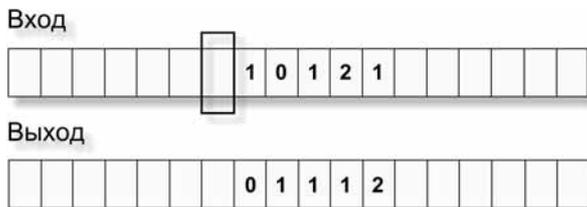


Рис. 4.16

### 4. Одинаковые символы (5 баллов).

Автор: А. А. Дуванов.

На ленте записана последовательность символов. Известно, что среди них только два одинаковых. Установить окно на первый из этих символов. В начальный момент окошко расположено перед записью. Пример начального и конечного состояний среды показан на рис. 4.17.

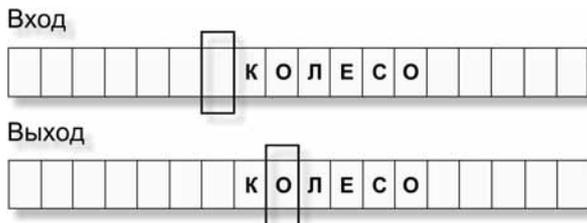


Рис. 4.17

### 5. Автомат (10 баллов).

Автор: А. А. Дуванов.

Построить в среде Корректора новый исполнитель Автомат. Автомат имеет ячейку памяти, которая изображается ящиком Корректора.

В начальный момент в ячейке памяти Автомата записано число 0, окно установлено слева от программы, которая представляет собой последовательность символов-команд «+» и «\*». Автомат выполняет последовательно команды программы и останавливается, когда программа заканчивается.

Смысл команд:

- + — добавить к ячейке памяти 1;
- \* — умножить ячейку памяти на 2.

*Задание*

1. Построить исполнитель Автомат в среде Корректора.
2. Написать программу для Автомата (по возможности, короткую), выполняющую которую исполнитель получил бы в ячейке памяти число 61.
3. Придумать способ получения заданного числа  $n$  за наименьшее число команд для исполнителя Автомат.

#### **Замечание 1**

Автомат хранит число в ячейке памяти как символьное число Корректора.

#### **Замечание 2**

После окончания работы Автомата для большей наглядности, записывайте содержимое ячейки памяти Автомата на ленту в виде обычного десятичного числа.

#### **Замечание 3**

Если в программе для Автомата обнаружится ошибка, нужно написать об этом сообщение «ОШ» в конце программы на ленте, а окошко Корректора установить на место ошибки.

На рис. 4.18 показана программа, в результате работы которой Автомат получает число 11.

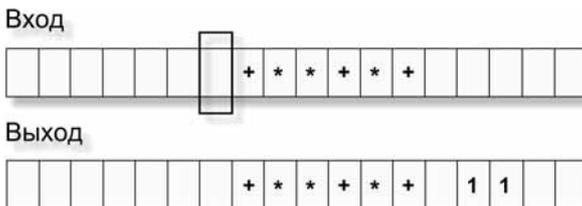


Рис. 4.18

## 6. Из двоичной в десятичную (10 баллов).

Автор: А. А. Дуванов.

На ленте Корректора записано целое положительное число в двоичной системе счисления. Перевести это число в десятичную систему. В начальный момент окошко установлено перед записью числа. Пример начального и конечного состояний среды показан на рис. 4.19.

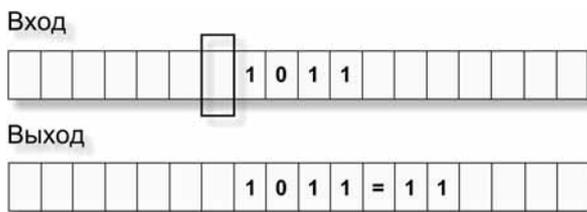


Рис. 4.19

## 7. Из римской в десятичную (10 баллов).

Автор: А. А. Дуванов.

На ленте Корректора записано натуральное число в римской системе счисления. Перевести это число в десятичную систему. Для обозначения цифр римской системы счисления использовать следующие обозначения (табл. 4.1).

Таблица 4.1

Римская цифра	Обозначение	Что означает
I	1	единица
V	5	пять
X	A	десять
L	(	пятьдесят

В начальный момент окошко установлено перед записью числа. Пример начального и конечного состояний среды показан на рис. 4.20.

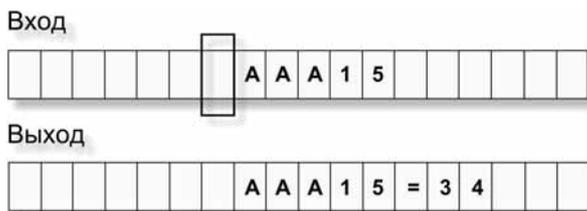


Рис. 4.20

## 8. Малыш (30 баллов).

Автор: А. А. Дуванов.

Построим в среде Корректора вычислительную машину Малыш.

Программу для этой машины будем записывать на ленту, а ящик использовать как регистр — специальную ячейку памяти.

Будем считать, что в алфавите Корректора:

ПУСТО 0 1 2 3 4 5 6 7 8 9 А В В Г Д Е Ж З И Й К Л  
 М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я - + / \*  
 = < > ( ) [ ] { } . , ! ? ; : ' " # | \$ % ~ @

символы пронумерованы, начиная с символа «0», а символ ПУСТО занимает особое положение и номера не имеет.

## Система команд Малыша

Каждая команда занимает две клетки на ленте Корректора.

### 1. Запись

3	z
---	---

*Пояснение.* Символ z записывается в ящик, стирая прежнее содержимое.

*Условное обозначение:* Я := z

*Пример*

В результате работы программы «3А» в ящике окажется символ «А».

### 2. Сложение

+	z
---	---

*Пояснение.* Символ, который получается сложением z с символьным числом из ящика, помещается в ящик, стирая прежнее содержимое.

*Условное обозначение:* Я := Я + z

*Пример*

В результате работы программы «3А+1» в ящике окажется символ «Б».

### 3. Вычитание

-	z
---	---

*Пояснение.* Символ, который получается вычитанием z от символьного числа из ящика, помещается в ящик, стирая прежнее содержимое.

*Условное обозначение:* Я := Я - z

*Пример*

В результате работы программы «3А-1» в ящике окажется символ «9».

#### 4. Повторение 1

П	z
---	---

*Пояснение.* Следующая команда выполняется z раз.

*Условное обозначение:* ПОВТОРИ z {команда}

##### **Замечание**

В программе команда повторения не может следовать подряд несколько раз, т. е. программа, в которой, например, встречается запись «П5П5», считается неверной.

*Пример*

В результате работы программы «32П2+3» в ящике окажется символ «8».

#### 5. Повторение 2

Я	
---	--

*Пояснение.* Следующая команда выполняется столько раз, каков порядковый номер символа в ящике перед началом повторения.

*Условное обозначение:* ПОВТОРИ Я {команда}

##### **Замечание**

Команды повторения (первого и/или второго типа) не могут следовать в программе несколько раз подряд.

*Пример*

В результате работы программы «32Я+3 в ящике окажется символ «8».

## Алгоритм выполнения программы

В начальный момент Малыш записывает в ящик символ «0», затем выполняет программу последовательно слева направо. Когда выполнена последняя команда, Малыш печатает через пустую клетку от программы содержимое ящика в виде: «символ=числовое значение».

Например, если в ящике оказался символ В, Малыш напечатает: «В=12».

## Авост

При выполнении программы могут происходить аварийные ситуации, на которые Малыш реагирует авостом (АВарийным ОСТановом). При авосте Малыш немедленно прекращает выполнение программы. Корректор выводит на экран табло с сообщением «Не могу».

(Подумайте, когда у Малыша может случиться авост? Подготовьте специальные тесты для проверки авоста.)

## Задания

1. (7 баллов) Запишите определение программы для Малыша формулами Бэкуса-Наура, постройте синтаксическую диаграмму (или таблицу) переходов для построенного определения. Пустую ленту можно считать корректной программой.
2. (4 балла) Напишите программу синтаксического анализатора для проверки правильности записи программы Малыша. Если в записи обнаруживается ошибка, Корректор пишет в конце программы Малыша слово ОШ и устанавливает окошко на место обнаруженной ошибки. Если ошибок нет, окошко устанавливается перед началом программы Малыша.
3. (10 баллов) Напишите для Корректора программу-интерпретатор. Эта программа должна читать на ленте Корректора программу Малыша и выполнять её. Если выполнение программы заканчивается без авоста, Корректор должен справа от программы записать на ленту содержимое ящика в виде: «символ=числовое значение».
4. (1 балл) Объедините синтаксический анализатор и интерпретатор в одну программу: сначала должен работать анализатор, затем, если ошибок в записи программы нет — интерпретатор. Какой ответ получит Малыш, выполняя программу «+2ПА+1»? Проверьте ваш ответ на построенной модели.
5. (3 балла) Напишите для Малыша программы решения следующих задач, затем проверьте их работу на построенной модели.  
Вычислить:
  - а)  $x^2 + x$ , при  $x = 3, 4, 5$ ;
  - б)  $2x^2 + 1$ , при  $x = 3, 4, 5$ ;
  - в)  $x^2 + 2x$ , при  $x = 5, 6, 7$ .
6. (6 баллов) Придумайте для Малыша три новых задачи, напишите их решение, проверьте работу полученных программ на построенном трансляторе.

## Глава 5



# Турниры 2000/2001

## 5.1. Турнир Кукарачи

Но я раздумывал о том,  
Нельзя ли павиана  
Заставить в бубен бить хвостом  
И петь под фортепьяно.

*Л. Кэрролл*

### Построения на клетчатом поле

#### 1. Программируемое перемещение (3 балла).

Автор: А. А. Дуванов.

Исполнитель расположен в левом верхнем углу поля, клетке (1,1). Где-то в первом столбце расположен кубик с ненулевой цифрой, справа от которого ещё один кубик с символом (рис. 5.1).

	1	2	3	4	5	6	7	8	9	10
1										
2										
3	5	К								
4										

Рис. 5.1

Необходимо переместить кубик с символом вправо по горизонтали на число клеток, задаваемое цифрой на первом кубике. После выполнения перемещения символьного кубика остальные элементы среды должны вернуться в исходное состояние (рис. 5.2).

Дополнительные условия:

- 1) Поле исполнителя бесконечно вправо и вниз.
- 2) Надпись на символьном кубике может быть любой.

	1	2	3	4	5	6	7	8	9	10
1										
2										
3	5						к			
4										

Рис. 5.2

## 2. Деление пополам (5 баллов).

Автор: А. А. Дуванов.

На поле исполнителя из кубиков собран вертикальный столбик. Исполнитель расположен слева перед первым кубиком основания (рис. 5.3).

				8						
				7						
				6						
				5						
				4						
				3						
				2						
				1						

Рис. 5.3

Разделить столбик на два, не нарушая вертикального расположения кубиков (рис. 5.4).

				4	8					
				3	7					
				2	6					
				1	5					

Рис. 5.4

Дополнительные условия:

- 1) Поле исполнителя бесконечно по всем четырём направлениям.
- 2) Число кубиков в исходном столбике не меньше двух и чётно.
- 3) Надписи на кубиках могут быть любыми.

### 3. Ракета (5 баллов).

Автор: А. А. Дуванов.

На поле исполнителя из кубиков собран вертикальный столбик. Исполнитель расположен слева перед первым кубиком основания (рис. 5.5).

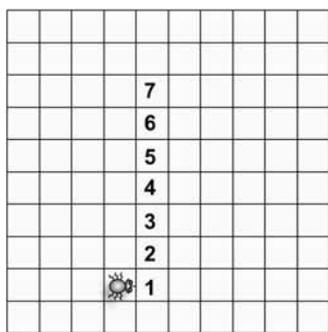


Рис. 5.5

Построить из кубиков «ракету» (рис. 5.6).

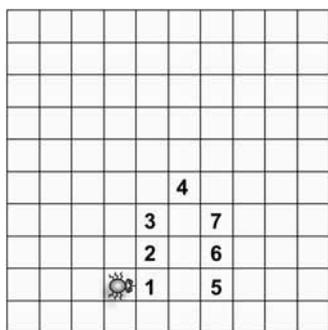


Рис. 5.6

Дополнительные условия:

- 1) Поле исполнителя бесконечно по всем четырём направлениям.
- 2) Число кубиков в исходном столбике — любое нечётное число, большее одного.
- 3) Надписи на кубиках могут быть любыми.

## 4. Опрокидывание столбика (8 баллов).

Автор: А. А. Дуванов.

На поле исполнителя из кубиков собран вертикальный столбик. Исполнитель расположен слева перед первым кубиком основания (рис. 5.7).

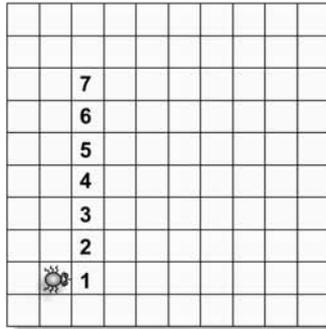


Рис. 5.7

«Опрокинуть» столбик на горизонталь так, чтобы кубик, расположенный в основании, остался на месте (рис. 5.8).

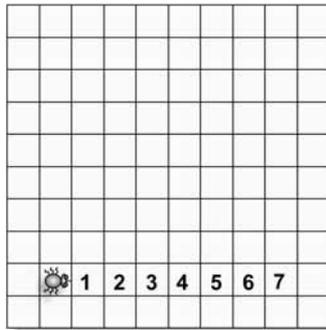


Рис. 5.8

Дополнительные условия:

- 1) Поле исполнителя бесконечно по всем четырём направлениям.
  - 2) Число кубиков в исходном столбике не меньше двух.
  - 3) Надписи на кубиках могут быть любыми.
5. Разобрать по брёвнышку (8 баллов).

Автор: А. А. Дуванов.

На поле исполнителя из кубиков собран прямоугольник. Исполнитель расположен слева перед первым кубиком основания (рис. 5.9).

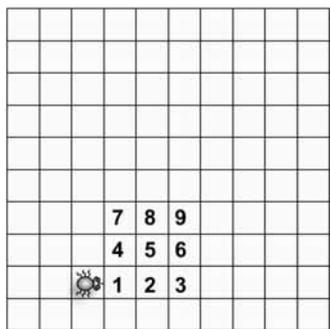


Рис. 5.9

Преобразовать прямоугольник в горизонтальный отрезок (рис. 5.10).

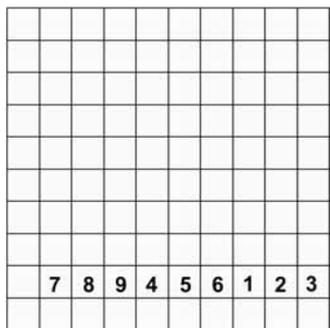


Рис. 5.10

Дополнительные условия:

- 1) Поле исполнителя бесконечно по всем четырём направлениям.
  - 2) Размеры прямоугольника — произвольные.
  - 3) Надписи на кубиках могут быть любыми.
6. Кирпич на другой столбик (7 баллов).

