

№ 1 (290) 1-7 января 2001

еженедельное
приложение к газете
«ПЕРВОЕ СЕНТЯБРЯ»

ИНФОРМАТИК

ИНФОРМАТИКА
2001
НОВОГО ВЕКА

Ю.А. ПЕРВИН

Зимние вечера

Информатика

для начинающих

Выпуск I

Материалы
Роботландского
университета

Купон для получения скидки
на программную поддержку
“Зимних вечеров” помещен
на с. 18

Содержание*

(первый семестр)

Разговор за чашкой чая (вместо предисловия)

Раздел I. Введение

Вечер 1. Про программы, пиктограммы и курсор

Рабочий стол

“Корешки и верхки”

Как не заблудиться в городе

Как построить задачу

Вечер 2. Что такое алгоритм

Нужен четкий план

Переезд через реку

Как задача о переезде через реку решается на компьютере

Вечер 3. Строковый редактор

Какие бывают ошибки

“Правилка” показывает мультимедиа

Упражнения “Правилки”

Вечер 4. Побеседуем с компьютером

“Привет”

Как работает программа “Привет”

Сказка по выбору

Вечер 5. Хранить, чтобы помнить

Как запомнить информацию

Календарь

Записная книжка

Раздел II. Алгоритмы и исполнители

Вечер 6. “Ханойские башни”

“Ханойская башня” — игрушка или задача?

От сложного к простому

Что такое рекурсия

Подсчитаем число переносов

Вечер 7. Шахматный этюд

Координаты, шахматная доска

Задача о перестановке шести коней

Вечер 8. Исполнители

Исполнитель и его система команд

Что надо знать об исполнителе

Аварийные сообщения исполнителя

Как познакомиться с исполнителем

Квадратик

Вечер 9. Машинист

На железнодорожной станции

Исполнитель Машинист

Машинист сортирует состав

Вечер 10. Плюстик

Что такое стек

Как Плюстик работает со стекком

Задачи для Плюстика

Раздел III. Классификаторы

Вечер 11. Классификаторы-“догадалки”

Кто лишний

Кто с нами

Цепочки

Метки

Вечер 12. Классификаторы-“собиралки”

Монтажник

Игра 15

Пары

Вечер 13. Промежуточный финиш

Вечер 14. Перекрестная проверка

Разговор за чашкой чая

Моего соседа по лестничной площадке зовут Георгий Аркадьевич. Но внуки Георгия Аркадьевича часто (правда, за глаза) называют его по фамилии — дедушка Фёрстов. Это мой старый приятель. Я говорю “старый” вовсе не потому, что дедушка Фёрстов так уж стар. Вовсе нет. Просто мы с ним давно знакомы, дружим со студенческих лет.

Дедушка Фёрстов увлекается информатикой и компьютерами. Будучи чуть помоложе, он преподавал информатику в школе. Сейчас Георгий Аркадьевич на пенсии и все свое время отдает внукам — второклассникам Дине и Тимофею.

Недавно за чашкой чая Георгий Аркадьевич поделился со мною своими планами.

— Пора сажать моих внучат за компьютер.

— Не рано ли? — усомнился я. — Ведь в школе информатику со второго класса не преподают.

Но Георгий Аркадьевич резонно возразил:

— Во-первых, они сейчас начинают учиться думать. Компьютер им в этом очень поможет. Во-вторых, много интересного найдут они на компьютере для уроков русского языка, математики и природоведения. А скоро у них начнутся еще и уроки французского. Вот тут-то компьютерные упражнения будут для них очень полезны. В-третьих, я был недавно на родительском собрании у внуков. Оказывается, к будущему учебному году школа приобретает новый компьютерный класс специально для малышей.

Все это звучало убедительно.

— Но ведь это совсем не просто — научить малышей алгоритмам и программам, циклам и ветвлениям, редакторам и сортировкам.

Наверно, и на эти мои сомнения у дедушки Фёрстова был ответ. Но он не стал ничего объяснять, а пригласил меня заходить по вечерам; чтобы посмотреть, как его молодежь делает первые шаги в информатике.

— Когда мои дети были маленькими, — сказал он, — я придумывал им сказки, начинавшиеся словами: “Жил-был однажды компьютер...” Сейчас они выросли. Один учит школьников рисованию, используя на своих уроках машинную графику. Другой работает в редакции газеты, где без компьютеров никак не обойтись. Теперь их детям такие сказки рассказывать не надо. Внуки видят своими глазами компьютеры и дома, и в школе, и у родителей на работе...

Так я стал по вечерам приходить к Георгию Аркадьевичу Фёрстову. Я проходил на цыпочках в его кабинет и тихонько, стараясь не шуметь, усаживался на старый диван, чтобы видеть и экран компьютера, стоявшего на письменном столе дедушки Фёрстова, и склоненные над клавиатурой лица его любознательных внуков Дины и Тима.

* Приведено полное содержание первой части “Зимних вечеров”. В этот номер вошли материалы 1—4 вечеров и соответствующие им разделы книги для учителя. Продолжение читайте в февральских номерах.

Раздел I. ВВЕДЕНИЕ

Вечер первый

Про программы, пиктограммы и курсор

Рабочий стол

Когда в первый из этих вечеров дедушка Фёрстов пригласил ребят к своему компьютеру, они завизжали от восторга и наперегонки бросились в комнату Георгия Аркадьевича. Дина даже не успела доесть вкусный пирожок, которым ее только что угостила бабушка.

— Стоп, стоп! — решительно остановил их дедушка Фёрстов. — Никаких пирожков! Сначала все съешьте на кухне и только потом подходите к компьютеру. — Понятно, почему я это требую от вас?

— Конечно, конечно, — извиняющимся голосом за двоих ответила Дина. — Крошки могут насыпаться на клавиатуру, и она испортится.

— Правильно говоришь, Диночка. Никаких пирожков, конфеток, жвачки! И обязательно чистые руки.

— Угу! — негромко сказал Тим, почему-то пряча свои ладошки за спину.

Но этот жест не ушел от внимательного взгляда дедушки. Пришлось Дине возвращаться с пирожком на кухню, а Тим поспешил в ванную, чтобы отмыть руки: он незадолго до этого чистил свой велосипед и готовил его к большому воскресному походу.

— Ну, можно включать? — спросил Тим, когда дети наконец уселись по обеим сторонам от дедушки около компьютера.

— Не торопитесь. Сегодня компьютер включу я сам. А вы посмотрите.

Дедушка нажал на кнопку включения на передней панели компьютера. Загорелись лампочки-индикаторы, по экрану быстро побежали строчки символов.

— Компьютер загружается программами, с которыми он будет работать, — начал дедушка объяснения.

Тем временем на просторном экране появилось множество ярлычков с надписями.

