

**Александр Дуванов  
Алексей Рудь  
Виктор Семенко**

**АЗЫ  
ПРОГРАММИРОВАНИЯ  
ФАКУЛЬТАТИВНЫЙ КУРС**

**КНИГА ДЛ Я УЧИТЕЛ Я**

Санкт-Петербург

«БХВ-Петербург»

2005

УДК 681.3.06(075.3)  
ББК 32.973я721  
Д79

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

Д79 Азы программирования. Факультативный курс. Книга для учителя. — СПб.: БХВ-Петербург, 2005. — 496 с.: ил.

ISBN 5-94157-584-X

В книге обобщен уникальный многолетний опыт Роботландского сетевого университета по обучению детей программированию. Материал излагается в стиле курса «Азы информатики» и значительно расширяет и углубляет тему программирования, которая в базовом курсе нацелена на общеобразовательные задачи. Повышенное внимание уделяется основам формализации и построения алгоритмов, тестированию и отладке программ. Подробно рассмотрены алгоритмические конструкции: циклы двух видов, ветвление, применение процедур, рекурсия; целый раздел посвящен принципам построения трансляторов.

Изложена методика проведения занятий и турниров по программированию.

Прилагаемый компакт-диск содержит программные среды исполнителей и решения всех заданий учебника и задачника.

*Для учителей средних образовательных учреждений*

УДК 681.3.06(075.3)  
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. 39,99.

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 5-94157-584-X

© 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>1</b>  |
| Обращение к читателю .....   | 1         |
| История Кукарачи .....   | 2         |
| История Корректора .....   | 4         |
| <br>   |           |
| <b>Вступление .....</b>  | <b>7</b>  |
| Программирование в школьном курсе информатики .....                  | 7         |
| «Азы программирования» и «Азы информатики» .....                     | 8         |
| Примерный план факультатива .....                                    | 9         |
| Первый год обучения .....  | 9         |
| Второй год обучения .....  | 10        |
| Содержание книги .....   | 11        |
| Авторство задач и решений .....                                      | 12        |
| Авторство иллюстраций .....  | 12        |
| Содержание диска .....   | 12        |
| Благодарности .....  | 14        |
| Исполнители и язык программирования .....                            | 15        |
| Исполнитель Кукарача .....   | 15        |
| Исполнитель Корректор .....  | 16        |
| Язык программирования исполнителей .....                             | 19        |
| <br>   |           |
| <b>ЧАСТЬ I. КУКАРАЧА: МЕТОДИКА ЗАНЯТИЙ<br/>И РЕШЕНИЕ ЗАДАЧ .....</b> | <b>19</b> |
| <br>   |           |
| <b>Глава 1. Кукарача и его среда обитания .....</b>                  | <b>23</b> |
| Общие рекомендации и дополнения .....                                | 23        |
| Ответы на вопросы и решение задач .....                              | 25        |
| 1.1. Знакомство с Кукарачей .....                                    | 25        |
| 1.3. Задачи .....  | 25        |

|   |           |
|---|-----------|
| <b>Глава 2. Вася экономит свой труд .....</b>         | <b>29</b> |
| Общие рекомендации и дополнения .....                 | 29        |
| Стиль записи программ.....                            | 30        |
| Пустоты.....  | 31        |
| Структурирование.....                                 | 32        |
| Комментарии.....                                      | 33        |
| Имена процедур .....                                  | 33        |
| Когда код нужно оформлять в отдельную процедуру ..... | 34        |
| Визуальное отделение процедур .....                   | 35        |
| Ответы на вопросы и решение задач.....                | 36        |
| 2.1. Ток — это кот задом наперёд.....                 | 36        |
| 2.2. У компьютера есть память, и это хорошо.....      | 37        |
| 2.5. Задачи.....                                      | 37        |
| <br>  |           |
| <b>Глава 3. Новые команды и их повторение.....</b>    | <b>42</b> |
| Общие рекомендации и дополнения .....                 | 42        |
| Процедурное программирование .....                    | 42        |
| Команда повторения .....                              | 42        |
| Интерпретатор программ .....                          | 43        |
| Классификация ошибок программирования.....            | 43        |
| Построение алгоритма .....                            | 44        |
| Построение программы.....                             | 44        |
| Ответы на вопросы и решение задач.....                | 46        |
| 3.1. Кукарача говорит «Ах!» .....                     | 46        |
| 3.2. Процедурное программирование .....               | 47        |
| 3.3. Команда повторения .....                         | 47        |
| 3.5. Задачи.....                                      | 49        |
| <br>  |           |
| <b>Глава 4. Кукарача на распутье .....</b>            | <b>56</b> |
| Общие рекомендации и дополнения .....                 | 56        |
| Отдельные рекомендации, пояснения и примеры.....      | 58        |
| Переключатели .....                                   | 59        |
| Ответы на вопросы и решение задач.....                | 63        |
| 4.1. Команда ветвления .....                          | 63        |
| 4.2. Особые случаи.....                               | 65        |
| 4.3. Задачи.....                                      | 67        |
| <br>  |           |
| <b>Глава 5. Другой тип повторения .....</b>           | <b>79</b> |
| Общие рекомендации и дополнения .....                 | 79        |
| Ответы на вопросы и решение задач.....                | 79        |
| 5.1. Когда неизвестно число повторений.....           | 79        |