Автор: А. А. Дуванов.

На поле два столбика кирпичей. Снять верхний кирпич с первого столбика, положить на второй столбик и вернуться в исходное положение (слева от нижнего кирпича первого столбика). Столбики отстоят друг от друга на две клетки (рис. 5.11).

Дополнительные условия:

- 1) Поле исполнителя бесконечно вверх, вниз и вправо.
- 2) В первом столбике есть хотя бы один кирпич.

- 3) Во втором столбике кирпичи могут отсутствовать вовсе.  
 4) После выполнения задания столбики остаются на прежних местах.

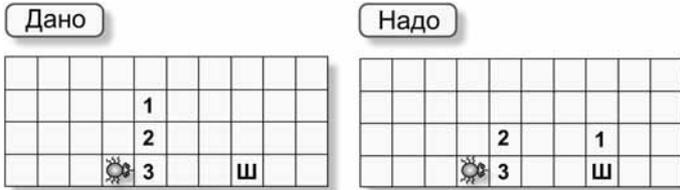


Рис. 5.11

## 7. Ханойская башня (15 баллов).

Автор: А. А. Дуванов.

Написать программу, которая решает на поле исполнителя задачу о Ханойской башне.

*Постановка задачи.*

Имеются 3 стержня и кольца разных размеров. В начальный момент все кольца нанизаны на первый стержень, при этом чем ниже кольцо, тем оно больше. Надо перенести все кольца с первого стержня на третий (второй стержень можно использовать как промежуточный).

Правила переноса:

- за один ход можно перемещать только одно кольцо;
- кольцо можно помещать либо на пустой стержень, либо на кольцо большего размера.

## Анализаторы и трансляторы

### 8. Проверка программы (6 баллов).

Автор: А. А. Дуванов.

#### Определение

`<программа>` := `<команда>` | `<программа><команда>`

`<команда>` := `<простая>` | `<цикл>`

`<простая>` := Л | П | В | Н

`<цикл>` := Ц `<повторитель>` `<простая>`

`<повторитель>` := 4 | 5

Проверить, является ли запись во второй строке поля программой. В начальный момент Кукарача расположен перед кубиками с записью:

■ запись

Если проверяемая запись — программа, установить Кукарачу в конец записи:

запись ▣

Если запись — не программа, поставить исполнитель под первым неверным символом:

запись

▣

## 9. Проверка программы-2 (6 баллов).

Автор: А. А. Дуванов.

### Определение

<программа> := <команда>|<программа><команда>

<команда> := <простая>|<цикл>

<простая> := Л|П|В|Н

<цикл> := Ц<повторитель><команда>

<повторитель> := 4|5

Проверить, является ли запись во второй строке поля программой. В начальный момент Кукарача расположен перед кубиками с записью:

▣запись

Если проверяемая запись — программа, установить Кукарачу в конец записи:

запись ▣

Если запись — не программа, поставить исполнитель под первым неверным символом:

запись

▣

## 10. Исполнитель Кроха (10 баллов).

Автор: А. А. Дуванов.

Кроха — это исполнитель, который умеет выполнять программы, записанные для него по следующему определению:

### Определение

<программа> := <команда>|<программа><команда>

<команда> := Л|П

Среда Крохи — пятая строка поля Кукарачи без первой клетки. Кроха отображается кубиком с надписью К. В начальный момент Кроха может быть в любом месте своей среды.

Напишите интерпретатор команд Крохи. Интерпретатор должен выполнять программы, написанные для Крохи. В начальный момент Кукарача расположен в клетке (3,1) перед программой (рис. 5.12).



Рис. 5.12

Смысл команд:

- Л — на одну клетку влево;
- П — на одну клетку вправо.

## 5.2. Турнир Корректора

На мой взгляд, суть программирования как ремесла заключается в создании и использовании уровней абстракции с целью навести мосты через пропасть, разделяющую физические возможности системы и желаемый логический результат. Эти зверски быстрые, но тупые, как пни, компьютеры постоянно нуждаются в подсказке, что и как нужно делать на следующем шаге, и поэтому становятся тяжёлым бременем для тех, кто их использует. В особенности — для программистов. Абстракция как раз и является тем самым золотым ключиком, который помогает справиться со всеми этими головоломными сложностями.

*Лу Гринзоу «Философия программирования»*

### Соглашения

Будем считать, что лента Корректора достаточно длинная, и краевые эффекты в решениях можно не учитывать.

Для вычислений рекомендуется использовать ящик исполнителя, а в качестве чисел — символьные числа Корректора. Можно считать, что данные на ленте устроены так, что при вычислениях не возникает аварийных ситуаций: результат и промежуточные значения не превышают по величине порядковый номер последнего алфавитного символа и не отрицательны.

## Рекуррентные последовательности

### 1. Последовательность-1 (2 балла).

Автор: А. А. Дуванов.

Первый член последовательности равен 1. Каждый следующий член получается приписыванием справа единицы к записи предыдущего.

$$a_1 = 1$$

$$a_{i+1} = a_i 1$$

Построить  $n$ -й член последовательности, если  $n$  кодируется порядковым номером символа, который Корректор видит в начальный момент в своём окошке. Считать, что символ «1» кодирует  $n = 1$ , и в начальный момент на ленте не могут быть записаны символы с меньшим порядковым номером. Пример начального и конечного состояний среды показан на рис. 5.13.

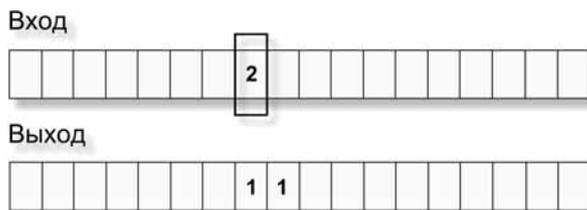


Рис. 5.13

### 2. Последовательность-2 (3 балла).

Автор: А. А. Дуванов.

Первый член последовательности равен 1. Каждый следующий член получается приписыванием справа 0, если последний символ записи предыдущего члена равен 1, или приписыванием справа 1 в противном случае.

$$a_1 = 1$$

$$a_2 = 10$$

$$a_3 = 101$$

...

Построить  $n$ -й член последовательности, если  $n$  кодируется порядковым номером символа, который Корректор видит в начальный момент в своём

окошке. Считать, что символ «1» кодирует  $n = 1$ , и в начальный момент на ленте не могут быть записаны символы с меньшим порядковым номером. Пример начального и конечного состояний среды показан на рис. 5.14.

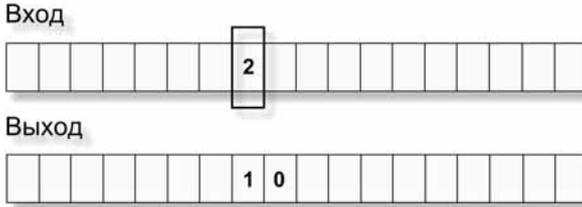


Рис. 5.14

### 3. Последовательность-3 (3 балла).

Автор: А. А. Дуванов.

Первый член последовательности равен 10. Каждый следующий член получается приписыванием справа двух символов, получаемых инвертированием последних двух знаков записи предыдущего члена.

$$a_1 = 10$$

$$a_2 = 1001$$

$$a_3 = 100110$$

...

Построить  $n$ -й член последовательности, если  $n$  кодируется порядковым номером символа, который Корректор видит в начальный момент в своём окошке. Считать, что символ «1» кодирует  $n = 1$ , и в начальный момент на ленте не могут быть записаны символы с меньшим порядковым номером. Пример начального и конечного состояний среды показан на рис. 5.15.

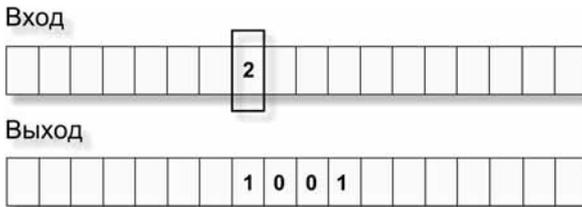


Рис. 5.15

### 4. Последовательность-4 (4 балла).

Автор: А. А. Дуванов.

Первый член последовательности представляет собой запись из нулей и единиц длины не меньше двух. Каждый следующий член получается



## 6. Последовательность-6 (6 баллов).

Автор: А. А. Дуванов.

Первый член последовательности представляет собой непустую последовательность нулей и единиц. Каждый следующий член получается приписыванием справа знака 0 или 1. Выбирается такой знак, чтобы число единиц и число нулей в каждом члене последовательности не отличались друг от друга больше, чем на 2.

Построить  $n$ -й член последовательности, если  $n$  кодируется символьным числом и  $n > 1$ . Сразу справа от  $n$  на ленте записан первый член последовательности.

## Лексический анализ выражений

## 7. Проверка программы (6 баллов).

Автор: А. А. Дуванов.

### Определение

`<программа>` := `<команда> | <программа><команда>`

`<команда>` := `<простая> | <цикл>`

`<простая>` := `Л | П | В | Н`

`<цикл>` := `Ц<повторитель><команда>`

`<повторитель>` := `4 | 5`

Проверить, является ли запись на ленте программой.

В начальный момент окно расположено слева перед записью. Если проверяемая запись — программа, написать следом за ней сообщение ОК.

Если запись — не программа, записать следом за ней сообщение ОШ и поставить окно на первый неверный символ.

## Классические алгоритмы

## 8. НОД двух чисел (7 баллов).

Автор: А. А. Дуванов.

Написать программу для вычисления наибольшего общего делителя двух положительных целых чисел. Полагать, что числа по величине меньше длины алфавита Корректора.

## 9. Из десятичной в двоичную (8 баллов).

Автор: А. А. Дуванов.

Написать программу для перевода десятичного целого неотрицательного числа в двоичную систему счисления. Полагать, что число по величине меньше длины алфавита Корректора.

10. Самое длинное слово (9 баллов).

Автор: А. А. Дуванов.

На ленте записан текст, состоящий из слов. Каждое слово завершает один пробел. Установить окошко Корректора на начало самого длинного слова. В начальный момент окошко расположено перед текстом. Пример начального и конечного состояний среды показан на рис. 5.18.

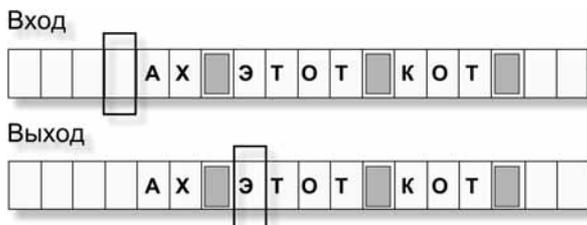


Рис. 5.18

11. Ханойская башня (10 баллов).

Автор: А. А. Дуванов.

Написать программу, которая выводит на ленту запись последовательности переносов, решающих задачу (см. описание задачи 7 в конкурсе Куркарачи).

## Глава 6



# Турниры 2001/2002

## 6.1. Турнир Кукарачи

Пусть простота проста,  
но хитра:  
до неё, как от зла  
до добра.  
В том секрет мастерства  
сих церквей,  
что добро проще зла,  
хоть трудней.

*А. Величанский*

## Вступительное слово куратора

Уважаемые коллеги! Придумывание конкурсных задач было моей особой заботой. Я изо всех сил стремился к высокой планке, установленной на курсе в 1997–2001 годах. Мне очень хотелось поддержать тот уровень красоты, простоты формулировок и сложности задач, который доставляет решившему задачу настоящее удовольствие.

В традициях курса «Азы программирования» предлагать одну «нерешаемую» задачу. В этот раз, к сожалению, задачи типа «Ханойской Башни» придумать не удалось.

Но есть одна задача, которую не решил (по крайней мере, сразу) автор курса. Я не называю её номер, чтобы участники не отложили эту задачу в сторону. Тем более, что нередко случаи, когда участники турнира находят решение лучше, проще и эффективнее авторского.

Первые две задачи разминочные. Думаю, их решат все.

Постарайтесь отшлифовать решения до блеска!

Задачи 4–6 навеяны прошлогодними «брёвнышками», которые Кукарача разбросал по полю. Теперь он решил навести порядок, распилить их и сложить по размеру.

В задачах 7–10 немного кубиков, но «игры» с ними не так уж просты.

Для решения задачи 11 можно использовать язык Бэкуса-Наура и построить таблицу переходов.

Хочу предостеречь новичков курса от двух ошибок, допущенных многими командами в прошлом году:

- 1) Частные решения. Это решение только для одного состояния среды исполнителя (расположения кубиков и Кукарачи). Все турнирные задачи предполагают составление программы, верно работающей *на множестве* исходных данных, *описанном в условии*.
- 2) Примеры исходных и конечных состояний среды — *всего лишь примеры!* Они не исключают других состояний (символов на кубиках, количества кубиков), *не противоречащих условию*. Текст условия имеет приоритет над примерами. Некоторые команды в прошлом году решили, что символы на кубиках обязаны быть такими же, как в приведённом примере. Это не так! Автор задачи подбирает примеры так, чтобы:

- с одной стороны, прояснить смысл задания;
- с другой стороны, не подсказывать путь к решению.

Бывает, что приведённый в условии пример сбивает с толку, причём совершенно без злого умысла автора задачи.

При решении и оформлении задач вам будет полезно знать основные критерии, по которым мы будем оценивать решения на перекрёстной проверке.

Вот они:

- 1) *Правильность*. Программа должна работать *при всех состояниях среды*, оговорённых в условии. Если это не так, решение неправильно.
- 2) *Эффективность*. Время выполнения программы желательно минимизировать. Рекурсивные вызовы, «съедающие» стековую память, должны применяться по необходимости. Цикл **пока** очень часто эффективнее рекурсии.
- 3) *Краткость и красота текста программы*. Короткая программа при прочих равных условиях лучше длинной, простая лучше сложной.
- 4) *Оформление*. Программа должна легко читаться, названия процедур — помогать понять алгоритм. Комментарии обязательно нужны, но содержательные и в меру.

Ну вот, кажется, и все советы, которые успеваю дать до начала турнира. Поверьте, очень переживаю за всех вас и желаю всем побед! Над задачами, над хитростями, над собой!

## 1. Ёлка (3 балла).

Автор: А. В. Рудь, Снежинск.

Кукарача находится в первом столбце. Справа от него вплотную стоит ряд из нечётного количества кубиков. На строку выше — плотный ряд кубиков, в котором на 2 кубика меньше, сдвинутый на один столбец вправо. В ряду на строку выше — ещё на 2 кубика меньше и т. д. Сооружение заканчивается одним кубиком и напоминает о приближении Нового года.

Кукарача хочет забраться на вершину ёлки и сиять там, словно звезда! Разрешается немного подвинуть ёлку вправо. Пример начального и конечного состояний среды показан на рис. 6.1.