— Работой компьютера управляют программы... — продолжил Георгий Аркадьевич. Дети понимающе кивали головами: мол, это каждому ясно.

— Перед вами — рабочий стол компьютера: экран, на котором сейчас видны обозначения некоторых из программ. Вот эти ярлычки, называемые *пиктограммами*, и являются обозначениями программ.

— Вот это и есть все компьютерные программы? — недоверчиво спросил Тим.

— Ну, конечно, не все, — улыбнулся дедушка Фёрстов, — а только самые необходимые, которыми я пользуюсь чаще всего. Однако вы потом увидите, что с рабочего стола можно легко добраться до любых программ, даже до тех, пиктограммы которых отсутствуют на столе.

Когда пиктограмм на экране не много, то выбрать из них ту, которая сейчас нужна, не трудно...

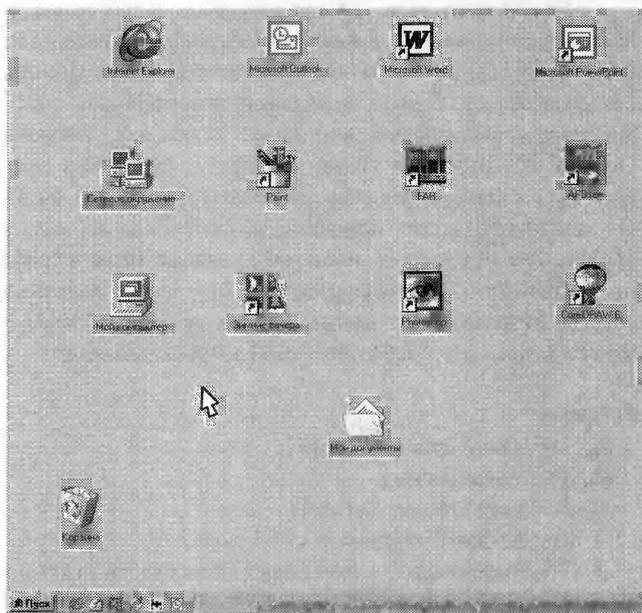
— Как в меню нашей школьной столовой, — встала Дина.

— Удачное сравнение! — обрадовался дедушка, хотя он отлично понимал, что у Дины мысль о сходстве школьного меню с небольшим числом пиктограмм на экране возникла вовсе не из-за внешнего сходства двух меню, а исключительно в силу того обстоятельства, что оба они отображают только небольшую часть всего множества: число пиктограмм столь же невелико по сравнению с множеством компьютерных программ, как и число строк в меню школьной столовой по сравнению с аппетитом детей. — Дело в том, что рабочий стол вместе с представленными на нем изображениями программ часто называют *пиктографическим меню*. Слово “меню” используется в информатике в более широком смысле, чем в повседневной жизни: меню — это выбор из некоторого ограниченного числа возможностей. Так, с точки зрения информатики расписание самолетов в аэропорту — это меню, оглавление сборника рассказов — это меню, да и список блюд в вашей школьной столовой — тоже меню. Мы с вами повидаем много разных меню: и в форме горизонтальных полосок, и в виде вертикальных столбцов, и, как сейчас у нас с вами, в виде набора пиктограмм, рассыпанных по рабочему полю.

— А как выбрать ту пиктограмму, которая нам нужна? — спросил Тим.

— Для этой цели используется *курсор*, — сказал дедушка.

Рукой, в которой он держал компьютерную мышку, дедушка сделал несколько плавных движений влево, вправо, вперед, назад, перемещая мышку по маленькому коврику. Небольшая остроносая стрелочка, словно повторяя перемещения мышки, забегала по экрану.



— Вот эта стрелочка и есть *курсор*. Он указывает активную область на экране, то есть то место, в котором сейчас будут развиваться события. Значит, если, например, вы хотите выбрать пиктограмму с надписью “Зимние вечера” (вот она — пиктограмма в виде окошка, которое смотрит из теплой комнаты во вьюжную тьму), надо поставить курсор на это изображение. В различных программах курсор может иметь разную форму. Не удивляйтесь, когда увидите, что в качестве указателя активного места экрана порхает бабочка или бегают слоненки.



Зимние вечера

— Почему же, дедушка, ты поставил курсор на пиктограмму зимнего вечера, но ничего не произошло? Я-то надеялся, что сейчас начнет работать программа, которая называется “Зимние вечера”?

— Ты очень торопишься, Тим, — заметил дедушка Фёрстов. — Ведь ты только что спросил: как выбрать нужную программу? Вот я тебе и показал, как делается выбор. Курсором. Надо подвести курсор на пиктограмму и щелкнуть левой кнопкой мышки.

— Щелкнуть? — недоверчиво спросил Тим. До сих пор он щелкал только орешки. Да еще иногда давал щелчки по лбу своим приятелям, когда они во дворе спорили на щелчки.

Дедушка спокойно разъяснил:

— Щелчком называют мягкое и короткое нажатие на кнопку мыши. Я часто буду говорить вам: “щелкните левой кнопкой”, “щелкните правой кнопкой”. Это означает нажатие пальцем на кнопку. Удобно указательным пальцем выполнять щелчок левой кнопкой, а средним — правой.

Так вот, — продолжил дедушка, — когда, выбрав пиктограмму, ты щелкаешь мышкой, пиктограмма чуть-чуть меняет цвет, подсвечивается. А на твой второй вопрос — как запустить программу на выполнение — и ответ будет другой.

— Так какой же? — Не торопиться Тим не мог, такой уж он неугомонный.

— Чтобы запустить программу, надо в тот момент, когда курсор находится на ее пиктограмме, а сама пиктограмма подсвечена, нажать на клавишу выполнения. (Видите вот эту большую клавишу в правой части клавиатуры с английской надписью *Enter*, что означает “войти”?) Впрочем, ту же задачу — запустить программу — выполняет и двойное нажатие левой кнопки мыши. Это даже, пожалуй, удобно. Ведь вы сейчас только что управляли мышкой, устанавливая курсор на выбранную пиктограмму. Значит, ваши пальчики еще не оторвались от мышки, этим можно воспользоваться и дважды щелкнуть левой кнопкой мышки.

Вопросы

1. Что такое пиктограмма?
2. Что такое меню?
3. Для чего нужен курсор?
4. Как можно управлять курсором?
5. О каких правилах техники безопасности дедушка разговаривал с внуками?