|  |            |
|--|------------|
| 5.2. Как обмануть интерпретатор.....                       | 81         |
| 5.3. Задачи.....   | 81         |
| <b>Глава 6. Кукарача хочет укусить себя за хвост .....</b> | <b>92</b>  |
| Ответы на вопросы и решение задач.....                     | 92         |
| 6.5. Задачи.....   | 92         |
| <b>Глава 7. Решения задач.....</b>                         | <b>106</b> |
| Задачи недели 2002/2003 учебного года.....                 | 106        |
| Задачи к главе 4 .....                                     | 106        |
| Задачи к главе 5 .....                                     | 118        |
| Задачи к главе 6 .....                                     | 127        |
| Задачи недели 2003/2004 учебного года.....                 | 134        |
| Задачи к главам 1–3 .....                                  | 134        |
| Задачи к главам 4–5 .....                                  | 147        |
| Задачи к главе 6 .....                                     | 175        |
| <b>ЧАСТЬ II. КОРРЕКТОР.....</b>                            | <b>179</b> |
| <b>Глава 8. Знакомство с исполнителем .....</b>            | <b>181</b> |
| Корректор и Кукарача .....                                 | 181        |
| Ввод информации.....                                       | 181        |
| Вывод информации .....                                     | 181        |
| Память.....  | 182        |
| Система команд .....                                       | 183        |
| Язык программирования.....                                 | 184        |
| Общие рекомендации .....                                   | 185        |
| Ответы на вопросы и задания.....                           | 186        |
| 8.1. Корректор и его среда обитания.....                   | 186        |
| 8.2. Попробуем управлять.....                              | 187        |
| 8.3. Управление при помощи программы.....                  | 187        |
| <b>Глава 9. Язык программирования.....</b>                 | <b>188</b> |
| Ответы на вопросы и задания.....                           | 188        |
| 9.1. Процедурное программирование .....                    | 188        |
| 9.2. Циклы.....  | 189        |
| 9.3. Развилки.....   | 196        |
| 9.4. Рекурсия.....   | 200        |

|  |            |
|--|------------|
| <b>Глава 10. Отладка программ.....</b>                           | <b>205</b> |
| Ответы на вопросы и задания.....                                 | 205        |
| 10.2. Синтаксические ошибки.....                                 | 205        |
| 10.3. Ошибки программирования .....                              | 207        |
| 10.4. Тестирование .....   | 208        |
| 10.5. Задачи .....   | 215        |
| <br>   |            |
| <b>Глава 11. Приёмы программирования Корректора .....</b>        | <b>222</b> |
| Алгоритмические формулы .....                                    | 222        |
| Обработка записи известной длины n .....                         | 224        |
| Обработка записи неизвестной длины.....                          | 225        |
| Поиск объекта, его изменение и возврат в исходное положение..... | 226        |
| Подсчёт двух количеств за один проход .....                      | 228        |
| Вставка символа в запись .....                                   | 231        |
| Удаление символа из записи .....                                 | 232        |
| Определение чётности суммы слагаемых .....                       | 233        |
| Алгоритмические формулы для логических выражений .....           | 236        |
| Операция <i>И</i> .....  | 236        |
| Операция <i>И</i> : модель с ветвью <i>ИНАЧЕ</i> .....           | 239        |
| Операция <i>ИЛИ</i> .....  | 241        |
| Ответы на вопросы и задания.....                                 | 242        |
| 11.1. Как найти конец текста .....                               | 242        |
| 11.2. Как вернуться в исходное место .....                       | 247        |
| 11.3. Задачи .....   | 251        |
| <br>   |            |
| <b>Глава 12. Арифметика чисел, палочек и символов .....</b>      | <b>265</b> |
| Структурное программирование и библиотечные процедуры .....      | 265        |
| Решение задач.....   | 267        |
| 12.1. Арифметика чисел .....                                     | 267        |
| 12.2. Арифметика палочек .....                                   | 271        |
| 12.3. Арифметика символов.....                                   | 281        |
| <br>   |            |
| <b>Глава 13. Преобразования, подсчёты, редактирование.....</b>   | <b>293</b> |
| Решение задач.....   | 293        |
| 13.1. Длина текста.....  | 293        |
| 13.2. Корректор оправдывает своё имя .....                       | 300        |
| <br>   |            |
| <b>Глава 14. Трансляторы.....</b>                                | <b>312</b> |
| Ответы на вопросы и решения задач.....                           | 312        |
| 14.1. Проверка объектов.....                                     | 312        |
| 14.2. Транслятор для Плюсика .....                               | 323        |