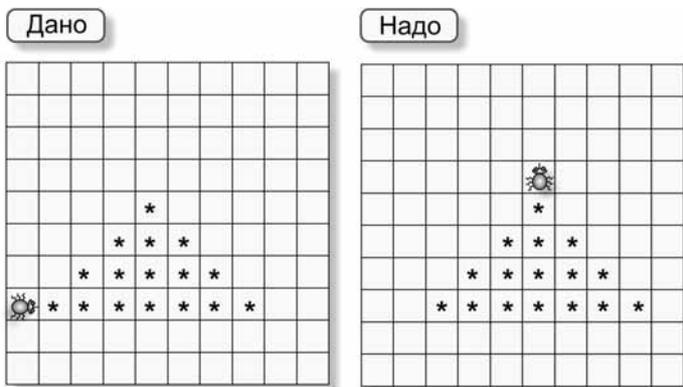


Рис. 6.1

## 2. Точка (4 балла).

Автор: А. В. Рудь, Снежинск.

В третьей строке, начиная со второго столбца, расположен плотный ряд кубиков, среди которых есть кубик с символом «.». Кукарача замер в клетке (1,3). Переставить точку на последнее место в записи и уплотнить ряд кубиков. Пример начального и конечного состояний среды показан на рис. 6.2.



Рис. 6.2

## 3. Сортировка (6 баллов).

Автор: А. В. Рудь, Снежинск.

В пятой строке вплотную стоят кубики, начиная со второго столбца. На каждом кубике либо 0, либо 1. Кукарача находится под первым слева кубиком. Требуется отсортировать кубики, т. е. переставить их так, чтобы все нули оказались слева, а единицы справа. Кубики должны оказаться в одной строке (любой) и стоять вплотную. Пример начального и конечного состояний среды показан на рис. 6.3.



Рис. 6.3

## 4. Распилить по метке (4 балла).

Автор: А. В. Рудь, Снежинск.

Среди плотно стоящих в одной строке кубиков есть единственный (не первый) кубик с символом «» — метка. Кукарача находится в этой же строке рядом с первым кубиком слева. Требуется передвинуть кубики, стоящие после метки, на строку выше остальных и выровнять ряды кубиков по левому краю. Пример начального и конечного состояний среды показан на рис. 6.4.

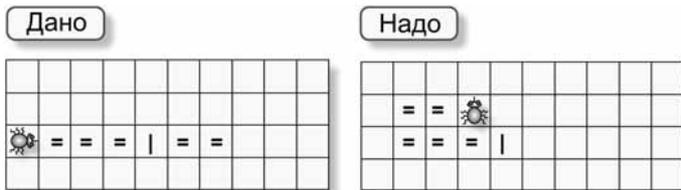


Рис. 6.4

## 5. Дрова на зиму (6 баллов).

Автор: А. В. Рудь, Снежинск.

Несколько кубиков стоят вплотную в строке, изображая «бревно». Кукарача под 3-м слева кубиком (или его местом, если количество кубиков меньше трёх). Требуется «распилить» бревно на части «длиной» по 2 ку-

бика и уложить в «поленницу» одно над другим. Возможно, последнее «полено» окажется короче. Поле Кукарачи считать бесконечным вправо и вверх. Пример начального и конечного состояний среды показан на рис. 6.5.

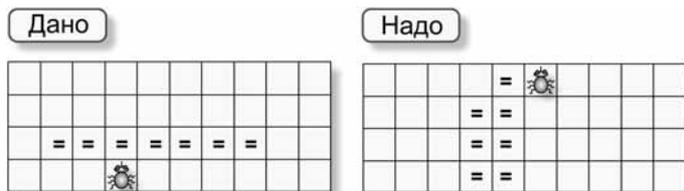


Рис. 6.5

6. Распилить натрое (8 баллов).

Автор: А. В. Рудь, Снежинск.

Начиная со второго столбца, в одной строке плотно стоят кубики, количество которых кратно 3. Кукарача под первым слева кубиком. Требуется «распилить бревно» на три равные части, т. е. расположить кубики в трёх соседних строках и выровнять по краям. Взаимное расположение кубиков в каждой части должно остаться неизменным. Поле Кукарачи считать бесконечным вверх и вправо. Пример начального и конечного состояний среды показан на рис. 6.6.



Рис. 6.6

7. Кукарача — перевозчик (8 баллов).

Автор: А. В. Рудь, Снежинск.

В третьем столбце (на «левом берегу реки») плотно стоят три перевёрнутых кубика с символами В (волк), К (коза) и П (капуста). Взаимное расположение кубиков неизвестно. Кукарача под нижним кубиком. Требуется передвинуть эти кубики в шестой столбец (на правый берег реки) по правилам роботландского Перевозчика: Кукарача может передвигать с берега на берег только по одному кубику, нельзя оставлять на одном берегу волка и козу, а также козу и капусту. Пример начального и конечного состояний среды показан на рис. 6.7.



Рис. 6.7

## 8. Обмен (6 баллов).

Автор: А. В. Рудь, Снежинск.

В третьей строке — два кубика. Кукарача под левым из них. Расстояние между кубиками неизвестно. Требуется поменять кубики местами. Пример начального и конечного состояний среды показан на рис. 6.8.



Рис. 6.8

## 9. Треугольник (9 баллов).

Автор: А. В. Рудь, Снежинск.

В третьей строке расположены два кубика. Третий кубик находится под правым в том же столбце. Расстояния между кубиками неизвестны. Требуется переставить кубики так, чтобы первый оказался на месте второго, второй — на месте третьего, а третий занял место первого. Поле считать бесконечным во все стороны. Пример начального и конечного состояний среды показан на рис. 6.9.

## 10. Проще некуда (10 баллов).

Автор: Катерина Алексеева, Снежинск.

Кукарача и кубик находятся в одной строке, кубик справа на неизвестном расстоянии. Требуется поменять их местами. Пример начального и конечного состояний среды показан на рис. 6.10.

## 11. Проверка записи (8 баллов).

Автор: А. А. Дуванов.

Запись состоит из нулей и единиц. В ней не должно быть более двух единиц, идущих подряд. Проверить правильность записи. Если запись

верная, установить исполнителя справа от неё, иначе — под первым неверным символом. В любом случае в конечном виде вся запись должна располагаться в одной строке. В начальный момент исполнитель стоит перед записью, вплотную к ней. Пример начального и конечного состояний среды показан на рис. 6.11.

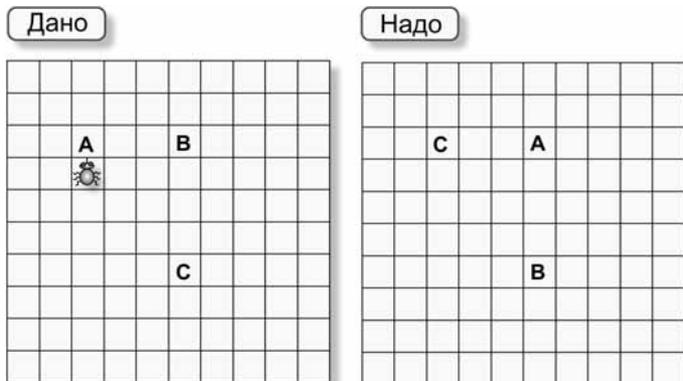


Рис. 6.9



Рис. 6.10



Рис. 6.11

## 6.2. Турнир Корректора

От малого большое зарождается.

*Русская поговорка*

Non multum, sed multa.

"Не много, но многое"

*Латинское крылатое выражение*

### Вступительное слово куратора

В конкурсе Корректора предлагается 12 задач. Первые пять — разминочные. Надеюсь, их решат все. Уверен, что решения будут очень разные. Будьте внимательны, найдите эффективные алгоритмы и тщательно протестируйте ваши программы, чтобы не потерять баллы по невнимательности. Остальные задачи — потруднее и пообъемнее — для любителей повозиться с Корректором как следует.

Для всех задач действуют следующие соглашения.

Будем считать, что лента Корректора достаточно длинная, и краевые эффекты в решениях можно не учитывать.

Для вычислений рекомендуется использовать ящик исполнителя, а в качестве чисел — алфавитные символы, считая, что символ изображает число, равное порядковому номеру этого символа в алфавите Корректора: символ «0» соответствует числу ноль, а символ «@» — числу 69. Символ пусто не соответствует никакому числу, кроме разве что «палочного» нуля. При этом можно считать, что данные на ленте устроены так, что при вычислениях не возникает аварийных ситуаций: результат или промежуточные значения не превышают по величине порядковый номер последнего алфавитного символа и не отрицательны.

#### 1. Сколько палочек? (2 балла).

Автор: А. В. Рудь, Снежинск.

На ленте записано палочное число. Окошко расположено слева от него. Написать справа от палочек знак « $\Leftarrow$ » и символ, соответствующий количеству палочек в записи. Пример начального и конечного состояний среды показан на рис. 6.12.

#### 2. Поиск максимума (3 балла).

Автор: А. В. Рудь, Снежинск.

Окошко установлено на первом символе записи. Удалить все символы с ленты, кроме одного: символа с наибольшим порядковым номером в алфавите Корректора. Пример начального и конечного состояний среды показан на рис. 6.13.

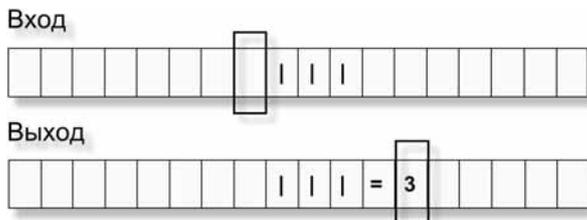


Рис. 6.12

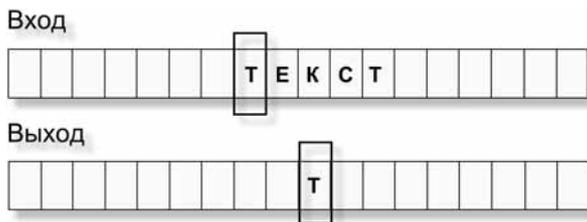


Рис. 6.13

3. Среднее арифметическое (5 баллов).

Автор: А. В. Рудь, Снежинск.

Найти среднее арифметическое двух символьных чисел, расположенных в соседних ячейках ленты. Если результат не целое число, то нужно записать два соседних символа, изображающие числа, наиболее близкие к результату. В начальный момент в окошко Корректора виден левый символ. Пример начального и конечного состояний среды показан на рис. 6.14.

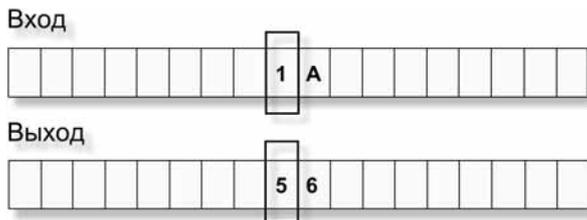


Рис. 6.14

4. Разгадай шифр-1 (6 баллов).

Автор: А. В. Рудь, Снежинск.

Первый символ записи на ленте виден в окошко Корректора. По приведённому примеру (рис. 6.15) разгадайте шифр и напишите программу для

дешифровки любой записи, зашифрованной с использованием этого шифра.

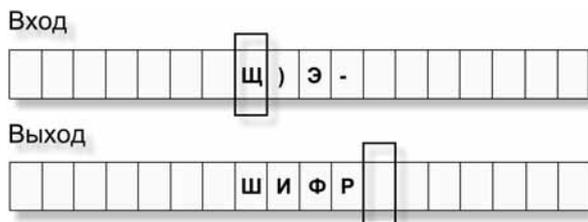


Рис. 6.15

5. Разгадай шифр-2 (5 баллов).

Автор: А. В. Рудь, Снежинск.

На ленте непонятная запись: «АЗАГКДА». В окошко Корректора виден первый слева символ. Разгадайте шифр без всяких примеров и напишите программу для дешифровки произвольной записи, зашифрованной с использованием этого шифра.

6. Умножение (7 баллов).

Автор: А. В. Рудь, Снежинск.

На ленте два символьных числа, и в окно Корректора видно первое из них. Записать на ленту результат умножения этих двух чисел в виде символьного числа. Пример начального и конечного состояний среды показан на рис. 6.16.

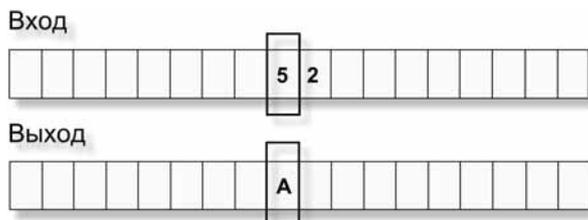


Рис. 6.16

7. Деление с остатком (7 баллов).

Автор: А. В. Рудь, Снежинск.

Пример на деление записан на ленту в виде «a/b» (a и b — символьные числа). Записать на ленту целую часть результата (в виде символьного числа), а остаток сохранить в ящике. В начальный момент в окно виден символ «/». Пример начального и конечного состояний среды показан на рис. 6.17.

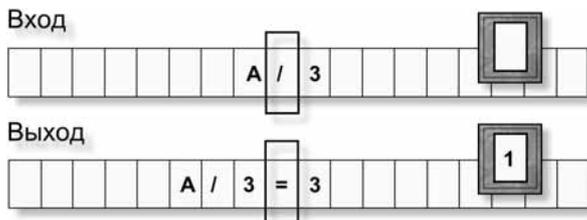


Рис. 6.17

## 8. Возведение в степень (9 баллов).

Автор: А. В. Рудь, Снежинск.

Задание на возведение числа в степень записано на ленту в виде « $a^b$ » ( $a$  и  $b$  — символьные числа). Записать результат в десятичном виде. В начальный момент в окошко Корректора виден символ « $\sim$ ». Пример начального и конечного состояний среды показан на рис. 6.18.

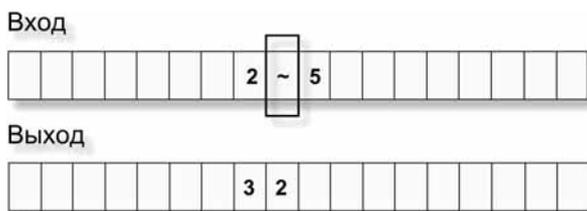


Рис. 6.18

## 9. Сократить дробь (10 баллов).

Автор: А. В. Рудь, Снежинск.

Правильная обыкновенная дробь записана на ленте в виде « $a/b$ » ( $a$  и  $b$  — символьные числа). Требуется, если возможно, сократить дробь. В начальный момент в окошко Корректора виден символ « $/$ ». Пример начального и конечного состояний среды показан на рис. 6.19.

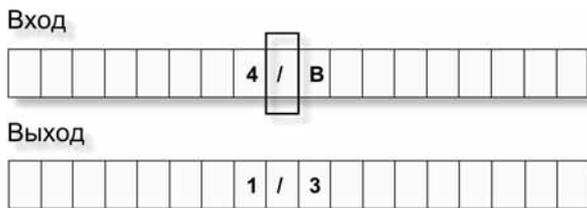


Рис. 6.19

## 10. Антиавтомат (8 баллов).

Авторы: А. В. Рудь и Лев Казанцев, Снежинск.

В системе команд Роботландского исполнителя Автомат всего 2 команды: + и \*. Среда Автомата — индикатор, который в начальном состоянии показывает число 0.