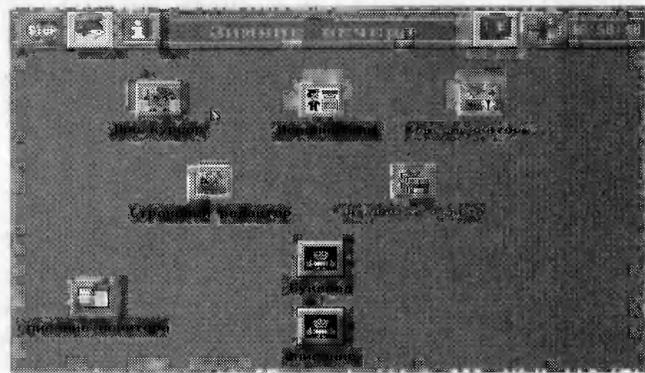
“Корешки и верхки”

— Можно я попробую щелкнуть кнопкой? — попросила дедушку Дина. — Мне хочется запустить нашу программу.

— Что ж, нам пора начинать. Давай, Диночка!

И дедушка Фёрстов чуть подвинулся, уступая внучке более удобное место около компьютера. Девочка осторожно обхватила мышку своей ладошкой, положила указательный палец на левую кнопку, дважды нажала на нее и тут же отпустила.

Не успел дедушка даже похвалить Дину за ее правильные действия, а во весь экран уже развернулось широкое окно, в верхней части которого было крупно написано то же название, что и на маленькой пиктограмме: “Зимние вечера”.



— Мне кажется, — сказала Дина, — что мы из одного рабочего поля попали в другое. Только пиктограммы на нем крупнее.

— Так оно и есть, — подтвердил дедушка, — именно здесь, в рабочем поле с названием “Зимние вечера”, мы и будем с вами делать все наши действия — выбирать нужные программы, входить в них, работать с ними.

— А теперь моя очередь! — сказал нетерпеливый Тим. Но, взяв в руки мышку, вынужден был спросить: — Куда же теперь идти?

Дедушка сказал:

— Мы только что разговаривали про курсор. И тут как раз есть пиктограмма именно с таким названием. Видишь, она чуть подмигивает.

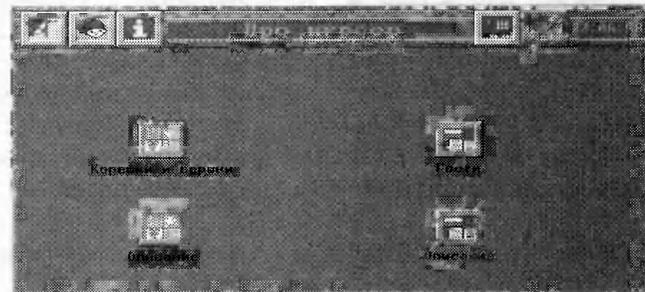
Тим прочитал “Про курсор”.

— Ясно: сюда.

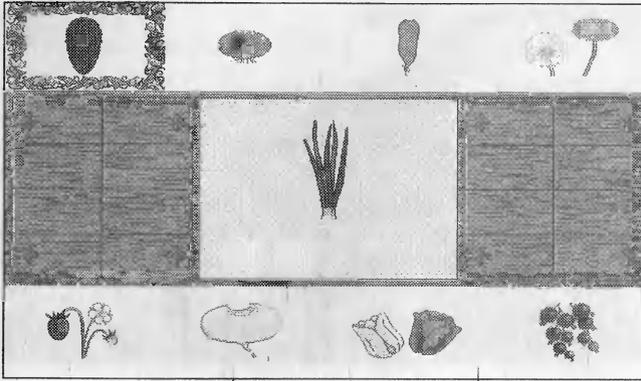
И опять раскрылось поле-экран.

— Слово матрешка, — прокомментировал Тим. — Все глубже и глубже. Но тут-то мы, кажется, добрались до программ.

Он уверенно подвел курсор на пиктограмму “Корешки и верхки” и еще раз щелкнул мышкой.



По краям экрана расположены части растений — словоя шишка, луковица, морковка, ягоды смородины, одуванчик, земляничка, красные гроздья рябины, репа, красивые цветы тюльпанов, а в самом центре — окошко, в нем видны зеленые стрелы лука.



Вокруг растения, нарисованного в левом верхнем углу, изображен прямоугольник, составленный из листьев и цветов.

— Он похож на веночек! — решила Дина.

— Вот так выглядит курсор в этой программе. Как и всякий курсор, он может передвигаться по экрану. Только мышка нам сейчас не поможет. Курсор в этой программе можно передвигать только с помощью клавиш — управляющих стрелок .

Пока дедушка Фёрстов рассказывал про курсор, Тим догадался, что требуется сделать в этой программе.

— Здесь надо найти ту часть растения, которая подходит к изображению в центральном окошке!

— Верно! — подтвердил дедушка. — Так какую же часть можно подобрать к этим зеленым перьям из окошка?

Дина, которая не раз помогала маме приготовить салат с луком, ответила мгновенно:

— Надо выбрать вот эту луковицу! А как?

Дедушка подсказал:

— Пользователь указывает компьютеру свой выбор тем, что ставит курсор-веночек на выбранное изображение и нажимает клавишу выполнения.

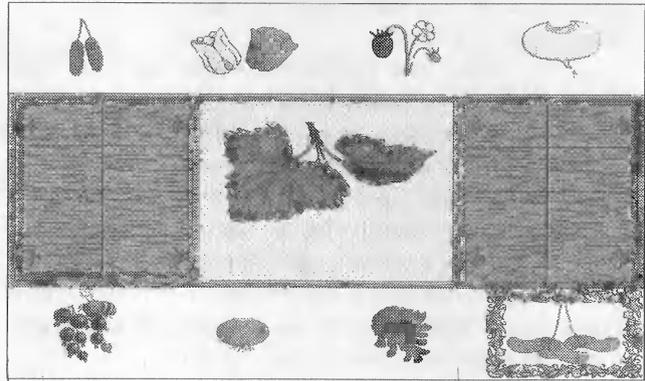
— Почему ты, дедушка, назвал меня “пользователем”?

— Уж не обиделась ли ты, Диночка? *Пользователем* называют любого человека, который работает с компьютером, обращается к нему, пользуется его услугами.

Дина поступила по дедушкиному совету. Тут же заиграла веселая музыка, раскрылись ставни слева и справа от окошка и стало видно огород, в котором растет лук.

А вот следующее задание оказалось для Дины не столь простым. Действительно, в центральном окошке экрана появился листок, который Дина не узнала. По краям экрана в этот раз разместились ягоды земляники, кленовые крылышки-семена, яркие тюльпа-

ны, репа, усатая луковица, березовые сережки и смородина. Дина сначала попробовала соединить листок с земляничкой. Не получилось: ставни не открылись. Не повезло и тогда, когда она выбрала кленовые крылышки.



А когда Дина ошиблась в третий раз, решив, что листок должен принадлежать березе, то компьютер выдал подсказку. Но получилось это так интересно, что ребята обрадовались этой живой подсказке не меньше, чем правильному решению: прилетел жужжащий шмель и усеялся на ягоды смородины, явно указывая, что надо было выбирать. Дина поступила так, как ей посоветовал шмель, и под звуки компьютерной музыки открылись ставни, за которыми дети увидели цветущий сад.