|  |            |
|--|------------|
| <b>Глава 15. Решения задач</b> .....                             | <b>342</b> |
| Задачи недели 2002/2003 учебного года.....                       | 342        |
| Задачи, рекомендуемые к главам 11 и 12.....                      | 342        |
| Задачи, рекомендуемые к главам 12 и 13.....                      | 347        |
| Задачи недели 2003/2004 учебного года.....                       | 360        |
| Задачи к главам 8 и 9.....                                       | 360        |
| Задачи к главе 10.....   | 375        |
| Задачи к главе 11.....   | 392        |
| Задачи к главе 12.....   | 403        |
| <br>   |            |
| <b>ЧАСТЬ III. ТРАНСЛЯТОР?.. ЭТО ОЧЕНЬ ПРОСТО!</b> .....          | <b>425</b> |
| <br>   |            |
| <b>Глава 16. Язык Бэкуса-Наура</b> .....                         | <b>427</b> |
| Ответы на вопросы и задания.....                                 | 427        |
| <br>   |            |
| <b>Глава 17. Кукарача и лексический анализ выражений</b> .....   | <b>432</b> |
| Определение 1.....   | 432        |
| Решение задач.....   | 434        |
| 17.5. Задачи.....  | 434        |
| <br>   |            |
| <b>Глава 18. Ах уж эта рекурсия!</b> .....                       | <b>448</b> |
| Удивительная рекурсия.....                                       | 448        |
| Подводные камни рекурсии.....                                    | 451        |
| Первый камень.....   | 451        |
| Второй камень.....   | 452        |
| Третий камешек (в адрес косвенной рекурсии).....                 | 452        |
| Классификация типов рекурсии.....                                | 452        |
| Прямая и косвенная рекурсия.....                                 | 452        |
| Терминальные и нетерминальные рекурсии.....                      | 455        |
| Решение задач.....   | 457        |
| 18.2. Задачи.....  | 457        |
| <br>   |            |
| <b>Глава 19. Лексический анализатор в среде Корректора</b> ..... | <b>467</b> |
| 19.2. Задачи.....  | 467        |
| <br>   |            |
| <b>Глава 20. Построение трансляторов</b> .....                   | <b>483</b> |
| <br>   |            |
| <b>Решение задачи</b> .....                                      | <b>483</b> |
| Задача 2.....  | 483        |

# 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).

## Обращение к читателю

Уважаемые коллеги!

Мы с вами любим информатику, а особенно, — что греха таить! — программирование.

Ведь программирование — это математика информатики: «ум в порядок приводит», и её музыка: доставляет изысканное наслаждение!

Мы с вами, конечно, вкусили, вкушаем и, надо думать, будем вкушать яблочки с этого дерева, пусть не самые крупные, но не менее сладкие и полезные, насыщенные витамином увлечённости и азарта.

К сожалению, в последнее время интерес молодежи к программированию угасает. Как жаль! Ведь программирование — не только замена пустому времяпрепровождению. Программирование — это солидный багаж для вступления в успешную жизнь. Спрос на программистов только растёт. И получают они за свою работу хорошие деньги. Гораздо большие, чем простые компьютерные пользователи.

А чем сейчас увлекается молодежь на компьютерах? Правильно: в основном игрушками. Кому нужны такие «спецы»?

В Роботландском университете есть несколько программистских курсов: Азы программирования (31), Буки программирования (32), Web-программирование (43). Каждый из этих курсов не просто интересен, но и уникален. Но в основе всех их лежат подходы, которые восходят к великому Дийкстре.

Классик Э. Дийкстра говорил, что самый главный язык, который должен знать программист, — это свой родной, на котором он изъясняется в повседневной жизни.

Иными словами, соль программирования не в языке программирования, а в умении чётко сформулировать задачу, выдвинуть идею решения, разработать алгоритм! И только потом перевести алгоритм в программу, записав несколько заклинаний на языке посвящённых!

Этот тезис особенно ярко проявляет свою суть в «Азах программирования». Здесь нет проблем с языком — он тривиален, а идей, алгоритмов — выше головы!

Книга обобщает многолетний опыт Роботландского университета, а ключевые особенности обучения можно конспективно изложить так:

- «Голая» алгоритмика, практически без структур данных, т. к. цель — хорошо прочувствовать алгоритмические конструкции: *циклы двух видов, ветвление, применение процедур, рекурсию.*
- Привычка максимально использовать то, что есть, думать, как обойтись меньшими ресурсами.
- Не отвлекаться на «машинные» особенности.
- Универсальный язык записи, с которого легко перейти на любой процедурный. Языка-то практически нет — одни алгоритмы!
- А вот элементы структурности есть, подобно Паскалю и Си.
- Вкус не к «наворотам», а именно к *алгоритмам*, к *мышлению*.
- Большой класс задач, тренирующих применение рекурсии. Не так уж просто, даже для старшеклассников и студентов.
- Целый раздел и серия задач «Транслятор — это просто!» Становятся понятными (не только учителю, но и 5–8-классникам) принципы построения анализаторов и интерпретаторов.