По команде + Автомат прибавляет к числу на индикаторе единицу; по команде \* Автомат умножает число на индикаторе на 2.

Например, выполняя программу «+\*\*\*+», Автомат получит число 10.

На ленте Корректора записан единственный символ, и он виден в окне. Символ этот изображает число, полученное Автоматом в результате выполнения некоторой программы. Требуется восстановить и записать на ленту самую короткую программу для получения этого числа. Пример начального и конечного состояний среды показан на рис. 6.20.

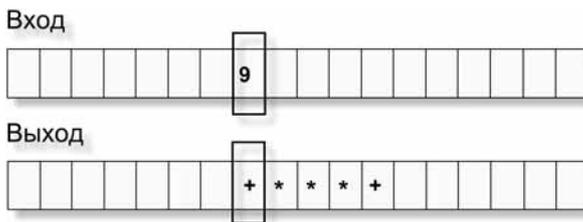


Рис. 6.20

## 11. Скобки (7 баллов).

Автор: А. В. Рудь, Снежинск.

Запись на ленте должна соответствовать следующему определению:

**Определение**

$\langle \text{запись} \rangle ::= \langle \text{пусто} \rangle | ( ) \langle \text{запись} \rangle | ( \langle \text{запись} \rangle )$

Проверить правильность записи. Если запись правильная, вывести после неё сообщение ОК, в противном случае написать ОШ и установить окно

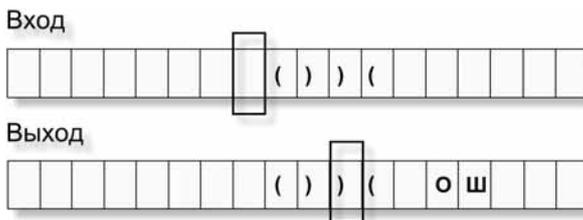


Рис. 6.21

на место ошибки. В начальный момент окошко Корректора установлено перед записью. Пример начального и конечного состояний среды показан на рис. 6.21.

12. Проверка выражения (12 баллов).

Автор: А. В. Рудь, Снежинск.

Выражение представляет собой запись примера на сложение со скобками.

При этом:

- в качестве чисел используются только цифры;
- вложенных скобок нет;
- в скобках может быть только сумма двух цифр;
- знак «+» используется только как бинарная операция, связывающая два слагаемых.

Примеры выражений:  $4$ ;  $7 + 2 + 3$ ;  $(4 + 6)$ ;  
 $8 + (3 + 5) + (0 + 3) + 2 + (5 + 8)$ .

- 1) Напишите определение описанного выражения в нотации Бэкуса-Наура (3 балла).
- 2) Составьте таблицу переходов (4 балла).
- 3) Напишите лексический анализатор, проверяющий, является ли запись на ленте выражением. Если запись — выражение, вычислите его значение (5 баллов).

В начальный момент окно расположено слева перед записью.

Если проверяемая запись — выражение, нужно записать следом за ней знак «=» и символ, изображающий значение выражения.

Если запись — не выражение, нужно записать следом за ней сообщение ОШ и поставить окно на первый неверный символ.

Пример начального и конечного состояний среды показан на рис. 6.22.

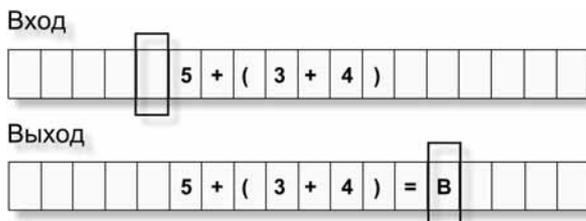


Рис. 6.22

# Глава 7



## Турниры 2002/2003

### 7.1. Турнир Кукарачи

Через несколько дней, вечером, Гена устроил маленькое совещание.

— Может, это не совсем тактично, что я хочу сказать, — начал он, — но всё-таки я скажу. Это мы просто здорово придумали! Но с тех пор как мы всё это здорово придумали, я потерял всякий покой! Даже ночью, когда все нормальные крокодилы спят, я должен вставать и принимать посетителей. Так продолжаться не может! Надо обязательно найти выход!

*Э. Успенский*

В приведённых далее задачах клетчатое поле исполнителя считается бесконечным, если только условие явно не оговаривает противное.

#### 1. Стыковка-1 (4 балла).

Автор: А. В. Рудь, Снежинск.

Кукарача в клетке (1,2). Во второй и третьей строках поля, начиная со второго столбца, плотно стоят два ряда кубиков. Требуется «сстыковать» ряды так, чтобы ряд из третьей строки стал продолжением ряда из второй строки. Взаимное расположение кубиков в исходных рядах должно остаться прежним. Пример начального и конечного состояний среды показан на рис. 7.1.



Рис. 7.1

## 2. Стыковка-2 (5 баллов).

Автор: А. В. Рудь, Снежинск.

Справа от Кукарачи (не обязательно рядом с ним) расположены два плотных непустых ряда кубиков. Между рядами одна или несколько пустых клеток. Требуется «склеить» ряды в один. Пример начального и конечного состояний среды показан на рис. 7.2.

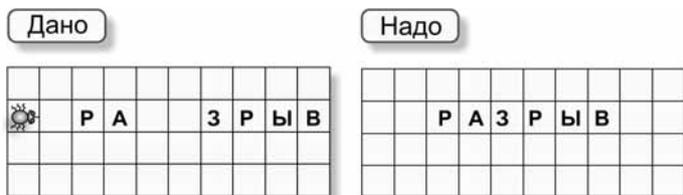


Рис. 7.2

## 3. Выбрать меньшее (6 баллов).

Автор: А. В. Рудь, Снежинск.

Кукарача в клетке (1,3). Во второй и четвёртой строках поля, начиная с первого столбца, плотно стоят два ряда кубиков. Требуется удалить с поля ряд большей длины. Если длины одинаковы, оставить на поле оба ряда. Взаимное расположение кубиков в рядах оставить неизменным. Пример начального и конечного состояний среды показан на рис. 7.3.



Рис. 7.3

## 4. Джип от джинна (6 баллов).

Авторы: Лебедев Александр и Игнатовский Олег, а также В. Ю. Матыкин, Новохопёрск.

Кукарача находится в клетке (2,2) и мечтает о добром джинне, который сделает ему сюрприз — подарит джип. Заветная буква П спрятана где-то на скрытом кубике в длинной лестнице универсамов (кубики У), начиная с 3 строки. Последний и предпоследний столбцы стандартного поля пусты.

Станьте добрым джинном! Сделайте сюрприз Кукараче! Подарите ему джип! Напишите программу, выполняя которую исполнитель отыщет букву П и поставит её на нужное место. Сдвинутые кубики на лестнице выравнивать не нужно. Пример начального и конечного состояний среды показан на рис. 7.4.



Рис. 7.4

5. Выстроить в ряд (7 баллов).

Автор: А. В. Рудь, Снежинск.

Кукарача находится в клетке (1,1). Во второй строке и следующих строках расположены плотные ряды кубиков, начинающиеся со 2-го столбца. Количество рядов и число кубиков в каждом ряду произвольно и не равно нулю. Требуется составить из кубиков один ряд. Взаимное расположение кубиков в конечном ряду не имеет значения. Пример начального и конечного состояний среды показан на рис. 7.5.



Рис. 7.5

## 6. Кукарача — грамотей (8 баллов).

Автор идеи: А. А. Кашников, Кострома.

Кукарача находится в первом столбце ниже третьей строки. Справа от него — непрерывный ряд кубиков, изображающий строку школьной тетради по русскому языку. Требуется проверить эту строку на выполнение правила «ЖИ-ШИ пиши с буквой И» и, если нужно, исправить ошибки, заменив кубики с буквой Ы, стоящие после кубиков Ж и Ш, на кубики с буквой И. Кубики с буквой Ы удалить с поля. Считать, что кубики с буквой И в достаточном количестве имеются над проверяемой строкой и стоят плотным рядом, начинающимся с того же столбца, что и проверяемая строка. Пример начального и конечного состояний среды показан на рис. 7.6.



Рис. 7.6

## 7. Поворот шахматной доски (9 баллов).

Автор идеи: В. П. Семенко, Рубцовск.

На поле Кукарачи из кубиков с цифрами 0 и 1 выложена «шахматная доска»  $2n \times 2n$  клеток ( $n$  — натуральное число). Исполнитель расположен под левым нижним кубиком, как на рис. 7.7. Требуется изменить расположение кубиков так, чтобы «доска» выглядела повернутой на 90 градусов. Пример начального и конечного состояний среды показан на рис. 7.7.

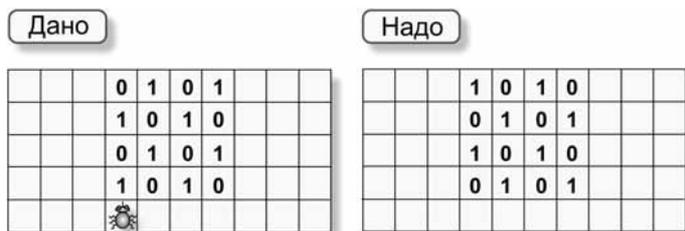


Рис. 7.7

## 8. Инверсия (10 баллов).

Автор: О. М. Федотов, Ижевск.

Кукарача находится в клетке (2,1). Справа от него вплотную расположены кубики с цифрами 0 и 1, и число нулей равно числу единиц. Требуется

«инвертировать» запись, т. е. переставить кубики так, чтобы на месте прежних нулей стояли единицы, а на месте прежних единиц — нули. Пример начального и конечного состояний среды показан на рис. 7.8.



Рис. 7.8

## 7.2. Турнир Корректора

На золотом крыльце сидели  
царь, царевич, король,  
королевич,  
сапожник, портной...  
Кто ты будешь такой?

*Детская считалочка*

### Вступительное слово куратора

Все задачи, кроме, возможно, одной или двух, могут показаться совсем простыми. Однако решение решению рознь. Постарайтесь найти самые рациональные алгоритмы, с минимальным количеством команд исполнителя. Во многих задачах возможны однопроходные решения, которые, надеюсь, доставят вам настоящее удовольствие.

Для всех задач действуют следующие соглашения:

1. Будем считать, что лента Корректора достаточно длинная, и краевые эффекты в решениях можно не учитывать. По этой же причине не предполагается использовать в решениях замкнутость ленты, обход вокруг ленты будет оцениваться как *очень нерациональный* прием.
2. Для «вычислений» рекомендуется использовать ящик исполнителя, а в качестве «чисел» — символные числа Корректора. При этом можно считать, что данные на ленте устроены так, что при вычислениях не возникает аварийных ситуаций: результат или промежуточные значения не превышают по величине порядковый номер последнего алфавитного символа и не отрицательны.
3. Символ **ПУСТО** (не путать с символом **ПРОБЕЛ**) не соответствует никакому числу, кроме, разве что, «палочного» нуля. Тем не менее, символом

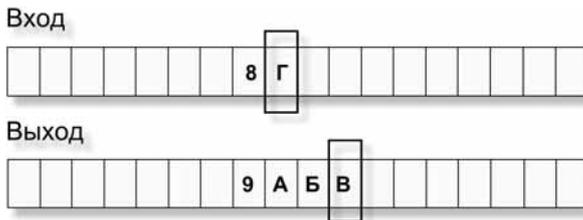
алфавита он является и может быть использован в исходных данных, если это не противоречит условию задачи. Этот символ с двух сторон ограничивает данные на ленте и может означать отсутствие данных, «пустую» запись.

*Куратор курса А. В. Рудь.*

1. От и до (3 балла).

Автор: А. В. Рудь, Снежинск.

На ленте в соседних ячейках записаны два символа из алфавита Корректора. Окошко установлено на правом из них. Требуется плотно записать на ленту символы, порядковые номера которых больше порядкового номера левого символа и одновременно меньше порядкового номера правого символа. Если таких символов нет, написать на ленте слово НЕТ. Исходные символы удалить с ленты. Пример начального и конечного состояний среды показан на рис. 7.9.

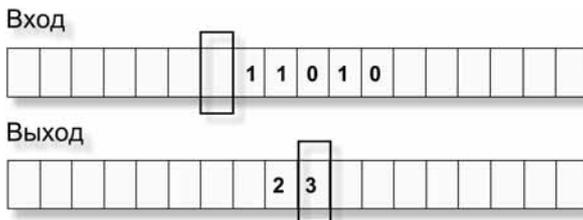


**Рис. 7.9**

2. Сколько нулей, сколько единиц (3 балла)?

Автор: А. В. Рудь, Снежинск.

На ленте запись, состоящая только из нулей и единиц. Окошко Корректора слева от записи. Написать на ленте вплотную два символа, изображающие количество нулей и единиц соответственно. Исходную запись с ленты удалить. Пример начального и конечного состояний среды показан на рис. 7.10.



**Рис. 7.10**

## 3. Счастливый билет (4 балла).

Автор: А. В. Рудь, Снежинск.

На ленте плотно записаны шесть десятичных цифр — номер троллейбусного билета. Окошко Корректора расположено перед записью. Билет считается счастливым, если сумма первых трёх цифр равна сумме трёх последних. Написать на ленте слово ДА, если билет счастливый, и слово НЕТ в противном случае. Исходную запись сохранять не нужно. Пример начального и конечного состояний среды показан на рис. 7.11.

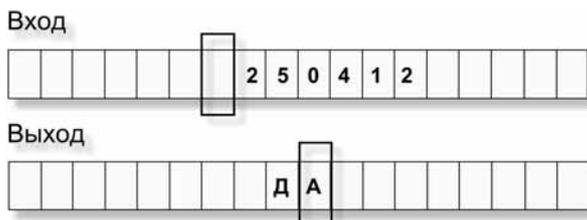


Рис. 7.11

## 4. Середина (5 баллов).

Автор: А. В. Рудь, Снежинск.

На ленте справа от окошка начинается непустая запись. Найдите её середину: если количество символов в записи нечётное, установите окно на средний символ, в противном случае вставьте в середину записи символ ПУСТО и установите окно на него. Пример начального и конечного состояний среды показан на рис. 7.12.



Рис. 7.12

## 5. Смешанное сложение (5 баллов).

Автор: А. В. Рудь, Снежинск.

На ленте последовательно записаны: палочное число, знак «+» и символическое число. Окно расположено перед записью. Получить результат

сложения в виде десятичного числа. Исходную запись удалить с ленты. Пример начального и конечного состояний среды показан на рис. 7.13.

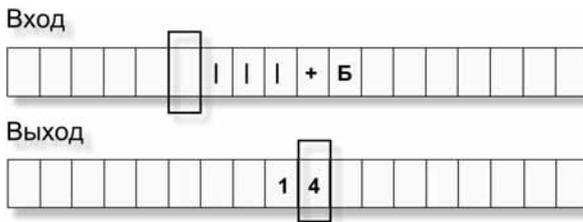


Рис. 7.13

6. Горка из чисел (5 баллов).

Автор: А. В. Рудь, Снежинск.

На ленте — единственный не пустой символ, и он виден в окне.