Дина и Тим с удовольствием поиграли в “Корешки и верхки”, хотя задачи этой программы и не показались им очень трудными.

Вопросы

1. Какую форму имеет курсор в программе “Корешки и верхки”?
2. Как управляют курсором в программе “Корешки и верхки”?
3. Как представлено меню в этой программе?
4. Как компьютер указывает ошибку пользователя?

Как не заблудиться в городе

Дедушка Фёрстов взглянул на часы, намекая, что дети уже засиделись за компьютером. Но и Тим, и Дина смотрели на дедушку таким умоляющим взглядом, что он понял: еще без одной программы сегодня не обойтись.

И он начал рассказ:

— Ну, хорошо. Вот вам еще одна компьютерная история: про обезьянку, которая пригласила в гости своих друзей... — Но тут же прервал себя: — Вы сами увидите эту историю на экране компьютера.

Тим поспешил заявить:

— Только теперь я буду у компьютера.

Дина благосклонно согласилась, дедушка кивнул головой. Тим задал первый вопрос:

— А где пиктограмма этой программы?

Дедушка показал нарисованную на экране маленькую записную книжку, рядом с которой можно было прочитать надпись: "Гости".

— Ага, понимаю, — сказал Тим, — записная книжка выбрана в качестве пиктограммы, потому что в эту книжку обезьянка записала гостям свой адрес. Так?



Гости

Дедушка ответил:

— Не совсем так. Наша обезьянка еще не умеет писать. Но она нарисовала в записную книжку, как могла, дом, в котором она живет. По этому рисунку тебе и предстоит найти дом обезьянки.

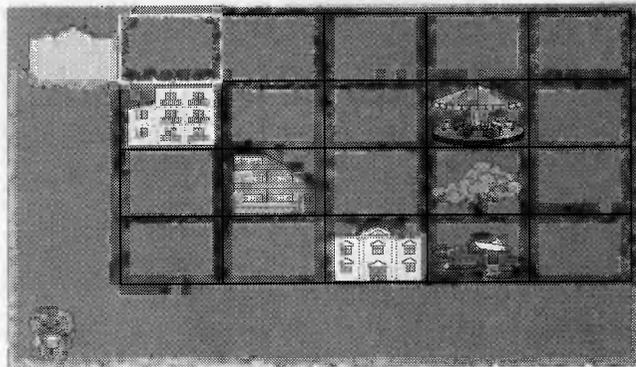
— А что значит "как могла"?

— Обезьянка живет в городе, где каждый дом отличается от других. Но обезьянка нарисовала только силуэт дома, его очертания. Вам прежде всего потребуются по силуэту определить на плане города нужный дом.

Тим щелкнул по пиктограмме "Гости".

— Ой! А в этой программе курсор — это симпатичный рыжий львенок.

догадался, что выбор домика следует отметить с помощью курсора — красной квадратной рамочки размером с клеточку плана. Эта красная рамочка легко слушалась управляющих стрелок. Поэтому Тим поставил курсор-рамочку на нужный домик и нажал клавишу выполнения.



Однако оказалось, что мало просто найти дом на плане. Надо выбрать, в каком направлении идти к домику: сначала вправо, потом вверх или наоборот, сначала вверх, потом вправо. Для этого выбирают одну из управляющих стрелок —  или . Когда стрелка выбрана, надо сосчитать, сколько кварталов-клеточек надо пройти по городу в выбранном направлении, и набрать это число в верхнем цифровом ряду клавиш. Если, например, на плане, показанном на предыдущем рисунке, Тим сначала выбрал стрелку , то ему надо ответить числом 3. После этого надо нажимать стрелку  и на месте вопросительного знака поставить число 1.

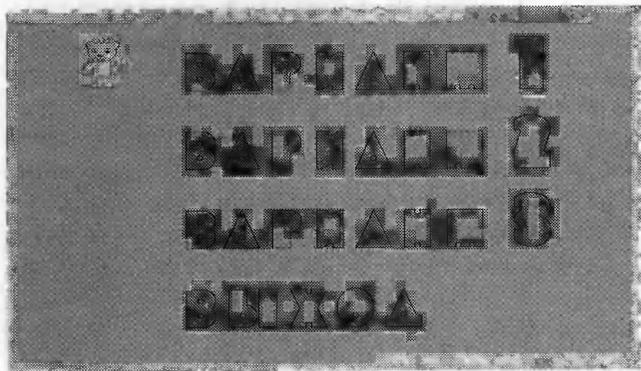
Поясняя детям правила игры, дедушка Фёрстов сказал:

— Вы видите, что положение домика на нашем плане всегда можно точно указать с помощью двух чисел. Первое из них указывает, сколько кварталов-клеточек надо пройти до домика, двигаясь в горизонтальном направлении слева направо, а второе — сколько клеточек следует пересечь снизу вверх. Числа, однозначно определяющие положение домика, называют *координатами*. Координаты очень важны в жизни: ими пользуются математики и географы, космонавты и геологи. И в наших компьютерных программах координаты встретятся еще не раз.

Тим ввел в компьютер координаты домика обезьяны.

Дальше начался спектакль. Несколько раз действие прерывалось, и мальчику, для того чтобы двинуться дальше, пришлось решать несложные арифметические задачи. Конечно, Тим и внимательно наблюдавшая за его операциями сестра легко решили эти задачи.

Гости расселись за столом. Обезьянка решила угостить их чаем и начала ставить чашки на поднос. Сколько же чашек надо поставить? Пришлось внимательно пересчитать число гостей (включая и обезьянку: она ведь тоже будет пить чай со своими друзьями).



Дедушка уточнил:

— Каждый раз, когда ты будешь запускать эту программу, курсор будет выглядеть по-разному — то львенок, то медвежонок, то слоненок. Это все гости обезьянки. А управлять этим курсором — заставлять его ходить вверх и вниз по левому краю экрана — можно с помощью стрелок  и .

— Дедушка, смотри-ка: на экране меню.

И Тим прочитал слова в четырех позициях меню:

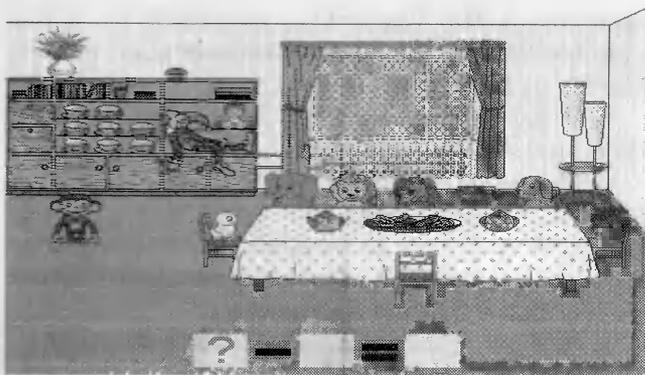
Вариант 1, Вариант 2, Вариант 3, Выход.