## 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.

Тараканы и Муравьи начали свой победный марш по нашим широким просторам. В качестве примера можно указать самарскую работу: Г. Н. Гутман, О. М. Карпилова. Муравьиные сказки. Книга для учащихся. Москва. Просвещение. 1993.

Самарский Муравей больше походит на Кукарачу (хотя в книге ссылки на роботландского исполнителя нет), чем на своего новосибирского однофамильца. Однако кукарачинский язык в Самаре неоправданно усложнён введением переменных, перечисляемым циклом, подпрограммой с параметрами и другими дополнениями, которые, с одной стороны, сводят на нет первоначальную простоту, а с другой — плохо вписываются в примитивную клетчатую среду и систему команд исполнителя.

Кукарача вошёл в состав курса Роботландии с самого начала. Алгоритмические структуры по своей сути — вещь житейская, общеобразовательная и необходимая для изучения именно в младших классах, когда формируется

стиль мышления ребёнка. Именно поэтому, а не с целью подготовки юных программистов, Кукарача внедрился в школы. Среда, понятная ребёнку, предельно простая система команд и структурный процедурный язык (без всяких goto) хорошо соответствовали поставленной задаче.

Придумав исполнителя, мы даже и не подозревали о 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 на уровне творческих безумств и вдохновения, съедающего всё время. 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 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 Корратора получили логическое продолжение, более оправданное в его псевдотьюринговской среде, чем на клетчатом поле коллеги.

# Вступление

## Программирование в школьном курсе информатики

Введение в программирование обязательно должно быть составной частью школьного курса информатики. Это как формулы в математике. Можно много говорить о важности математического мышления, но пока школьник на практике не пройдёт через вычисление гипотенузы по формуле Пифагора, все слова будут красивы, но бесполезны.

Так и в информатике. Можно много говорить об алгоритмическом мышлении, приводить примеры с алгоритмами перехода улицы и заварки чая, но пока школьник не напишет код с ветвлениями, циклами и рекурсией для конкретного формального исполнителя, не пройдёт через этап тестирования и отладки, разговор об алгоритмическом мышлении будет красив, но бесполезен.

С другой стороны, не хочется впадать и в другую крайность, подменять алгоритмическую культуру техническим программированием, а алгоритмику — языком программирования. Ведь не все школьники в будущем станут программистами, но алгоритмические навыки нужны всем как элемент культуры и как элемент повседневной практики, профессиональной (в любой области) и бытовой.

В курсе «Азы информатики» (см. начало следующего пункта) этот важный вопрос решается при помощи компромисса между простотой учебных исполнителей и полнотой алгоритмических построений на базе простого языка программирования.

Есть исполнитель (Кукарача, Корректор) с минимальной системой команд, есть язык программирования с минимальным числом интуитивно понятных конструкций и есть наглядные среды, в которых выполнение программы представлено визуально (перемещение исполнителей, кубиков, изменение сред на экране), а не скрыто в железных недрах системных плат. Серьёзные алгоритмические построения выполняются на этой примитивной базе в «чистом» виде, без технических деталей «настоящих» языков программирования и изощрённой сложности «настоящих» исполнителей.

Большая проблема для новичка — понятие переменной. Переменных нет в роботландском языке управления исполнителями. Вся техническая сторона программирования предельно проста и наглядна. Зато алгоритмические структуры — настоящие: процедуры, циклы, развилки, рекурсии. Задачи в средах Кукарачи и Корректора тоже настоящие (вплоть до построения трансляторов!). Блеск и восторг! Авторы верят, что читатели смогут в этом убедиться самостоятельно, без излишней агитации! Недаром Кукарача и Корректор живут уже 20 лет и продолжают набирать обороты.

## «Азы программирования» и «Азы информатики»

«Азы информатики» — это базовый курс информатики для начинающих. Он рассчитан на 5 лет школьного обучения, начиная с пятого класса общеобразовательной школы. Автор курса — А. А. Дуванов. Сохраняя методические идеи классической «Роботландии», новый курс предлагает школьнику и педагогу современные средства для реализации педагогической задачи, делает обучение более эффективным, увлекательным и контролируемым. Основной методический приём курса — формирование концептуальных основ информатики через практические задачи, решаемые на компьютере. Курс поддержан электронными лабораториями и бумажными учебниками издательства «БХВ-Петербург». Дополнительную информацию о курсе можно получить на сайте [www.botik.ru/~robot](http://www.botik.ru/~robot) или от автора с адреса [kurs@robotland.pereslavl.ru](mailto:kurs@robotland.pereslavl.ru).