Требуется последовательно записать на ленту:

- 1) Все символы, порядковый номер которых меньше номера исходного символа в порядке возрастания.
- 2) Сам исходный символ.
- 3) Ряд, построенный в первом пункте, в порядке убывания.

После работы программы окно установить на исходный символ. Пример начального и конечного состояний среды показан на рис. 7.14.

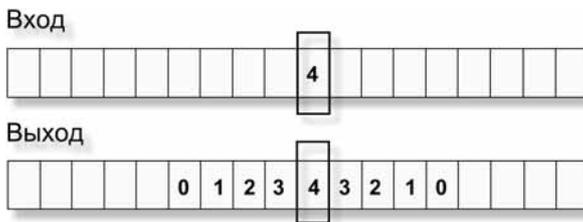


Рис. 7.14

7. Жребий на пальцах (6 баллов).

Автор: А. В. Рудь, Снежинск.

На ленте — два различных символа, записанные рядом. Они изображают количество пальцев на руке (от 0 до 5), которое наугад «выбросили» два игрока. Окно установлено на левом символе. Начиная с игрока, который «выбросил» меньше пальцев (назовём его первым), ведётся счёт от 1 до суммы «выброшенных» пальцев. То есть число 2 соответствует второму игроку, число 3 первому и т. д. Требуется установить окно на символе,

изображающем количество пальцев игрока, на котором закончится счёт. Пример начального и конечного состояний среды показан на рис. 7.15.

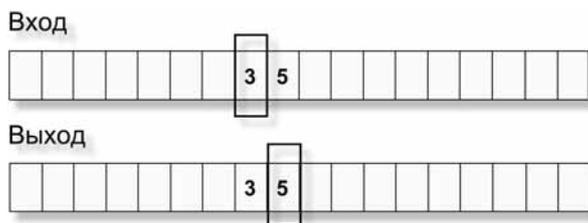


Рис. 7.15

### 8. Кому выходить (7 баллов)?

Автор: А. В. Рудь, Снежинск.

В круговой считалочке из  $m$  слов участвуют  $n$  человек. Все участники пронумерованы: 1-ый, 2-ой, ...,  $n$ -ый. На каком участнике закончится счёт, если начинать с первого? На ленте в соседних ячейках два символа: левый обозначает количество считающихся ( $n$ ), правый — количество слов в считалочке ( $m$ ). В окошко виден левый символ. После работы программы в окне должен быть виден символ, соответствующий номеру участника, на котором закончился счёт. Пример начального и конечного состояний среды показан на рис. 7.16.

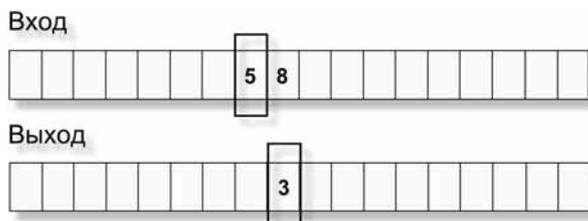


Рис. 7.16

### 9. Кто останется (10 баллов)?

Автор: А. В. Рудь, Снежинск.

В круговой считалочке из  $m$  слов участвуют  $n$  человек. Все участники пронумерованы: 1-ый, 2-ой, ...,  $n$ -ый. Считать начинают с первого. Участник, на котором закончился счёт, выбывает из круга. Оставшиеся участники повторяют считалочку, причём начинают считать с участника, следующего за выбывшим по кругу в направлении счёта. Если повторить считалочку  $n - 1$  раз, останется один участник. Требуется найти его номер.

На ленте два символа, записанных через одну пустую ячейку. Левый обозначает количество считающихся ( $n$ ,  $n > 0$ ), правый — количество слов в считалочке ( $m$ ,  $m > 0$ ). В окошко Корректора виден левый символ. После работы программы в окошке Корректора должен быть символ, соответствующий номеру участника, оставшегося в круге. Пример начального и конечного состояний среды показан на рис. 7.17.

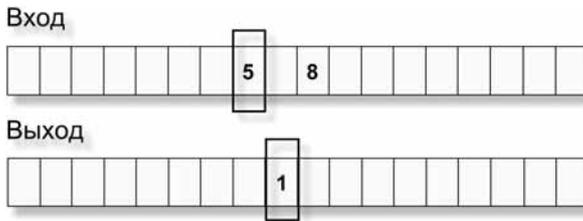


Рис. 7.17

#### 10. Умножение палочек (8 баллов).

Автор: Артем Букирь, Тольятти.

На ленте записаны два палочных числа, разделённые знаком умножения (символ «\*»). Знак умножения виден в окошке Корректора. Получить на ленте результат умножения в виде символического числа и в виде палочного числа, через знак равенства. Исходную запись на ленте можно не сохранять. Пример начального и конечного состояний среды показан на рис. 7.18.

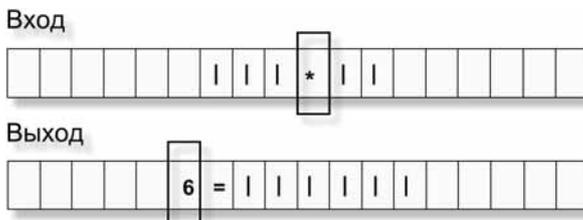


Рис. 7.18

#### 11. Хохотунчики (8 баллов).

Автор: А. В. Рудь, Снежинск.

В языке хохотунчиков все слова строятся только из трех слогов: ХА, ХИ и ХО. Слово может содержать любое количество слогов, например: ХАХА, ХИХИХО, ХИ. В записи любого хохотунского предложения принято разделять слова пробелами. Между словами могут быть запятые, которые пишутся слитно с левым словом. После запятой пробел

обязателен. Предложение должно заканчиваться восклицательным знаком, который записывается слитно с последним словом.

Посмотрите, как коротко и точно выглядит рекурсивное определение хохотунского предложения на языке Бэкуса-Наура:

```
<предложение> := <слово> ! | <слово> <пробел> <предложение> |
                <слово> , <пробел> <предложение>
<слово> := <слог> | <слог> <слово>
<слог> := X <гласная>
<гласная> := A | И | О
<пробел> ::= ■
```

Задания:

- 1) Составьте таблицу переходов для будущей программы (4 балла).
- 2) Запрограммируйте лексический анализатор, проверяющий, является ли запись на ленте хохотунским предложением (4 балла).

В начальный момент окно расположено слева перед записью (рис. 7.19).



Рис. 7.19

Если проверяемая запись — выражение, нужно написать следом за ней ОК (рис. 7.20).



Рис. 7.20

Если запись — не выражение, нужно записать следом за ней сообщение ОШ и поставить окно на первый неверный символ (рис. 7.21).



Рис. 7.21

# Глава 8



## Турниры 2003/2004

### 8.1. Турнир Кукарачи

— Ну и жарница тут, — сказал он.  
— Это я молодых петушков поджариваю, — сказала королева.  
— Славно, — сказал Дурень Ганс. — Так и мне заодно можно зажарить мою ворону?

*Г.-Х. Андерсен*

#### 1. Три мешка (3 балла).

Автор: А. В. Рудь, Снежинск.

Я вам не Крот, могу и по три мешка сразу таскать! — сказал Кукарача, но задумался: лабиринт какой-то узкий... Составьте программу для перетаскивания трёх мешков (М) из начального положения в конечное (как на рис. 8.1). Двигать стены лабиринта (Ж) нельзя. Поле исполнителя имеет размер  $10 \times 10$ .



Рис. 8.1

## 2. Разнеси подарки (3 балла).

Автор: А. В. Рудь, Снежинск.

Решил Кукарача Дедом Морозом поработать. Помогите ему разобрать две стопки подарков и разнести по комнатам как можно быстрее. На рис. 8.2: 01234, 56789 — стопки подарков. Поле исполнителя имеет размер  $10 \times 10$ . Двигать стены комнат (Ж) нельзя. Начальное и конечное состояния среды показаны на рис. 8.2.



Рис. 8.2

## 3. Первый и последний (4 балла).

Автор: А. В. Рудь, Снежинск.

Кукарача — в клетке (4,3). Справа от него без промежутка находится плотный непустой ряд кубиков неизвестной длины. Требуется поменять местами первый и последний кубики в ряду. При этом ряд и сам Кукарача должны остаться на своих местах. Пример начального и конечного состояний среды показан на рис. 8.3.



Рис. 8.3

## 4. Букет цветов (5 баллов).

Автор: И. В. Садовая, Тверь.

Лето. Солнце. Цветы (кубики Ц). Кукарача решил собрать букет. Четыре цветка у него уже есть, помогите добавить ещё три.

Один цветок расположен в 3-й строке правее 5 столбца. Другой — в 4 столбце ниже 5 строки, а третий — точно на главной диагонали ниже клетки (5,5). Начальное и конечное положение исполнителя в клетке (1,1). Поле считать бесконечным вправо и вниз. Пример начального и конечного состояний среды показан на рис. 8.4.



Рис. 8.4

## 5. Земельный участок (6 баллов).

Автор: В. П. Семенко, Рубцовск.

Кукарача купил земельный участок квадратной формы и оградил его кубиками. Стороны квадрата (они параллельны сторонам поля исполнителя) в Кукарачинске измеряются количеством пустых клеток между соответствующими вершинами. Например, квадрат со стороной 2 выглядит так (надписи на кубиках могут быть любыми):

А..В

....

....

Г..В

За заслуги перед Роботландией Кукараче позволили увеличить площадь участка в 4 раза. Как это сделать, если в начальный момент исполнитель находится в клетке (2,2), первая вершина в клетке (3,3), а сам квадрат (с ненулевой стороной) располагается правее и ниже исполнителя.

После выполнения программы первая вершина должна остаться на прежнем месте, а участок «подрасти» вправо и вниз. На рис. 8.5 показан пример начального и конечного состояний среды.



Рис. 8.5

#### 6. Горка-1 (6 баллов).

Автор: В. П. Семенко, Рубцовск.

Кукарача катается с симметричной двускатной горы (рис. 8.6.) то в одну, то в другую сторону. Горка крутая, но спуск — короткий!

Помогите Кукараче переделать горку на односкатную, увеличив тем самым длину спуска. Пример начального и конечного состояний среды показан на рис. 8.6. Кукарача расположен на вершине горки. Высота горки заранее неизвестна, но она не меньше двух рядов. Конечное положение исполнителя и новой горки неважно. Поле считается бесконечным во всех направлениях.



Рис. 8.6

## 7. А и Б (8 баллов).

Автор: А. В. Рудь, Снежинск.

Кукарача находится в клетке (4,3). Справа от него без промежутка находится плотный ряд кубиков неизвестной длины, в котором обязательно есть кубики с буквами А и Б, причём ровно по одному. Поменяйте местами кубики А и Б. Положение Кукарачи и ряда после выполнения программы неважно. Пример начального и конечного состояний среды показан на рис. 8.7.



Рис. 8.7

## 8. Найди друга (9 баллов).

Автор: В. П. Семенко, Рубцовск.

Однажды Корректор (К) гулял в парке и потерялся.

Парк имеет прямоугольную форму, стороны его параллельны сторонам поля Кукарачи, а ограда собрана из кубиков (любых, но среди них нет кубиков К).

Кукарача начал поиск друга с верхнего левого угла парка (рис. 8.8).



Рис. 8.8

Помогите найти Корректора и поставить его в ту клетку поля, с которой начался поиск. Размеры парка неизвестны, но известно, что его ограда не лежит в крайних столбцах и строках поля исполнителя. Конечное положение исполнителя и ограды неважно. Поле считать бесконечным вправо и вниз. На рис. 8.8 показан пример начального и конечного состояний среды.

9. Горка-2 (10 баллов).

Автор: В. П. Семенко, Рубцовск.

На поле исполнителя такая же горка, как и в задаче 6. Помогите Кукараче зеркально отразить горку относительно нижнего ряда кубиков. Высота горки неизвестна, но она не менее двух рядов. Положение исполнителя перед работой — вплотную слева от нижнего ряда кубиков. Поле считать бесконечным во всех направлениях. На рис. 8.9 показан пример начального и конечного состояний среды.

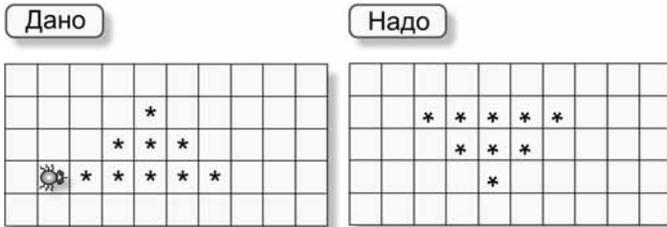


Рис. 8.9

10. Просто обмен (11 баллов).

Автор идеи: Денис Зуб, Снежинск.

Редактор условия: А. В. Рудь, Снежинск.



Рис. 8.10

На поле Кукарачи 4 кубика, по два в двух некрайних строках, причём два кубика расположены в первом столбце, а кубики в нижней строке расположены не в соседних клетках. Символы на кубиках могут быть любые. Кукарача находится под правым кубиком из нижней строки. Требуется поменять местами кубики, расположенные не в первом столбце (в приведённом примере это кубики Б и Г). Положение двух других кубиков и исполнителя после работы программы несущественно. Поле исполнителя не ограничено вправо. На рис. 8.10 показан пример начального и конечного состояния среды.

### 11. Чаше и меньше (12 баллов).

Автор: В. Ю. Матыкин, Новохопёрск.

Кукарача находится в клетке (2,1), а справа от него расположен плотный ряд кубиков с цифрами. Найти наиболее часто встречающуюся цифру и поставить её в первую строку. При наличии нескольких таких цифр выбрать наименьшую из них. Поле исполнителя не ограничено вправо и вниз. Пример начального и конечного состояний среды показан на рис. 8.11.



Рис. 8.11

## 8.2. Турнир Корректора

— Вот это я люблю, — сказала королева. — Ты за словом в карман не лезешь. Тебя я и возьму в мужья. Но знаешь, каждое слово, что мы сказали и скажем, записывают и печатают в утренней газете.

*Г.-Х. Андерсен*

### Определения

#### Определение 1. Символьное число

Под символьным числом будем понимать любой символ из алфавита Корректора, кроме символа пусто. Значением символьного числа является его

порядковый номер в алфавите Корректора по отношению к символу «0» (нуль), номер которого полагается равным числу 0.

Под арифметическими действиями над символьными числами будем понимать соответствующие действия над их алфавитными номерами.

### **Допущение 1. Длина алфавита**

Во всех задачах, если не оговорено обратное, считается, что алфавит Корректора бесконечный.

### **Допущение 2. Длина ленты**

Во всех задачах, если не оговорено обратное, считается, что лента Корректора бесконечная.

### **Определение 2. Отрезок**

Под отрезком на ленте будем понимать последовательность (слева направо) ячеек, в первой из которых записан символ Н (начало отрезка), в последней — К (конец отрезка), а промежуточные ячейки (тело отрезка) пусты. Число ячеек в теле отрезка будем называть его длиной. Отрезок нулевой длины занимает две ячейки на ленте:

НК

### **Определение 3. Запакованный отрезок**

Запакованный отрезок представляется на ленте тремя последовательными символами:

$H_nK$

Здесь:

1. Н — начало отрезка,  $n$  — символьное число, задающее длину отрезка;
2. К — конец запакованного отрезка.