— Работа программы несколько отличается, в зависимости от того, какой вариант ты выберешь.

Для того чтобы войти (с одобрения дедушки) в первый вариант программы, Тиму даже не пришлось передвигать курсор-львенок — тот уже стоял около надписи *Вариант 1*. Оставалось только нажать клавишу выполнения.

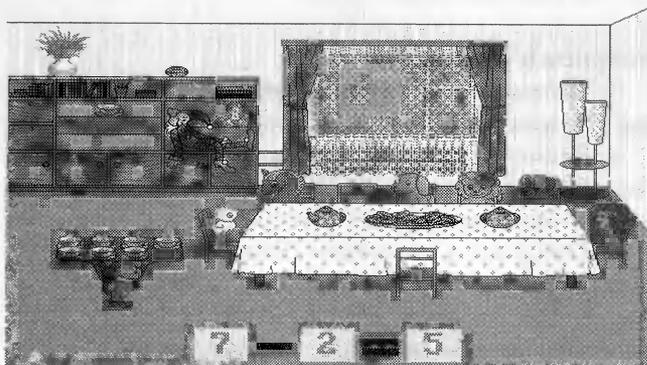
Теперь львенок стоял в уголке, словно ожидая дальнейших действий Тима. На экране Тим увидел расчерченный на квадраты план города, а в левом верхнем углу — силуэт домика, в который обезьянка пригласила гостей.

Найти среди трех домиков небольшого городка тот, куда надо послать львенка, Тиму было нетрудно. Он



Но обезьянка поставила на поднос только часть нужных чашек. Сколько чашек стоит на подносе у обезьянки? Тим сосчитал чашки, а на экране компьютера уже готов следующий вопрос: сколько чашек надо добавить? И когда все вычисления были сделаны правильно, обезьянка с подносом, заполненным чашками, направилась к столу угощать гостей. А компьютер заиграл веселую песенку:

*...С голубого ручейка
Начинается река,
Ну а дружба
Начинается с улыбки.*



Вопросы

1. Какую форму имеют курсоры в программе "Гости"?
2. Как выглядит меню в программе "Гости"?
3. Что такое координаты?

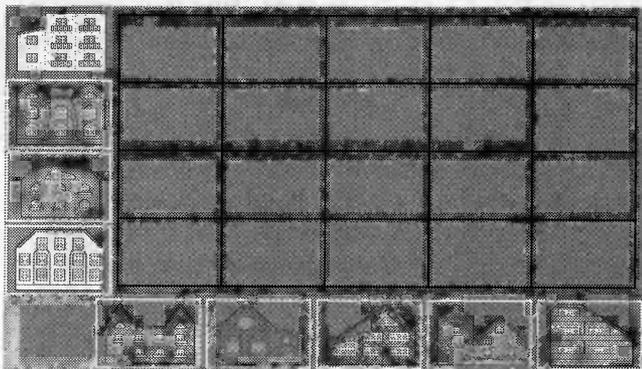
Как построить задачу

Вернувшись в самое начало программы, дети решили посмотреть, как работает второй вариант программы. Они без труда определили особенности этого варианта: город стал больше; теперь это не крохотный городок из трех зданий, а поселок из девяти разных домов. Конечно, задача становится более сложной, но все правила сохраняются. Поэтому Дина, сменившая Тима у компьютера, справилась с приемом многочисленных гостей без особого труда.

А вот третий вариант оказался неожиданным.

Прежде чем решать задачу про встречу гостей, здесь предстоит ее (задачу!) построить. Весь строительный материал находится под рукой. План города из 20 клеточек

чек (5 × 4) готов. Правда, все клеточки пусты, ни одного здания на плане нет. Все здания, которые надо поместить на план, размещены по краям экрана, справа и внизу. На одном из них — курсор в виде синей рамочки.



Как только вы нажимаете клавишу выполнения, отмеченное этим курсором здание попадает на клетчатое поле плана. Теперь это здание окружено красной рамкой. Это — новое представление курсора, который можно перемещать по полю и, выбрав желаемое место, нажать клавишу выполнения. Так на плане города появляется здание. Таким же образом надо поставить остальные здания, а затем еще и клумбу, тележку с мороженым, лужу.

Город готов. Вот по краям клетчатого поля появляются девять силуэтов. После выбора одного из них в левом нижнем углу появляется приглашенный в гости зверек. Можно начинать игру (как во втором варианте).

Рассказав детям о том, как можно в третьем варианте программы самостоятельно соорудить задачу, дедушка Фёрстов попросил Дину построить город для Тима, а потом — Тима подготовить задачу для Дины. Ребятам понравилось строить задачу не меньше, чем ее решать.

Наконец, настала пора попрощаться с компьютером (хотя убедить в этом внуков дедушке было нелегко). Дедушка вышел из программы и выключил компьютер. Хотя Дина и Тим пытались уследить за его действиями, они не смогли их запомнить. Но дедушка им сказал:

— У вас еще будет время поупражняться в выключении компьютера. Сегодня я сделаю это сам. А сейчас пока бегите к бабушке. Она приготовила вам заслуженный чай с пирожками.

Вопросы и упражнения

1. Какие формы у курсоров в третьем варианте программы "Гости"?
2. Вызовите третий вариант программы "Гости" и сами постройте план города. Поставьте домик обезьянки в поле с координатами (2,4). Решите построенную задачу.
3. А теперь постройте новую задачу так, чтобы обезьяний домик находился на поле (1,1).

Вечер второй

Что такое алгоритм

Нужен четкий план

Когда на следующий день Дина и Тим пришли в дедушкин кабинет, он почему-то не пригласил ребят к компьютеру, а задал внучке вопрос:

— Дина, умеешь ли ты варить манную кашу?

— Да, конечно, — гордо ответила Дина, которой не раз приходилось помогать маме на кухне.

— Попробуй-ка прямо сейчас рассказать мне рецепт приготовления манной каши, а я его запишу.

И дедушка Фёрстов, приподняв свои пальцы над клавишами, приготовился записывать рецепт под диктовку Дины.

— Вкусную манную кашу, — начала Дина, — готовят на молоке. Наливают в кастрюлю примерно литр молока. Доводят его до кипения (только за молоком надо следить внимательно, чтобы оно не “убежало”), а потом насыпают тонкой струйкой манную крупу. Полстакана. При этом надо равномерно помешивать молоко ложкой, чтобы каша получилась без комков. Вслед за крупой надо насыпать еще немного сахара и чуть-чуть посолить. Как только в кипящей каше начнут появляться пузырьки-“вулканчики”, можно выключать огонь. В кашу надо добавить масло. Лучше сливочное. Так будет вкуснее. А после этого — накладывайте кашу на тарелку и кушайте! В нашей семье очень любят такую кашу.