Тема программирования, конечно, отражена в курсе «Азы информатики», но в небольшом, общеобразовательном объёме (как закрепление алгоритмики на уровне кодирования формального исполнителя).

Используя же «Азы программирования», учитель сможет поставить факультатив для тех школьников 5–8 классов, для которых программирование станет будущей успешной профессией. Книга поможет увлечённым и азартным ребятам раскрыть свои интересы, развить способности.

Эмоциональный фон книги увлечёт темой, а её содержание научит серьёзному программированию, очищенному от шелухи сложных языковых конструкций и многочисленных технических деталей производственных языков программирования.

Материал книги обкатан многолетней практикой курсов Роботландского университета, выверен педагогически и является уникальным (серьёзные задачи в примитивных средах).

## Примерный план факультатива

По мнению авторов, на базе «Азов программирования» можно построить двухгодичный факультатив по программированию для учащихся 5–8 классов.

В таблицах 1 и 2 приводится примерный план, рассчитанный на 68 часов (2 года, по 1 часу в неделю). В графе «Тема» обозначено соответствие изучаемого материала разделам книг «Азы программирования. Магия для начинающих. Книга ученика» («Учебник»), «Азы программирования. Задачи роботландских турниров» («Задачник»).

## Первый год обучения

Таблица 1

| Номер                      | Тема   | Часы |
|----------------------------|--|------|
| Учебник. Часть I. Кукарача |  |      |
| 1                          | Глава 1. Кукарача и его среда обитания (среда, управление при помощи команд)                 | 1    |
| 2                          | Глава 2. Вася экономит свой труд (понятие программы)   | 1    |
| 3                          | Глава 3. Новые команды и их повторение (процедурное программирование, команда цикла ПОВТОРИ) | 1    |
| 4                          | Глава 7. Задачи 14–19  | 1    |
| 5                          | Глава 4. Кукарача на распутье (ветвления)  | 2    |
| 6                          | Глава 7. Задачи 1–6  | 2    |
| 7                          | Глава 5. Другой тип повторения (цикл ПОКА)   | 2    |
| 8                          | Глава 7. Задачи 7–11, 20–27  | 3    |
| 9                          | Глава 6. Кукарача хочет укунить себя за хвост (рекурсивные программы)                        | 2    |
| 10                         | Глава 7. Задачи 12–13, 28  | 2    |
| 11                         | Олимпиада  | 1    |
| 12                         | Разбор решений   | 1    |
| Задачник (задачи Кукарачи) |  |      |
| 13                         | Глава 1. Олимпиада 1996/1997   | 1    |
| 14                         | Глава 2. Турнир 1997/1998  | 2    |
| 15                         | Глава 3. Турнир 1998/1999  | 2    |

Таблица 1 (окончание)

| Номер  | Тема                         | Часы |
|--------|------------------------------|------|
| 16     | Глава 4. Турнир 1999/2000    | 2    |
| 17     | Глава 5. Турнир 2000/2001    | 2    |
| 18     | Глава 6. Турнир 2001/2002    | 2    |
| 19     | Глава 7. Турнир 2002/2003    | 2    |
| 20     | Глава 8. Турнир 2003/2004    | 2    |
| 21     | Глава 9. Олимпиада 2004/2005 | 1    |
| 22     | Олимпиада                    | 1    |
| 23     | Разбор решений               | 1    |
| Итого: |                              | 34   |

## Второй год обучения

Таблица 2

| Номер                        | Тема   | Часы |
|------------------------------|--|------|
| Учебник. Часть II. Корректор |  |      |
| 1                            | Глава 8. Знакомство с исполнителем                 | 1    |
| 2                            | Глава 9. Язык программирования Корректора          | 1    |
| 3                            | Глава 15. Задачи 7–12                              | 1    |
| 4                            | Глава 10. Отладка программ                         | 1    |
| 5                            | Глава 15. Задачи 13–17                             | 1    |
| 6                            | Глава 11. Приёмы программирования Корректора       | 1    |
| 7                            | Глава 15. Задачи 18–21                             | 1    |
| 8                            | Глава 12. Арифметика чисел, палочек и символов     | 2    |
| 9                            | Глава 15. Задачи 1–3, 22–26                        | 2    |
| 10                           | Глава 13. Преобразования, подсчеты, редактирование | 2    |
| 11                           | Глава 15. Задачи 4–6                               | 2    |
| 12                           | Глава 14. Трансляторы                              | 2    |