### **Определение 4. Запись**

Под записью на ленте будем понимать конечную последовательность символов из алфавита Корректора, отличных от символа ПУСТО.

### **Определение 5. Слово**

Запись будем называть словом, если в ней нет символов ПРОБЕЛ.

1. Сложение отрезков (4 балла).

Автор: А. А. Дуванов.

На ленте последовательно (без промежутков) записано несколько запакованных отрезков. Написать программу, которая строит обычный незапа-

кованный отрезок, длина которого равна сумме длин исходных отрезков. В начальный момент окно установлено на первый символ записи. Пример начального и конечного состояний среды показан на рис. 8.12.

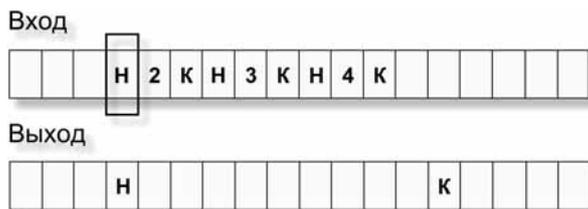


Рис. 8.12

## 2. Два отрезка (4 балла).

Автор: А. А. Дуванов.

Будем считать, что два отрезка имеют общую часть, если начало и (или) конец одного отрезка расположен между началом и концом другого отрезка. Длиной общей части назовём число пустых клеток, которые одновременно принадлежат обоим отрезкам, или число 0, если таких клеток нет.

Вычислить длину общей части двух отрезков на ленте. Ответ записать в виде символического числа. Начальное положение окна — на первом символе первого слева отрезка. Символы Н и К с ленты удалить. Примеры начального и конечного состояний среды показаны на рис. 8.13.

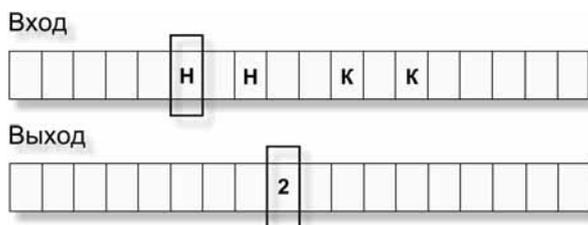


Рис. 8.13

## 3. Деление отрезка пополам (7 баллов).

Автор: А. А. Дуванов.

На ленте — обычный (незапакованный) отрезок ненулевой длины. Пометить середину отрезка символом «:», если длина отрезка нечётное число и двумя символами «:», если длина отрезка число чётное. Начальное положение окна — на первом символе отрезка. Примеры начального и конечного состояний среды показаны на рис. 8.14.

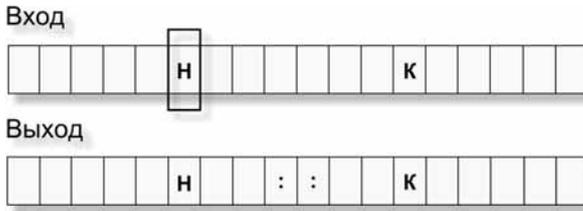


Рис. 8.14

## 4. Дата (4 балла).

Автор: Н. Д. Шумилина, Тверь.

Преобразовать дату, записанную в формате:

ЧислоНазваниеМесяцаГод

к виду:

Число.НомерМесяца.Год

В исходном и конечном состоянии среды окно установлено на первый символ записи. Пример начального и конечного состояний среды показан на рис. 8.15.

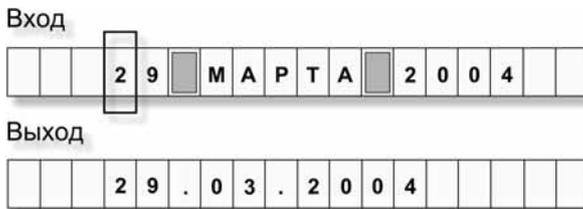


Рис. 8.15

## 5. Хитрое умножение (5 баллов).

Автор: Н. Д. Шумилина, Тверь.

Помогите Корректору вычислить произведение двух чисел, каждое из которых больше 96, но меньше 100. Запись исходного примера не сохранять. В исходном состоянии окно установлено на первый символ записи. Пример начального и конечного состояний среды показан на рис. 8.16.

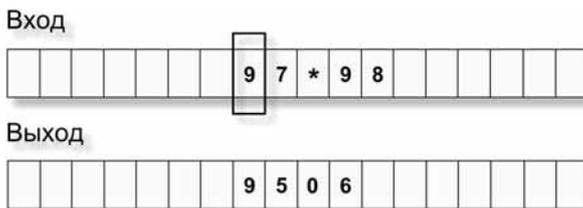


Рис. 8.16

## 6. Шифр-1 (6 баллов).

Автор: В. П. Семенко, Рубцовск.

На ленту записано зашифрованное слово. По приведённым примерам разгадайте шифр и напишите программу дешифровки. Начальное положение окна — на первом символе записи. Примеры начального и конечного состояний среды показаны на рис. 8.17.

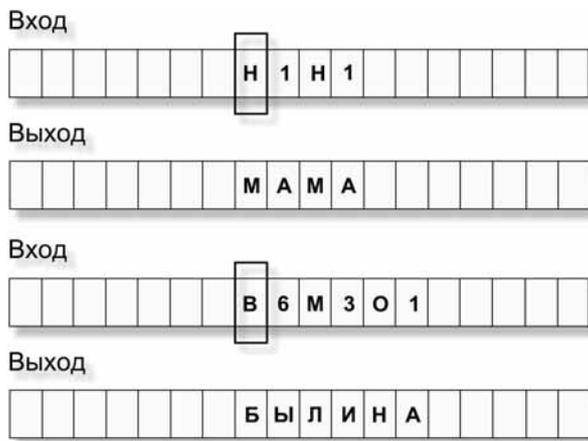


Рис. 8.17

## 7. Шифр-2 (8 баллов).

Автор: В. П. Семенко, Рубцовск.

На ленту записано зашифрованное слово. По приведённым примерам разгадайте шифр и напишите программу дешифровки. Начальное положение окна — на первом символе записи. Примеры начального и конечного состояний среды показаны на рис. 8.18.

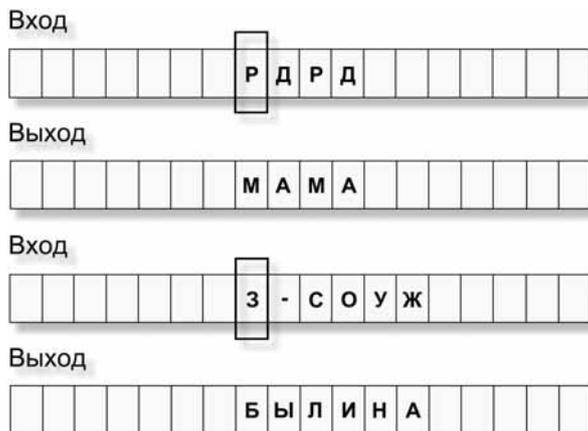


Рис. 8.18



## 10. Раздвоитель (8 баллов).

Автор: В. П. Семенко, Рубцовск.

Раздвоитель — это автомат, который получает на вход символьное число и работает с ним по следующему алгоритму:

**ПОКА** число не равно нулю, делать:

**ЕСЛИ** число чётное

**ТО** заменить его частным от деления на 2

**ИНАЧЕ** уменьшить число на 1

Создайте модель Раздвоителя на ленте Корректора. Модель должна работать над исходным символьным числом по описанному выше алгоритму и вычислять количество операций, выполненных Раздвоителем. Результат записывается на ленту в символьном виде. Пример начального и конечного состояний среды показан на рис. 8.21.

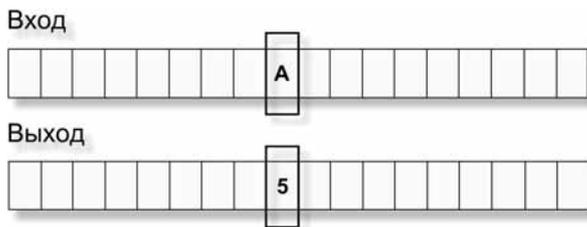


Рис. 8.21

## 11. Координаты вектора (5 баллов).

Автор: В. П. Семенко, Рубцовск.

Вектор задаётся на ленте символьными числами (возможно со знаком «-») в виде:

$(x_1, y_1)(x_2, y_2)$

Здесь:

- $x_1, y_1$  — координаты начала вектора;
- $x_2, y_2$  — координаты конца вектора.

Вектор коллинеарно сместили в начало координат. Найти новую координату его конца. Ответ записать в виде:

$(x_3, y_3)$

В начальный момент окно установлено на первый символ записи. Пример начального и конечного состояний среды показан на рис. 8.22.

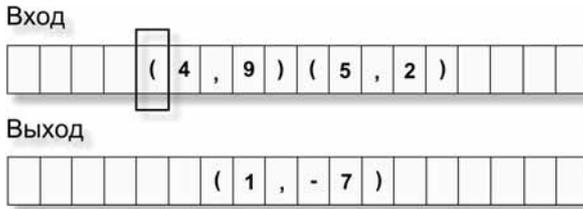


Рис. 8.22

## 12. Шахматное поле (9 баллов).

Автор: В. П. Семенко, Рубцовск.

Координаты шахматной клетки задаются символьными числами в виде:

$(x1,y1)$

Здесь:

- $x1$  — номер столбца (при счёте слева направо);
- $y1$  — номер строки (при счёте снизу вверх).

На ленту без промежутков записаны координаты двух шахматных клеток. Выяснить, являются ли эти клетки клетками одного цвета. Окно установлено на первый символ записи. Исходную запись на ленте можно не сохранять. Примеры начального и конечного состояний среды показаны на рис. 8.23.

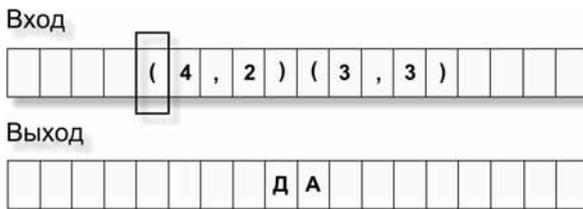


Рис. 8.23



## Глава 9

# Турниры 2004/2005

С 2004/2005 учебного года курс «Азы программирования» в Роботландском университете разделился на два:

□ Курс 31. Азы-программирования-1. Кукарача;

□ Курс 32. Азы-программирования-2. Корректор.

На момент сдачи книги в издательство на курсе 32 занятия ещё не проходили (предполагается, что учениками курса 32 станут выпускники курса 31), а на курсе 31 закончил работу первый семестр.

Темами семестра были линейные алгоритмы, алгоритмы с ветвлениями и циклами **повтори** и **пока**.

Предлагаемые далее задачи ориентированы на применение именно этих алгоритмических структур, без использования рекурсии.

## Олимпиада Кукарачи

— Какой в этом смысл? — спросил Кролик.

— Ну, — сказал Пух, — мы всё время ищем Дом и не находим его. Вот я и думаю, что если мы будем искать эту Яму, то мы её обязательно **не** найдём, потому что тогда мы, может быть, найдём то, чего мы как будто не ищем, а оно может оказаться тем, что мы на самом деле ищем.

— Не вижу в этом большого смысла, — сказал Кролик.

— Нет, — сказал Пух скромно, — его тут нет. Но он собирался тут быть, когда я начал говорить. Очевидно, с ним что-то случилось по дороге.

## 1. Квадратик (3 балла).

Автор: Н. Д. Шумилина, Тверь.

Кукарача находится в клетке (1,1). Помогите ему на стандартном поле  $10 \times 10$  выложить квадратик из 8 кубиков с символами «\*» так, как показано на рис. 9.1. Изначально кубики плотно стоят в 5 ряду, начиная с клетки (5,2).

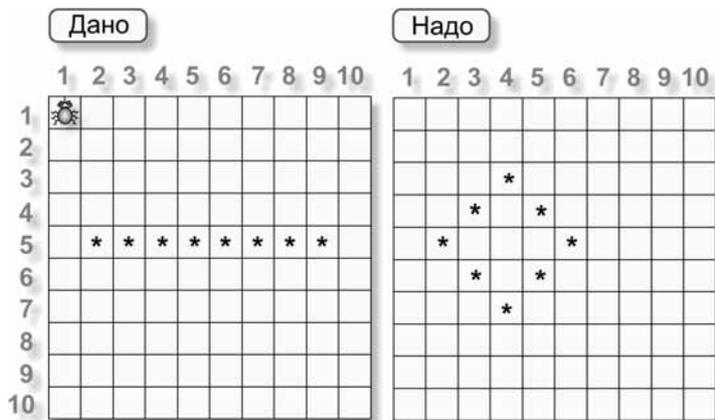


Рис. 9.1

## 2. Восстановление связи (2 балла).

Автор: Николай Глухих, Новохопёрск.

Между двумя телефонами НОКИА3310 и САМСУНГ нарушилась связь, обозначаемая символом «\*». Кукарача должен исправить название телефона и восстановить соединение. Поле размером  $10 \times 10$ . Исполнитель находится в клетке (1,1). Начальное и конечное положения срезы показаны на рис. 9.2.



Рис. 9.2

## 3. Хитрый гость (5 баллов).

Автор: В. П. Семенко, Рубцовск.

Как-то Корректор зашёл в гости к другу Кукараче. Не застав его дома, он решил занять себя и стал выкладывать из кубиков слова на поле размером  $10 \times 10$ . Корректор выкладывал слова парами по строчкам, причём второе слово каждой пары составлял из букв первого слова по одному и тому же правилу.

Не дождавшись друга, Корректор оставил записку: «Я составил на поле 6 пар слов, каждое слово из трёх букв. Второе слово каждой пары получается из первого по придуманному мною правилу. Потом я переставил некоторые кубики, и правильной осталась только одна пара. Разгадай моё правило и поставь кубики в нужном порядке!»

Помогите Кукараче решить задачу Корректора. Начальное состояние среды показано на рис. 9.3.

## 4. Крот в отпуске (4 балла).

Автор: В. П. Семенко, Рубцовск.

Мудрый Крот укатил в отпуск, оставив за себя Кукарачу. Нужно вытолкнуть все мешки (кубики с цифрами от 1 до 6) за пределы поля. Кукарача должен работать по правилам Крота, т. е. толкать впереди себя только по одному кубику и не сдвигать стены лабиринта (кубики с символом «\*»). Начальное состояние среды показано на рис. 9.4.