Дедушка Фёрстов, быстро-быстро шелкая по клавишам, все успел записать. Однако Дине показалось, что дедушка не совсем доволен ее рассказом. Дедушка сказал:

— Дина в самом деле несколько раз варила манную кашу (потому-то я ее об этом и спросил). А вот ты, Тим, сможешь ли по рецепту Дины накормить нас семейным блюдом?

Тим приник к экрану, чтобы внимательно перечитать рассказ сестры, и начал критически оценивать ее рецепт:

— Конечно, все написано понятно, по-русски. Только мне было бы трудно воспользоваться этим текстом. Это не план приготовления каши, а какой-то рассказ про чувства и настроения.

Дина обиделась:

— А что тебе не нравится?

— Вот ты говоришь “насыпать немного сахара”. А может, можно не песок сыпать, а положить кусочки рафинада? А кроме того, что значит “немного”? Это стакан? Или столовая ложка? Или, может быть, чайная ложечка? А уж твое выражение “чуть-чуть посолить”! Что такое “чуть-чуть”? И сколько масла надо взять, чтобы положить в кашу? Я этого не знаю.

Дедушке пришлось признать:

— Пожалуй, Тим прав. Я верю, что Дина может приготовить кашу, но твое описание, Дина, страдает неточностями. Поэтому воспользоваться таким описа-

нием пока нельзя. Рецепт надо записывать так, чтобы с его помощью блюдо мог приготовить даже человек, никогда не варивший кашу.

— И как бы ты это сделал? — все еще обиженно проговорила Дина.

— Например, так:

1. Взять 1 литр молока.
2. Довести молоко до кипения.
3. Насыпать тонкой струйкой полстакана манной крупы, постоянно помешивая.
4. Добавить три чайные ложки сахарного песка.
5. Добавить щепотку соли.
6. Варить 15 минут.
7. В готовую кашу добавить столовую ложку сливочного масла и размешать.

И дедушка Фёрстов заключил:

— Давайте впредь стараться записывать план наших действий в строгом, точном порядке. План решения задачи, написанный так, что его может выполнить тот, для кого он написан, называется *алгоритмом*. Тогда можно говорить и об исполнителе алгоритма: это тот, кто умеет прочитать и выполнить каждый пункт такого плана. У нас записан алгоритм приготовления манной каши для исполнителя Тим. Этот же план не будет алгоритмом для исполнителя Вася, который не знает, что означает слово “литр”.

Вопросы и упражнения

1. Почему Тим критически отнесся к рецепту сестры? Почему рецепт Дины нельзя назвать алгоритмом?

2. Напишите рецепт приготовления яичницы из трех яиц. Постарайтесь сделать описание таким, чтобы его можно было называть алгоритмом.

3. Придумайте не меньше пяти физических упражнений утренней зарядки. Опишите алгоритм такой зарядки. Постарайтесь его выполнить. (Кстати, такой алгоритм полезно выполнять каждый день.)

Перевоз через реку

Дедушка Фёрстов, который только что обсуждал с ребятами алгоритм приготовления манной каши, вдруг круто сменил тему разговора:

— Хотите, я расскажу вам небольшую сказку?

Дина и Тим ожидали рассказа о новых программах, но от неожиданного предложения про сказку все-таки не отказались.

Возвращался старик с базара домой. Вел с собой волка и козу и нес на плечах кочан капусты. Повстречалась на пути речка.

Глубока речка, а моста нет.

Вдруг видит старик: стоит у берега лодка. Обрадовался. Только уж очень мала лодка: старик может в нее сесть один или взять с собой только одного из “пассажиров” — либо волка, либо козу, либо капусту. Втроем в лодке не поместиться.

Переехать на другой берег было бы нетрудно, да вот беда: уж очень недружны “пассажиры”. Нельзя

старика оставлять волка и козу вместе: волк съест козу. Козу с капустой тоже нельзя оставлять наедине: коза съест капусту. Это только со стариком они все такие смирные.

Сел старик на берегу, задумался: “Трудна задача...”

Дина и Тим не стали спорить с персонажем дедушкиной притчи: каждый раз, когда они пытались найти решение задачи, кто-нибудь из героев сказки поедал другого. Дина, которая очень не любит кроважанные истории, предложила:

— Давайте перед переездом на другой берег накормим волка так, чтобы он на козу и смотреть не захотел...

А у Тима появилась совсем другая мысль:

— Можно сделать так: насадить вилок капусты козе на рога. Тогда капуста и место занимать не будет, и коза ее не достанет, как бы голодна она ни была.

Дедушка Фёрстов только посмеялся:

— Вы сами отлично понимаете, что ваши предложения — это нарушения правил задачи. Строгое соблюдение правил — главное условие построения алгоритма.

— Значит, и здесь надо построить алгоритм перевоза через реку? — воодушевился Тим.

— Конечно! — подтвердил дедушка.

— Тогда, — догадалась Дина, — надо с самого начала договориться о формальном способе записи всех действий перевозчика.

И чуть смущенно добавила:

— Чтобы не получилось тех неточностей, которые вы у меня увидели в рецепте манной каши.

— Замечательная мысль, — похвалил дедушка.

Дина, вдохновленная поддержкой, продолжала:

— Например, если перевозчик должен переезжать с левого берега на правый, достаточно нарисовать стрелку слева направо. Очень короткое обозначение. Всем понятное. И не надо придумывать длинные фразы, которые разные люди могут понимать по-разному. Да и ошибиться в таких длинных фразах очень легко.

Тим не стал спорить: сестра высказала понравившуюся ему мысль. Он подхватил идею об обозначениях в алгоритме переезда и сказал:

— А рядом со стрелкой, например справа от нее, можно писать одно из слов — “волк”, “коза” или “капуста”. Тогда сразу станет ясно, кого должен везти с собой перевозчик.

— Молодцы, — одобрил дедушка Фёрстов. И тут же предложил детям разобраться в такой ситуации:

— А как в таком случае описать положение, когда старичок-первозчик едет на другой берег один?

— Разве такое может быть?! — в один голос воскликнули ребята.

— А почему бы нет? — вопросом на вопрос ответил дедушка Фёрстов. — Вот посмотрите. Первое действие: перевоз козы с левого берега на правый. Следуя вашим предложениям, я напишу:

→ КОЗА

Ребята молча кивнули: все верно, коза с перевозчиком на правом берегу, а волк с капустой (что совсем не опасно) остались на левом.

— Что делать дальше?