Таблица 2 (окончание)

| Номер   | Тема  | Часы |
|---|---|------|
| Учебник. Часть III. Транслятор?.. Это очень просто! |   |      |
| 13  | Глава 16. Язык Бэкуса-Наура                       | 1    |
| 14  | Глава 17. Кукарача и лексический анализ выражений | 1    |
| 15  | Глава 18. Ох уж эта рекурсия!                     | 1    |
| 16  | Глава 19. Лексический анализатор Корректора       | 1    |
| 17  | Глава 20. Примитивный транслятор                  | 2    |
| 18  | Олимпиада   | 1    |
| 19  | Разбор решений                                    | 1    |
| Задачник (задачи Корректора)                        |   |      |
| 20  | Глава 2. Турнир 1997/1998                         | 1    |
| 21  | Глава 3. Турнир 1998/1999                         | 1    |
| 22  | Глава 4. Турнир 1999/2000                         | 1    |
| 23  | Глава 5. Турнир 2000/2001                         | 1    |
| 24  | Глава 6. Турнир 2001/2002                         | 1    |
| 25  | Глава 7. Турнир 2002/2003                         | 1    |
| 26  | Глава 8. Турнир 2003/2004                         | 1    |
| 27  | Олимпиада   | 1    |
| 28  | Разбор решений                                    | 1    |
| Итого:  |   | 34   |

## Содержание книги

В этой книге вы найдёте дополнительные материалы по темам, излагаемым в книге ученика, методические рекомендации и решение всех задач (решения заданий задачника помещены на CD, который сопровождает эту книгу).

Задач очень много, решения не всегда короткие, поэтому на большей части этих страниц вы обнаружите программные коды, снабжённые описаниями алгоритмов и наборами данных для тестирования.

Решаясь на такую объёмную публикацию, мы сказали себе:

- рабочие коды в книге обязаны быть: они должны демонстрировать на практике те приёмы и те стили программирования, которые обсуждаются на теоретических страницах. Если в кодах, поставляемых на CD, авторы

из-за недостатка времени (а чаще просто из-за лени) позволяют себе определённую небрежность, то в бумажном варианте это категорически невозможно! Все коды должны иметь идеальный вид без малейших расхождений с рекомендациями на страницах учебника;

- коды, опубликованные в книге, должны легко читаться и пониматься без всякого запуска на компьютере, с ними можно будет работать прямо на диване, ну, может быть, вооружившись ручкой и блокнотом. Этот принцип, конечно, универсально относится к любому программированию, но для учебных кодов он особенно важен. Поэтому программы мы предварительно подробно описывали алгоритмическими описаниями и не жалели строк для комментариев (эта конструкция языка программирования использовалась нами чаще других);
- программирование немыслимо без тщательно спланированного тестирования. Все программные решения мы будем снабжать наборами продуманных тестов и создавать тем самым мостик от общих рекомендаций по отладке к привитию практических навыков тестирования.

Ох, и задали мы себе работу, ох, и возложили на себя ответственность! Читателю предстоит проверить, насколько хорошо авторы сумели выполнить поставленные перед собой задачи.

## Авторство задач и решений

Условие каждой задачи и каждое решение сопровождаются в книге ссылкой на автора этой задачи и этого решения. Если такой ссылки нет, то это означает, что автором задачи и (или) решения является А. А. Дуванов.

## Авторство иллюстраций

Автором всех иллюстраций в учительской, ученической книгах и задачнике является А. А. Дуванов. Для построения рисунков были использованы художественные заготовки А. А. Русса (художник Роботландии).

## Содержание диска

Диск содержит:

- программные среды исполнителей Кукарача и Корректор (автор продукта А. А. Дуванов);
- коды программ, разбираемых на страницах учебника;
- коды программ-решений заданий учебника;

- коды программ-решений заданий задачника;
- методические рекомендации и описание решений заданий задачника в двух видах:
  - для печати в виде doc-файлов с картинками высокого разрешения;
  - для просмотра с экрана в виде гипертекстового приложения.

Программные коды Кукарачи располагаются на CD по адресу:

```
./azpr/kurs/coc97/task/
```

Программные коды Корректора располагаются на CD по адресу:

```
./azpr/kurs/kor97/task/
```

Папки `task` имеют следующую структуру:

I (Часть I. Кукарача)

- 1 (Решения задач первой главы части I.)
- 2 (Решения задач второй главы части I.)
- 3 (Решения задач третьей главы части I.)
- 4 (Решения задач четвёртой главы части I.)
- 5 (Решения задач пятой главы части I.)
- 6 (Решения задач шестой главы части I.)
- 7 (Решения задач седьмой главы части I.)
- book (Коды программ из текста учебника.)

II (Часть II. Корректор)

III (Часть III. Транслятор?.. Это очень просто!)

IV (Задачник)

Имена программ построены по следующему правилу:

`nmmkxxx.coc` — программа для Кукарачи;

`nmmkxxx.kor` — программа для Корректора.

Здесь:

- `n` — номер части;
- `mm` — номер главы внутри части;
- `kk` — номер пункта внутри главы;
- `xx` — номер задачи внутри пункта.

Если задача имеет несколько вариантов решения, то к имени добавляется ещё и номер варианта.