Дано

	1	2	3	4	5	6	7	8	9	10
1	🐛									
2	К	О	Т		Т	О	К			
3	К	О	Д		Д	Е	К			
4	П	А	С		С	И	П			
5	В	Е	С		С	У	В			
6	М	И	Р		Р	О	М			
7	К	У	Б		Б	А	К			
8										
9										
10										

Рис. 9.3

Дано

	1	2	3	4	5	6	7	8	9	10
1	*	*	*	*	*	*	*	*	*	*
2	*	*	*	*				*	*	*
3	*			*	3			*	*	*
4	*		4	*			2	*	*	*
5	*							1		*
6	*		*	*		*	*	*		*
7	*					5				
8	*	*	*	*	6	*	*	*		*
9	*	*	*	*					🐛	*
10	*	*	*	*	*	*	*	*	*	*

Рис. 9.4

## 5. Кукарача — цветовод (6 баллов).

Автор: В. П. Семенко, Рубцовск.

Кукарача решил посадить две клумбы с цветами в 5-ой и 7-ой строчках своего поля. На одной клумбе — нарциссы (кубики с буквой Н), а на

другой — тюльпаны (кубики с буквой Т). Всего по восемь цветов на каждой клумбе. Но семена перемешались, и на каждой клумбе выросло по четыре нарцисса и по четыре тюльпана в случайном порядке.

Кукараче захотелось пересадить цветы так, чтобы в пятой строке плотно росли нарциссы, а в седьмой плотно росли тюльпаны. Он хвастливо заявил Корректору, что выполнит эту работу с закрытыми глазами, независимо от того, в каком порядке выросли цветы на клумбах!

Клумбы начинаются со второго столбца стандартного поля  $10 \times 10$ , цветы на них растут плотно, а Кукарача в начальный момент находится в клетке (6,1). Пример начального и конечного состояний среды показан на рис. 9.5.



Рис. 9.5

6. Цифры по порядку (5 баллов).

Автор: В. П. Семенко, Рубцовск.

Кукарача находится в клетке (2,1). Начиная с клетки (2,2) расположен плотный ряд из восьми кубиков с цифрами 1, 2, 3 и 4. Цифры стоят произвольным образом, и каждая повторяется ровно два раза.

Требуется упорядочить цифровой ряд по возрастанию, т. е. сначала расположить единицы, за ними — двойки, затем — тройки и, наконец, — четвёрки. Упорядоченный ряд должен остаться на месте исходного. Размеры поля  $10 \times 10$ . Пример начального и конечного состояний среды показан на рис. 9.6.

7. Красные и синие (8 баллов).

Автор: Н. Д. Шумилина, Тверь.

Во второй строке, начиная с клетки (2,2), находится плотный ряд красных (К) кубиков, за которым без промежутков следует плотный ряд си-

них (С) кубиков. Количество кубиков каждого цвета произвольно, в том числе, возможно, равно нулю.

Поменяйте местами «красные» и «синие» ряды. Результат должен располагаться во второй строке, начальный столбец не важен. В начальный момент Кукарача находится в клетке (2,1). Поле считать бесконечным вправо. Пример начального и конечного состояний среды показан на рис. 9.7.



Рис. 9.6



Рис. 9.7

### 8. Репка (7 баллов).

Автор В. П. Семенко, Рубцовск.

Кукарача находится в клетке (2,1). Справа, в этой же строке расположены кубики с буквами Р (Репка), Д (Дед), Б (Бабка), В (Внучка), Ж (Жучка), К (Кошка), М (Мышка) в указанном порядке. Репка растёт не ближе девятого столбца. Остальные кубики отстоят друг от друга и от репки не менее, чем на две клетки.

Чтобы выдернуть репку, Кукарача должен собрать всех сказочных героев сразу слева от репки, и порядок их следования должен соответствовать сказке. Кто-то из персонажей может отсутствовать, но репка и мышка

присутствуют обязательно. Поле исполнителя считать бесконечным вправо. Пример начального и конечного состояний среды показан на рис. 9.8.

### 9. Лабиринт (5 баллов).

Авторы: Александр Емельянов и Павел Шевелев, Качканар.

На стандартном поле  $10 \times 10$  в 3-ей, 6-ой и 9-ой строках расположены по 9 кубиков (буква Н в первом столбце, остальные кубики с буквой С), образуя стенки лабиринта и проходы (в заранее неизвестных местах).

Помогите Кукараче пройти лабиринт из клетки (10,1) в клетку (1,1), восстановив положение сдвинутых кубиков. Пример начального и конечного состояний среды показан на рис. 9.9.

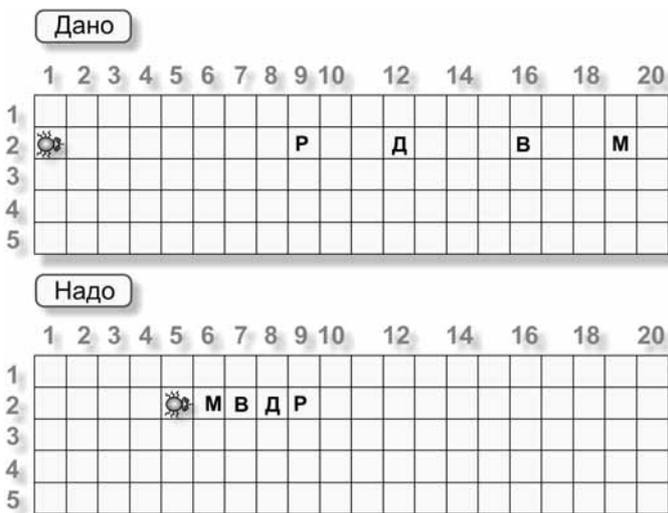


Рис. 9.8



Рис. 9.9

# Ссылки на задачи

Ниже представлена группировка задач по темам, изложенным в книге «Азы программирования. Магия для начинающих». Присутствуют ссылки только на задачи, предназначенные для самостоятельного решения. Названия задач роботландских конкурсов из книги «Азы программирования. Задачи роботландских турниров» (в табл. 1–3 ссылки на эту книгу обозначены ссылкой «Задачник») сопровождаются указанием сложности в баллах.

## Часть I. Кукарача

Таблица 1

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
1.	Молоток в моток	l.1	l.1.3.1	26
2.	Мишку в мышку	l.1	l.1.3.2	27
3.	Починить колесо	l.1	l.1.3.3	27
4.	Роль Кукарачи	l.1	l.1.3.4	27
5.	Имя знакомой кошки	l.1	l.1.3.5	27
6.	Что кричит кукушка	l.2	l.2.5.1	36
7.	Домашнее животное	l.2	l.2.5.2	36
8.	Из пяти ног одно пятно	l.2	l.2.5.3	36
9.	Кому поет Кукарача	l.2	l.2.5.4	37
10.	Исправление примера	l.2	l.2.5.5	37
	Лесенка	l.3	l.3.5.1	52

Таблица 1 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
11.	Прогулка по маршруту	l.3	l.3.5.2	53
12.	Контроль поля	l.3	l.3.5.3	53
13.	Пароход	l.3	l.3.5.4	53
14.	Колесо	l.3	l.3.5.5	53
15.	Машинист	l.3	l.3.5.6	54
16.	Очистить поле от мусора	l.3	l.3.5.7	54
17.	Прогулка	l.3	l.3.5.8	54
18.	Провернувшийся циферблат	l.3	l.7.14	104
19.	Цифровая головоломка	l.3	l.7.15	104
20.	Опять пятёрка	l.3	l.7.16	105
21.	Крот	l.3	l.7.17	105
22.	Квадрат	l.3	l.7.18	106
23.	Прогулка (3 балла)	l.3	Задачник 2.1.1	31
24.	Кубик в прямоугольнике (4 балла)	l.3	Задачник 3.1.1	46
25.	Три мешка (3 балла)	l.3	Задачник 8.1.1	125
26.	Разнеси подарки (3 балла)	l.3	Задачник 8.1.2	126
27.	Квадратик (3 балла)	l.3	Задачник 9.1	140
28.	Восстановление связи (2 балла)	l.3	Задачник 9.2	140
29.	Хитрый гость (5 баллов)	l.3	Задачник 9.3	141
30.	Крот в отпуске (4 балла)	l.3	Задачник 9.4	141
31.	Посадить дерево	l.4	l.4.1.2	60
32.	Игра	l.4	l.4.3.1	67
	Крестики-нолики	l.4	l.4.3.2	67
	Космическое путешествие	l.4	l.4.3.3	68
33.	Клад	l.4	l.4.3.4	68
34.	Медведь	l.4	l.4.3.5	68

Таблица 1 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
35.	Кот	1.4	1.4.3.6	69
36.	Экран	1.4	1.4.3.7	69
37.	Найти букву и встать на её место	1.4	1.4.3.8	69
38.	Кук	1.4	1.7.1	98
39.	Больной кот	1.4	1.7.2	98
40.	Шумы на линии	1.4	1.7.3	98
41.	Чаепитие в Кукарачинске	1.4	1.7.4	100
42.	Новогодние подарки	1.4	1.7.5	100
43.	Экран на прежнем месте	1.4	1.7.6	100
44.	Значение цифры (3 балла)	1.4	Задачник 1.7	26
45.	Умножение цифры на 11 (4 балла)	1.4	Задачник 2.1.6	34
46.	Кукарача — перевозчик (8 баллов)	1.4	Задачник 6.1.7	103
47.	Кукарача — цветовод (6 баллов)	1.4	Задачник 9.5	141
48.	Цифры по порядку (5 баллов)	1.4	Задачник 9.6	142
49.	Мяч в корзину	1.5	1.5.3.1	79
50.	Нарастить стену	1.5	1.5.3.2	79
	Провести по прямоугольнику А-Б-В-Г	1.5	1.5.3.3	80
51.	Ишак	1.5	1.5.3.4	80
	Мышь	1.5	1.5.3.5	80
52.	Упорядочить цифры	1.5	1.5.3.6	80
53.	Горка	1.5	1.7.7	101
54.	Какие оценки нужны Кукараче	1.5	1.7.8	101
55.	Последний станет первым	1.5	1.7.9	102
56.	Грибы	1.5	1.7.10	102

Таблица 1 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
57.	Грибы-2	l.5	l.7.11	102
58.	Камень, ножницы, бумага	l.5	l.7.19	106
59.	Белая мышка	l.5	l.7.20	106
60.	Уа-Ау	l.5	l.7.21	107
61.	Новогодняя сказка	l.5	l.7.22	108
62.	Машина времени	l.5	l.7.23	108
63.	Зелёный ряд	l.5	l.7.24	108
64.	Пример	l.5	l.7.25	110
65.	Торт	l.5	l.7.26	110
66.	Расставь цифры (4 балла)	l.5	Задачник 2.1.2	32
67.	Сложение двух палочных чисел (5 баллов)	l.5	Задачник 2.1.3	32
68.	Баскетбол (4 балла)	l.5	Задачник 2.1.8	35
69.	Умножение на 10	l.5	Задачник 2.1.9	35
70.	Ёлка (3 балла)	l.5	Задачник 6.1.1	101
	Точка (4 балла)	l.5	Задачник 6.1.2	101
71.	Стыковка-1 (4 балла)	l.5	Задачник 7.1.1	113
72.	Первый и последний (4 балла)	l.5	Задачник 8.1.3	126
	Красные и синие (8 баллов)	l.5	Задачник 9.7	142
73.	Репка (7 баллов)	l.5	Задачник 9.8	143
74.	Лабиринт (5 баллов)	l.5	Задачник 9.9	144
75.	Тигр	l.6	l.6.5.1	93
76.	Кубики на строку ниже	l.6	l.6.5.2	93
77.	Починить коридор	l.6	l.6.5.3	94
78.	Длина слова	l.6	l.6.5.4	94
79.	Умножение на 2	l.6	l.6.5.5	94
80.	Новый исполнитель	l.6	l.6.5.6	95
81.	Построение лесенки	l.6	l.6.5.7	95

Таблица 1 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
82.	Подсчитать число букв и очистить поле	l.6	l.6.5.8	95
83.	Грибы-3	l.6	l.7.12	103
84.	Циклическая перестановка	l.6	l.7.13	103
85.	Отрезки	l.6	l.7.27	111
86.	Деление на два (3 балла)	l.6	Задачник 1.1	23
87.	Треугольник (5 баллов)	l.6	Задачник 1.3	25
88.	Прямоугольник (6 баллов)	l.6	Задачник 1.4	25
89.	Вычитание двух палочных чисел (10 баллов)	l.6	Задачник 2.1.4	33
	Деление палочного числа на 2 (5 баллов)	l.6	Задачник 2.1.5	33
90.	Подсчёт букв О (8 баллов)	l.6	Задачник 2.1.7	35
91.	Разворот слова (6 баллов)	l.6	Задачник 2.1.10	37
	Количество согласных (8 баллов)	l.6	Задачник 2.1.11	37
92.	Середина отрезка (6 баллов)	l.6	Задачник 2.1.12	37
93.	Кубик с цифрой (5 баллов)	l.6	Задачник 3.1.2	47
94.	Уплотнение записи (7 баллов)	l.6	Задачник 3.1.3	47
95.	Квадрат (5 баллов)	l.6	Задачник 3.1.9	53
96.	Срединный перпендикуляр (6 баллов)	l.6	Задачник 3.1.10	53
97.	Вниз на Рав (8 баллов)	l.6	Задачник 3.1.11	54
98.	Прямоугольник (10 баллов)	l.6	Задачник 3.1.12	55
99.	Деление отрезка (6 баллов)	l.6	Задачник 3.1.13	55
100.	Сложение строк (5 баллов)	l.6	Задачник 4.1.1	67

Таблица 1 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
101.	Удаление символов (5 баллов)	l.6	Задачник 4.1.2	69
102.	Удаление символов-2 (10 баллов)	l.6	Задачник 4.1.3	69
103.	Построение окружности (8 баллов)	l.6	Задачник 4.1.8	75
104.	Поворот треугольника (8 баллов)	l.6	Задачник 4.1.10	76
	Смещение линейки (8 баллов)	l.6	Задачник 4.1.11	77
105.	Построение лесенки (8 баллов)	l.6	Задачник 4.1.12	77
106.	Построение лесенки-2 (8 баллов)	l.6	Задачник 4.1.13	77
107.	Программируемое перемещение (3 балла)	l.6	Задачник 5.1.1	85
	Деление пополам (5 баллов)	l.6	Задачник 5.1.2	86
108.	Ракета (5 баллов)	l.6	Задачник 5.1.3	87
109.	Опрокидывание столбика (8 баллов)	l.6	Задачник 5.1.4	88
110.	Разобрать по брёвнышку (8 баллов)	l.6	Задачник 5.1.5	88
111.	Кирпич на другой столбик (7 баллов)	l.6	Задачник 5.1.6	89
112.	Ханойская башня (15 баллов)	l.6	Задачник 5.1.7	90
113.	Сортировка (6 баллов)	l.6	Задачник 6.1.3	102
114.	Распилить по метке (4 балла)	l.6	Задачник 6.1.4	102
115.	Дрова на зиму (6 баллов)	l.6	Задачник 6.1.5	102
116.	Распилить натрое (8 баллов)	l.6	Задачник 6.1.6	103
117.	Обмен (6 баллов)	l.6	Задачник 6.1.8	104
118.	Треугольник (9 баллов)	l.6	Задачник 6.1.9	104
119.	Проще некуда (10 баллов)	l.6	Задачник 6.1.10	104