— В самом деле, — согласился Тим, — сейчас перевозчик может оставить козу одну на правом берегу и в одиночку плыть обратно на левый берег за кем-нибудь другим.

Ребята задумались: как бы обозначить такое совершенно правильное действие перевозчика? Дедушка подсказал:

— Раз перевозчик никого с собой не взял, то ничего около стрелки писать не надо. Вот так:

←

В этом случае достаточно показать только направление переезда. Ну, теперь, когда все обозначения согласованы, сможете записать алгоритм?

Дина и Тим, склонившись над листком бумаги, дружно обсуждали алгоритм переезда. Получилось не сразу. Но, наконец:

— Ура!

— Покажите-ка.

Дедушка Фёрстов был очень доволен, когда увидел четкое описание алгоритма:

→ КОЗА

←

→ ВОЛК

← КОЗА

→ КАПУСТА

←

→ КОЗА

— Все верно. Только это не единственный алгоритм, который решает поставленную задачу.

— Есть еще один способ? — удивился Тим.

— Да.

— Что ж, поищем?

Вопросы и упражнения

1. Может ли перевозчик поплыть на лодке один, оставив на берегу всех троих — волка, козу и капусту?

2. Найдите еще один способ решения поставленной задачи.

3. Когда к Тиму и Дине приехал в гости их двоюродный брат Сережа и ребята рассказали ему притчу о перевозчике, волке, козе и капусте, Сережа предложил внести в обозначения действий алгоритма такие изменения:

1) вместо стрелок писать слова ВЛЕВО или ВПРАВО;

2) обозначать участников переезда не целыми словами, а только первыми буквами этих слов;

3) поездку старика-первозчика обозначать не пустым местом, а буквой “п” (первой буквой слова “первозчик”).

Какие из предложений Сережи можно принять, а с каким нельзя согласиться? Почему?

4. Вот еще одна задача про переправу через реку.

На одном берегу лодка, два мальчика и солдат. На лодке могут плыть два мальчика или один взрослый.

Как солдату переправиться на противоположный берег, а лодку вернуть назад?

5. А вот задача чуть-чуть посложнее:

Три рыцаря и три оруженосца подошли к берегу реки. У берега привязана лодка на двух человек. Каждый оруженосец так предан своему хозяину, что отказывается оставаться на берегу или в лодке с другими рыцарями в отсутствие хозяина (но готов быть один или с другими оруженосцами). Как им всем переправиться через реку?

Придумайте обозначения для действий алгоритма и запишите его.

Как задача о перевозе решается на компьютере

Когда ребята решали задачу про перевоз волка, козы и капусты, дедушка советовал им изображать положение на берегах реки с помощью предметов, которые находились под рукой: река — это лист бумаги, лежащий посередине стола, лодка — это блюдо, перевозчик — кофейная чашечка, волк — дедушкина авторучка, коза — карандашик, а капуста — розовый листик.

Дедушка хотел даже объяснить, что эти предметы — только условные обозначения, а вовсе не настоящие персонажи сказки, но ребятам это условное соглашение было настолько понятно, что Тим сам взялся объяснить:

— Ничего тут удивительного нет. Надо же как-нибудь обозначить тех, кто участвует в перевозе. Важно не то, как мы обозначим волка или козу. Главное — договориться, как мы сами будем понимать наши предметы, которые понарошку изображают волка и козу. Мы всегда так играем!

Зато и у Тима, и у Дины возник другой вопрос. Хотя ребята с интересом отнеслись к задачам дедушки Фёрстова о переправе через реку, Тим с некоторым огорчением поглядывал в сторону компьютера, экран которого призывно светился.

— Дедушка, — спросил Тим, — а почему ты нас пригласил в свой кабинет, где стоит компьютер? Ведь все эти задачки мы могли решать и в нашей комнате, где стоит письменный стол.

— Про все задачки я говорить не стану, но одну из них вы можете решить и на компьютере.

— Правда? — воодушевленно воскликнули Дина и Тим, усаживаясь около дедушкиного компьютера.

— Я уже вижу пиктограмму “Перевозчик”. Вот она: на ней нарисован дедушка с лодкой и веслом. Это наша задача?

— Да, Диночка. Войди в эту программу.

Девочка подвела курсор к пиктограмме “Перевозчик” и щелкнула левой кнопкой. Ребята, конечно, сразу узнали задачу про волка, козу и капусту — так точно они были изображены на экранной картинке.

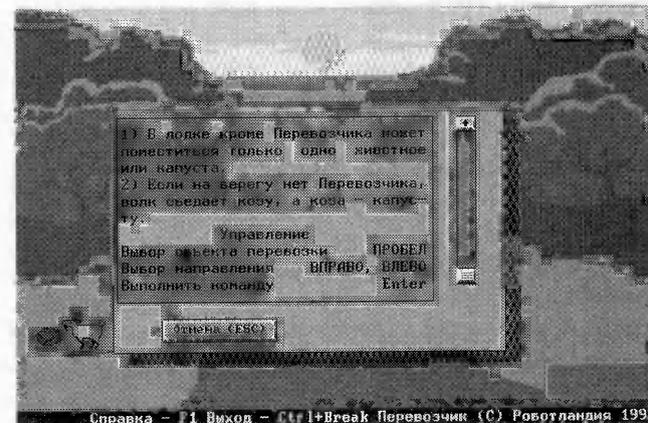


Но дедушка Фёрстов обратил внимание детей не на знакомые изображения — река, солнышко, берег, лес и известные уже персонажи, а на синюю полосу в нижней части экрана.

— Это *информационная строка*. В этой строке вы можете прочитать и имя программы, и название фирмы-изготовителя. А вот еще два важных указания, которыми мы сможем пользоваться не только работая с “Перевозчиком”, но и обращаясь ко всем другим программам. Вот одно из них: “**F1** — справка”.

Это означает, что, нажав на клавишу **F1** (вот она, в верхнем ряду клавиш на клавиатуре), вы получите в появившемся на экране окошке информацию о том, как можно управлять программой, а также о правилах этой задачи.

Дина тут же нажала **F1**, и ребята действительно прочитали информацию о задаче. Правда, кое-что дедушка Фёрстов уже рассказывал о “Перевозчике”, но о клавишах, которые используются при управлении “Перевозчиком”, Дина прочитала впервые. Для этого ей пришлось нажать на стрелку **↓**, после чего указатель в правой части окна стал спускаться. Окошко словно пошло вниз, открывая ранее невидимые строки. (“Это *полоска протяжки*, — пояснил дедушка Фёрстов. — Она позволяет прокручивать длинный текст справочной информации, который не помещается в окне”):



— А вот еще одна надпись в информационной строке, — сказал Тим. — Тут написано: “Выход из программы — **Ctrl** + **Break**”.