### Пример

#### Файл

`./azpr/kurs/coc97/task/I/2/10205031.coc`

является первым вариантом решения задачи 3 из пятого пункта второй главы первой части книги учебника.

Методические рекомендации и описание решений заданий задачника располагаются на CD по адресу:

□ `./book/print/` — doc-файлы для печати;

□ `./book/screen/` — гипертекстовое приложение, стартовый файл `index.htm`.

Вы можете время от времени заглядывать в сетевой каталог:

**`ftp://ftp.botik.ru/rented/robot/univer/3/book/`**

В эту папку мы будем помещать обновления и дополнения материалов, содержащихся на CD к этой книге.

## Благодарности

Исполнители Кукарача и Корректор не увидели бы свет, как и эта книга, если бы не пионерские труды Г. А. Звенигородского, проложившие первые тропинки для школьной информатики в Новосибирском академгородке.

Трудно представить появление героев (а значит, и самой книги) без работы благовещенской группы, в которой особую роль играли Олег Какаулин и Валентина Немилостива. Дополнительно тёплые слова благовещенским художникам, создавшим портреты Кукарачи (Наташа Медведева) и Корректора (Алексей Морозов).

Е. П. Лилитко можно назвать духовным отцом Кукарачи (вдохнул в исполнителя рекурсивную душу).

Непосредственными участниками затеи являются авторы великолепных задач и решений. Их имена украшают книгу в соответствующих местах. Особая благодарность талантливому человеку, юному программисту из Снежинска — Даниле Первушину. Его остроумные находки не раз приводили обложечных авторов в восхищение!

Я. Н. Зайдельман совершил над рукописью колоссальный труд: он проверил тексты, задачи, решения. В результате рабочих дискуссий многие страницы были улучшены принципиально.

Наконец, спасибо всем участникам роботландских курсов «Азы программирования» (Магам, Магистрам и всем остальным). Материал книги вырос из вашего нескучного коллективного труда.

## Исполнители и язык программирования

Далее приводится краткое описание Кукарачи, Корректора и языка программирования этих исполнителей.

### Исполнитель Кукарача

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



Рис. 1. Клетчатое поле с исполнителем и кубиками

Кукарача может переползть 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 и табл. 3).



Рис. 2. Система команд Кукарачи

Таблица 3

| 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 следующие проверки (запись <символ> обозначает шаблон, на месте которого может быть любой символ) (табл. 4).

Таблица 4

| 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).

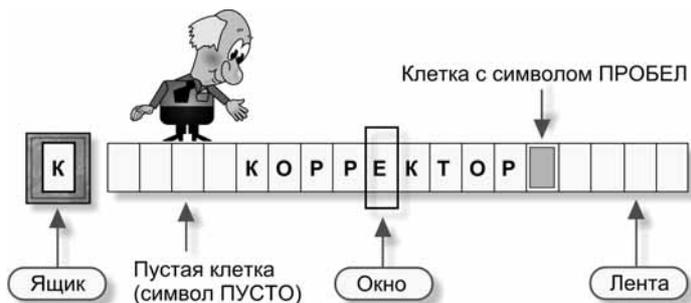


Рис. 3. Среда Корректора

На рис. 4 и в табл. 5 приводится полный набор команд Корректора и его алфавит.



Рис. 4. Система команд Корректора

Таблица 5

| Команда<br>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. 6).

Таблица 6

| 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.

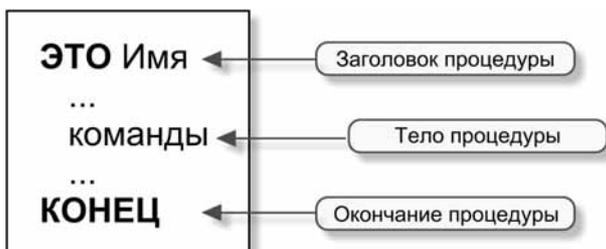


Рис. 5. Описание процедуры

## 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

Команды исполнителя представлены в табл. 7.

Таблица 7

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



## **Часть I**

# **Кукарача: методика занятий и решение задач**

**Глава 1. Кукарача и его среда обитания**

**Глава 2. Вася экономит свой труд**

**Глава 3. Новые команды и их повторение**

**Глава 4. Кукарача на распутье**

**Глава 5. Другой тип повторения**

**Глава 6. Кукарача хочет укусить себя за хвост**

**Глава 7. Решение задач**

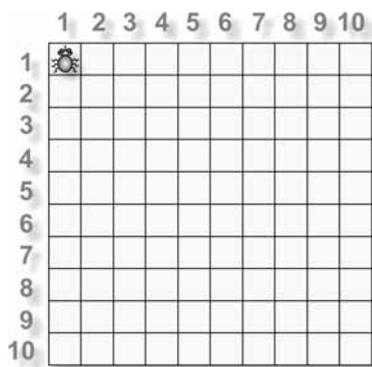


## Глава 1

# Кукарача и его среда обитания

## Общие рекомендации и дополнения

В этом разделе среда исполнителя предьявлена в виде клетчатого поля размером  $10 \times 10$  (рис. 1.1).