Таблица 1 (окончание)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
	Стыковка-2 (5 баллов)	l.6	Задачник 7.1.2	114
120.	Выбрать меньшее (6 баллов)	l.6	Задачник 7.1.3	114
121.	Джип от джина (6 баллов)	l.6	Задачник 7.1.4	114
122.	Выстроить в ряд (7 баллов)	l.6	Задачник 7.1.5	115
123.	Поворот шахматной доски (9 баллов)	l.6	Задачник 7.1.7	116
124.	Инверсия (10 баллов)	l.6	Задачник 7.1.8	116
125.	Букет цветов (5 баллов)	l.6	Задачник 8.1.4	127
	Земельный участок (6 баллов)	l.6	Задачник 8.1.5	127
126.	Горка-1 (6 баллов)	l.6	Задачник 8.1.6	128
127.	А и Б (8 баллов)	l.6	Задачник 8.1.7	129
128.	Найди друга (9 баллов)	l.6	Задачник 8.1.8	129
129.	Горка-2 (10 баллов)	l.6	Задачник 8.1.9	130
130.	Просто обмен (11 баллов)	l.6	Задачник 8.1.10	130
131.	Чаще и меньше (12 баллов)	l.6	Задачник 8.1.11	131

## Часть II. Корректор

Таблица 2

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
1.	От нечётного числа отнять единицу	ll.8	ll.8.3.1	<b>124</b>
2.	Удалить начальные пробелы	ll.9	ll.9.2.1	134

Таблица 2 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
3.	Удалить начальные и конечные пробелы	II.9	II.9.2.2	134
4.	Сумма единиц	II.9	II.9.2.3	135
5.	Сумма цифр	II.9	II.9.2.4	135
6.	Упорядочить два символа	II.9	II.9.3.3	140
7.	Двери	II.9	II.9.3.4	140
	Многоточия	II.9	II.9.4.1	141
8.	Проверить порядок	II.9	II.9.4.2	142
9.	Радиус окружности	II.9	II.15.7	246
10.	Распаковка отрезка	II.9	II.15.8	246
11.	Пятёрки	II.9	II.15.9	247
12.	Распиливание бревна	II.9	II.15.10	247
13.	ЧУ-ЩУ	II.9	II.15.11	248
14.	Подсчитаем брёвнышки	II.9	II.15.12	248
15.	Копия в обратном порядке (5 баллов)	II.9	Задачник 2.2.1	39
16.	Дополнение до 9 (4 балла)	II.9	Задачник 2.2.8	41
17.	Разметка отрезка (3 балла)	II.9	Задачник 3.2.11	64
18.	Последовательность-1 (2 балла)	II.9	Задачник 5.2.1	93
	Последовательность-2 (3 балла)	II.9	Задачник 5.2.2	93
19.	Последовательность-3 (3 балла)	II.9	Задачник 5.2.3	94
20.	Последовательность-4 (4 балла)	II.9	Задачник 5.2.4	94
21.	Шифр-1 (6 баллов)	II.9	Задачник 8.2.6	96
22.	Есть ли цифры в записи	II.10	II.10.5.1	158
23.	Уплотнение двух символов	II.10	II.10.5.2	159

Таблица 2 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
24.	Записать в порядке убывания	II.10	II.10.5.3	159
25.	Записать в порядке возрастания	II.10	II.10.5.4	159
26.	Запаковка отрезка	II.10	II.15.13	249
27.	Друзья	II.10	II.15.14	249
	Перестановка	II.10	II.15.15	249
28.	Левое зеркало	II.10	II.15.16	250
29.	Дефрагментация ленты	II.10	II.15.17	250
30.	Окно на первый символ записи	II.11	II.11.1.1	163
31.	Первое повторное вхождение символа	II.11	II.11.1.2	163
32.	Последнее вхождение символа	II.11	II.11.1.3	163
33.	ЗаклЮчить в круглые скобки	II.11	II.11.2.1	168
34.	Проверка записей на совпадение	II.11	II.11.2.2	168
	Продублировать в обратном порядке	II.11	II.11.4.1	175
35.	Проверка записи целого числа	II.11	II.11.4.2	175
36.	Какое число больше	II.11	II.11.4.3	175
37.	Убрать повторные вхождения символа	II.11	II.11.4.4	176
38.	Распаковка архива	II.11	II.11.4.5	176
39.	Симметричный отрезок	II.11	II.15.18	251
40.	Деление отрезка	II.11	II.15.19	251
41.	Среднее арифметическое	II.11	II.15.20	252
42.	Самое длинное бревно	II.11	II.15.21	252
43.	Палиндром (10 баллов)	II.11	Задачник 2.2.2	39
44.	Отношение чисел (10 баллов)	II.11	Задачник 2.2.3	40

Таблица 2 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
45.	Закраска окружности (3 балла)	II.11	Задачник 3.2.7	63
46.	Одинаковые символы (5 баллов)	II.11	Задачник 4.2.4	79
	Поиск максимума (3 балла)	II.11	Задачник 6.2.2	106
47.	Разгадай шифр-1 (6 баллов)	II.11	Задачник 6.2.4	107
48.	Разгадай шифр-2 (5 баллов)	II.11	Задачник 6.2.5	108
49.	Середина (5 баллов)	II.11	Задачник 7.2.4	119
50.	Дата (4 балла)	II.11	Задачник 8.2.4	134
51.	Вычитание двух чисел	II.12	II.12.1.1	188
52.	Умножить на 2	II.12	II.12.1.2	188
	Умножить на 4	II.12	II.12.1.3	189
53.	Умножить на 16	II.12	II.12.1.4	189
54.	Умножить на 3	II.12	II.12.1.5	189
55.	Умножить на 6	II.12	II.12.1.6	190
56.	Умножить палочное число на 2	II.12	II.12.2.1	196
57.	Умножить палочное число на 3	II.12	II.12.2.2	196
58.	Умножить палочное число на 6	II.12	II.12.2.3	196
59.	Отношение двух палочных чисел	II.12	II.12.2.4	197
60.	Первые палочные числа	II.12	II.12.2.5	197
61.	Первые чётные палочные числа	II.12	II.12.2.6	197
62.	Первые нечётные палочные числа	II.12	II.12.2.7	197
63.	Сумма символов	II.12	II.12.3.1	203
64.	Особые цифры	II.12	II.12.3.2	203
65.	Шифровка	II.12	II.12.3.3	204

Таблица 2 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
66.	Вычитание символов	II.12	II.12.3.4	204
67.	Умножение символов	II.12	II.12.3.5	204
	Точка	II.12	II.15.1	243
68.	Повтор	II.12	II.15.2	244
69.	Неисправная клавиатура	II.12	II.15.3	244
70.	Складывай и вычитай	II.12	II.15.22	253
71.	Неправильная дробь	II.12	II.15.23	254
	Возведение в квадрат	II.12	II.15.24	254
72.	Округление	II.12	II.15.25	255
73.	Умножение на пять	II.12	II.15.26	255
74.	Деление на 2 (7 баллов)	II.12	Задачник 2.2.4	40
75.	Пример на сложение (7 баллов)	II.12	Задачник 2.2.5	40
76.	Деление палочками (6 баллов)	II.12	Задачник 2.2.11	43
77.	Закраска пересечения окружностей (8 баллов)	II.12	Задачник 3.2.8	63
78.	Последовательность Фибоначчи (6 баллов)	II.12	Задачник 3.2.9	63
79.	Последовательность Фибоначчи-2 (8 баллов)	II.12	Задачник 3.2.10	64
80.	Построение отрезка (5 баллов)	II.12	Задачник 3.2.12	65
81.	Автомат (10 баллов)	II.12	Задачник 4.2.5	80
82.	Последовательность-5 (4 балла)	II.12	Задачник 5.2.5	95
83.	Последовательность-6 (6 баллов)	II.12	Задачник 5.2.6	96
84.	НОД двух чисел (7 баллов)	II.12	Задачник 5.2.8	96
85.	Сколько палочек? (2 балла)	II.12	Задачник 6.2.1	106

Таблица 2 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
86.	Умножение (7 баллов)	II.12	Задачник 6.2.6	108
	От и до (3 балла)	II.12	Задачник 7.2.1	118
87.	Сколько нулей, сколько единиц? (3 балла)	II.12	Задачник 7.2.2	118
	Счастливый билет (4 балла)	II.12	Задачник 7.2.3	119
88.	Смешанное сложение (5 баллов)	II.12	Задачник 7.2.5	119
89.	Горка из чисел (5 баллов)	II.12	Задачник 7.2.6	120
90.	Умножение палочек (8 баллов)	II.12	Задачник 7.2.10	122
91.	Сложение отрезков (4 балла)	II.12	Задачник 8.2.1	132
92.	Два отрезка (4 балла)	II.12	Задачник 8.2.2	133
93.	Деление отрезка пополам (7 баллов)	II.12	Задачник 8.2.3	133
94.	Хитрое умножение (5 баллов)	II.12	Задачник 8.2.5	134
95.	Число букв О в тексте	II.13	II.13.1.1	213
96.	Сумма наименьшей и наибольшей цифр числа	II.13	II.13.1.2	213
97.	Число слов в тексте	II.13	II.13.1.3	213
98.	Убрать лишние пробелы	II.13	II.13.2.1	222
99.	Добавить пробелы после знаков препинания	II.13	II.13.2.2	222
100.	Заменить # образцом	II.13	II.13.2.3	222
101.	Проверить вхождение образца в запись	II.13	II.13.2.4	223
102.	Упорядочить символы по возрастанию	II.13	II.13.2.5	223
103.	Полусумма	II.13	II.15.4	244

Таблица 2 (продолжение)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
104.	Голосование символов	II.13	II.15.5	244
	Количество перевёртышей	II.13	II.15.6	245
	Количество цифр в тексте (6 баллов)	II.13	Задачник 2.2.6	41
105.	Кратность трём (10 баллов)	II.13	Задачник 2.2.7	41
106.	Испорченное слово (6 баллов)	II.13	Задачник 2.2.9	42
107.	Алфавит записи (6 баллов)	II.13	Задачник 2.2.10	42
108.	НОД (10 баллов)	II.13	Задачник 2.2.12	43
109.	Популярная цифра (5 баллов)	II.13	Задачник 4.2.1	78
110.	Пересечение записей (8 баллов)	II.13	Задачник 4.2.2	78
111.	Сортировка записи (5 баллов)	II.13	Задачник 4.2.3	79
112.	Из двоичной в десятичную (10 баллов)	II.13	Задачник 4.2.6	81
113.	Из римской в десятичную (10 баллов)	II.13	Задачник 4.2.7	81
114.	Из десятичной в двоичную (8 баллов)	II.13	Задачник 5.2.9	96
115.	Самое длинное слово (9 баллов)	II.13	Задачник 5.2.10	97
116.	Ханойская башня (10 баллов)	II.13	Задачник 5.2.11	98
117.	Среднее арифметическое (5 баллов)	II.13	Задачник 6.2.3	107
118.	Деление с остатком (7 баллов)	II.13	Задачник 6.2.7	108
119.	Возведение в степень (9 баллов)	II.13	Задачник 6.2.8	109
	Сократить дробь (10 баллов)	II.13	Задачник 6.2.9	109
120.	Антиавтомат (8 баллов)	II.13	Задачник 6.2.10	110

Таблица 2 (окончание)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
121.	Скобки (7 баллов)	II.13	Задачник 6.2.11	110
122.	Жребий на пальцах (6 баллов)	II.13	Задачник 7.2.7	120
	Кому выходить? (7 баллов)	II.13	Задачник 7.2.8	121
123.	Кто останется? (10 баллов)	II.13	Задачник 7.2.9	121
124.	Шифр-2 (8 баллов)	II.13	Задачник 8.2.7	135
125.	Тарабарский язык (7 баллов)	II.13	Задачник 8.2.8	136
126.	Перевод с тарабарского (7 баллов)	II.13	Задачник 8.2.9	136
127.	Раздвоитель (8 баллов)	II.13	Задачник 8.2.10	137
128.	Координаты вектора (5 баллов)	II.13	Задачник 8.2.11	137
129.	Шахматное поле (9 баллов)	II.13	Задачник 8.2.12	138
130.	Проверка идентификатора	II.14	II.14.1.1	229
131.	Проверить число	II.14	II.14.1.2	230
132.	Пример на сложение в программу для Плюсика	II.14	II.14.2.2	239
133.	Пример на сложение и вычитание в программу для Плюсика	II.14	II.14.2.3	239
134.	Трансляция ПОВТОРИ	II.14	II.14.2.4	239
135.	Модель нового исполнителя	II.14	II.14.2.5	240

## Часть III. Транслятор?.. Это очень просто!

Таблица 3

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
1.	Проверка числа-1 (Кукарача)	III.17	III.17.5.1	272
2.	Проверка числа-2 (Кукарача)	III.17	III.17.5.2	273
3.	Проверка числа-3 (Кукарача)	III.17	III.17.5.3	273
4.	Проверка числа-4 (Кукарача)	III.17	III.17.5.4	274
5.	Проверка выражения (Кукарача)	III.17	III.17.5.5	275
6.	Проверка выражения (Кукарача)	III.18	III.18.2.1	281
7.	Проверка узора-1 (Кукарача)	III.18	III.18.2.2	282
8.	Проверка узора-2 (Кукарача)	III.18	III.18.2.3	283
9.	Анализатор-5 (9 баллов)	III.18	Задачник 3.1.8	51
10.	Морковь и капуста (5 баллов)	III.18	Задачник 4.1.4	70
11.	Морковь и капуста-2 (8 баллов)	III.18	Задачник 4.1.5	72
12.	Странные тексты (6 баллов)	III.18	Задачник 4.1.6	73
13.	Странные тексты-2 (7 баллов)	III.18	Задачник 4.1.7	74
	Проверка программы (6 баллов)	III.18	Задачник 5.1.8	90

Таблица 3 (окончание)

Номер по порядку	Название задачи	Ссылка на тему книги «Азы программирования. Магия для начинающих»	Адрес задачи	Страница
14.	Проверка программы-2 (6 баллов)	III.18	Задачник 5.1.9	91
15.	Исполнитель Кроха (10 баллов)	III.18	Задачник 5.1.10	91
16.	Проверка записи (8 баллов)	III.18	Задачник 6.1.11	104
17.	Кукарача — грамотей (8 баллов)	III.18	Задачник 7.1.6	116
18.	Проверка числа (Корректор)	III.19	III.19.2.1	289
19.	Проверка выражения-1 (Корректор)	III.19	III.19.2.2	290
20.	Проверка выражения-2 (Корректор)	III.19	III.19.2.3	291
21.	Малыш (30 баллов)	III.19	Задачник 4.2.8	82
22.	Проверка программы (6 баллов)	III.19	Задачник 5.2.7	96
23.	Проверка выражения (12 баллов)	III.19	Задачник 6.2.12	111
24.	Хохотунчики (8 баллов)	III.19	Задачник 7.2.11	122