**Рис. 1.1.** Клетчатое поле с исполнителем

Однако многие интересные задачи будут относиться к более широкому клетчатому пространству и даже бесконечному во всех направлениях.

Кукарача ползает внутри поля, подчиняясь командам **ВЛЕВО**, **ВПРАВО**, **ВВЕРХ**, **ВНИЗ**. Команда **СТОЯТЬ** является «пустой», она не меняет среду и положение исполнителя в ней (рис. 1.2).

Когда Кукарача наткнется на один или несколько кубиков, он толкает их перед собой, пока не вытолкнет за пределы поля. Сам исполнитель покинуть среду не в состоянии (сообщение «Не могу»).

Для представления Кукарачи можно использовать общую схему знакомства с исполнителем, вынесенную на классную доску, кодоскоп или демонстрационный экран (рис. 1.3).



Рис. 1.2. Система команд Кукарачи

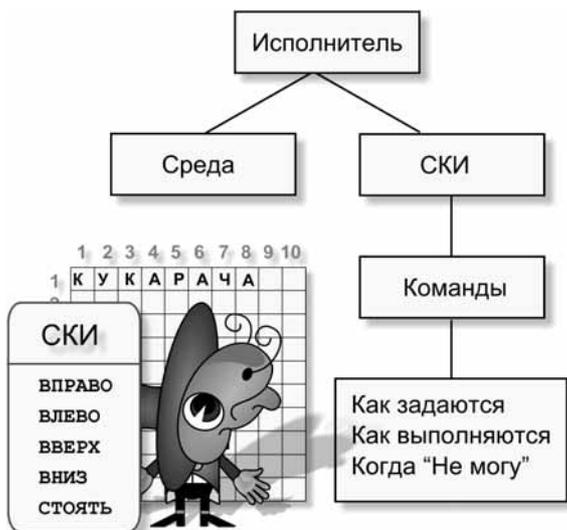


Рис. 1.3. Схема знакомства с исполнителем

Следуя нисходящему методу работы над проектом (технология «сверху вниз»: сначала общие идеи, затем проработка деталей), рекомендуется следующая последовательность этапов решения задачи:

1. Обсуждение задачи и построение траектории движения исполнителя.
2. Составление алгоритма в общем виде на русском языке.
3. Детализация алгоритма (возможно в нескольких уровнях) на русском языке.
4. Запись программы (последовательность команд из СКИ).
5. Выполнение.

Можно записывать программу параллельно с выполнением, так как это делал Петя в задаче с куликом. Такая работа полезна, т. к. наглядно показывает взаимосвязь алгоритма и программы. При этом хорошо видно, например, что одному пункту алгоритма может соответствовать несколько команд из СКИ.

## Ответы на вопросы и решение задач

### 1.1. Знакомство с Кукарачей

1. Что сделает Кукарача, получив команду **НАЛЕВО**?

**Ответ.** Покажет сообщение «Не понимаю».

2. Кукарача стоит в левом нижнем углу поля. Как он выполнит команду **ВНИЗ**?

**Ответ.** Покажет сообщение «Не могу».

3. Кукарача находится в клетке (1,1). Где он окажется после выполнения следующих команд?

- |           |           |           |           |
|-----------|-----------|-----------|-----------|
| а) ВПРАВО | б) ВПРАВО | в) ВПРАВО | г) ВПРАВО |
| ВНИЗ      | ВПРАВО    | ВНИЗ      | ВНИЗ      |
| ВПРАВО    | ВНИЗ      | ВЛЕВО     | ВЛЕВО     |
| ВНИЗ      | ВЛЕВО     | ВНИЗ      | СТОЯТЬ    |
| ВНИЗ      | ВЛЕВО     | ВНИЗ      |           |

**Ответ.** а) в клетке (4,3); б) в клетке (2,1); в) в клетке (4,1); г) в клетке (2,2).

### 1.3. Задачи

1. Превратить молоток в моток (рис. 1.4).



Рис. 1.4

### Решение

В табл. 1.1 приводится алгоритм решения и команды исполнителя, реализующие шаги алгоритма.

Таблица 1.1

| Шаги алгоритма                             | Команды                    |
|--|----------------------------|
| 1. Выбросить Л и шагнуть к следующей букве | ВВЕРХ ВНИЗ ВПРАВО          |
| 2. Выбросить О и шагнуть к следующей букве | ВВЕРХ ВНИЗ ВПРАВО          |
| 3. Подойти к концу слова                   | ВПРАВО ВПРАВО ВПРАВО ВВЕРХ |
| 4. Уплотнить запись                        | ВЛЕВО ВЛЕВО                |