— Это значит, — ответил дедушка Фёрстов, — что выйти из программы можно, нажав аккорд двух клавиш — **Ctrl** и **Break**. Во всех описаниях программ (и на наших занятиях тоже) аккорд записывается названиями клавиш, которые соединены знаком “+”. Посмотрите, где расположены на клавиатуре клавиши нашего аккорда.

В разговор вступила Дина:

— Что такое “аккорд”, я знаю (Дина недавно начала заниматься в музыкальной школе). Только аккорд этих двух клавиш мне кажется неудобным: уж очень далеко расположены эти две клавиши друг от друга.

— Осторожно, Дина, — предупредил дедушка. — Аккорд на клавишах компьютерной клавиатуры набирают совсем не так, как на фортепиано. Музыкальный аккорд — это обязательно одновременное нажатие нескольких клавиш. В информатике же, набирая аккорд, надо сначала нажать первую клавишу, а затем, не отпуская ее, нажать вторую. Следите за этим внимательно. И не забывайте, что все нажатия на компьютерные клавиши должны быть мягкими и короткими.

— Можно попробовать?

— Конечно, можно. Правда, тогда мы выйдем из программы. Но ведь теперь для вас не составит труда снова в нее войти.

Пока Дина пробовала нажать аккорд **Ctrl** + **Break**, дедушка Фёрстов объяснил, почему этот аккорд так непросто набирать.

— Это сделано специально: трудный аккорд невозможно нажать нечаянно. Значит, даже неловкий новичок может чувствовать себя спокойно — программа не прервется случайно во время работы.

Тим, внимательно следивший за действиями Дины, сказал:

— Теперь, когда мы прочитали справку-подсказку, начнем.

И, обращаясь к Дине, добавил:

— Нажми-ка стрелку : нам ведь надо ехать на правый берег.

Толстая красная стрелка высветилась в небольшом светло-сером окошечке в правом нижнем углу экрана.

— Это поле ввода команд, — пояснил дедушка. — Все команды, которые вы напишете для “Перевозчика”, появляются в этом окне.

— Теперь надо выбрать, кого перевозить. В том алгоритме, который мы сейчас написали на листке бумаги, первой надо было посадить в лодку козу. В инструкции-справке говорилось, что выбирать объект для перевозки надо нажатием на клавишу пробела. Ну-ка...

Дина нажала на ПРОБЕЛ. Но в поле ввода команд рядом со стрелкой появился волк.



— Ой! — от неожиданности воскликнула Дина. Но тут же нажала ПРОБЕЛ еще раз. И снова не то: волк заменил кочан капусты. Только после следующего нажатия на ПРОБЕЛ в поле ввода оказалась нужная команда: коза со стрелкой .

— Когда команда набрана в поле ввода, ее надо передать программе “Перевозчик” для выполнения... — начал дедушка, а закончил эту фразу Тим, запомнивший строки справки-инструкции:

— Для этого надо нажать клавишу **Enter**. Не поэтому ли она называется клавишей выполнения?

— Правильно, — сказал дедушка Фёрстов. — Вы уже умеете набирать команды. Запишите весь алгоритм перевозки. И следите за сообщениями, которые программа будет выводить на экран.

— Что это за сообщения?

— Если ваш алгоритм верный и вы не допустили ошибок при наборе команд, то в конце вы получите поздравление с успехом белыми буквами на нежном голубом фоне. Если же ваша ошибка приведет к печальному исходу — волк съест козу или коза скушает капусту, — то ужасное слово “Съели!” вы увидите в прямоугольнике тревожного красного цвета. Придется начинать составление алгоритма с самого начала. Итак, решите знакомую задачу, а я сделаю короткий телефонный звонок.

Дина начала свою самостоятельную работу. Однако ей скоро пришлось остановиться, потому что на экране появилось сообщение, о котором дедушка не упоминал: в красном прямоугольнике ярко горела надпись “Не могу”. Дина напугалась и решила подождать, когда подойдет дедушка, но Тим спокойно сказал ей: “Дай-ка я посмотрю”. И почти сразу же воскликнул:

— Все ясно! На каком берегу стоит у тебя сейчас перевозчик?

— На правом, конечно.

— А какую команду ты ему дала?

Дина взглянула на поле ввода команды и смущенно произнесла:

— НАПРАВО. У меня тут стоит правая стрелка. Это я нечаянно нажала не ту стрелку.

— Так вот. Старичок перевозчик стоит на правом берегу, а ты отправляешь его направо. Понятно, что он тебе ответил “Не могу”.

Вопросы и упражнения

1. Как прочитать справочную информацию о программе?
2. Какими клавишами управляется “Перевозчик”?
3. В каком месте экрана записываются команды “Перевозчику”?
4. Чем отличаются аккорды в музыке и в информатике?
5. Сравните придуманные ребятами обозначения для команд алгоритма переправы с теми обозначениями, которые используются в программе “Перевозчик”.
6. Какие сообщения “Перевозчик” может выдавать на экран?

Вечер третий

Строковый редактор

Какие бывают ошибки

Начиная очередную беседу с внуками около компьютера, дедушка Фёрстов заботливо взял в руки клавиатуру:

— В программах, с которыми мы успели познакомиться, использовалось совсем немного клавиш. Пора уже выходить на простор клавиатуры. В сегодняшних наших упражнениях мы не пропустим ни одну из букв новых клавиш.

Чуть оробев, Дина сказала:

— Но ведь когда клавиш так много, очень легко ошибиться!

Дедушка Фёрстов успокоил ребят:

— Не ошибается только тот, кто ничего не делает. Поэтому мы и будем сегодня учиться исправлять ошибки.

Очень просто исправлять *оперативные* ошибки. Так называют ошибку, которая только что сделана и сразу же обнаружена.

Например, я хочу набрать слово “стол”, но я чуть промахнулся пальцем, и у меня получилось “стог”. Эту ошибку я увидел сразу же, как только буква “г” появилась на экране. Тогда я нажимаю клавишу, которая располагается чуть выше клавиши выполнения. На ней иногда рисуют стрелку справа налево, а иногда пишут буквы BS. Это начало английских слов *back space*, то есть “обратный пробел”. Нажатие на клавишу обратного пробела приводит к тому, что курсор сдвигается назад на одну позицию, стирая при этом только что написанный символ. Итак, ошибочная буква стерта (кстати, у клавиши `BackSpace` есть еще одно очень понятное название — “клавиша стирания”) и, значит, можно продолжать писать снова, на сей раз правильно.

— А разве бывают еще какие-нибудь ошибки, кроме оперативных? — спросил Тим.

— Конечно, бывают. Это так называемые *отсроченные* ошибки.

Например, ученик утром набрал на клавиатуре какой-то текст, а учителю приходится проверять этот текст после обеда. Все ошибки давно сделаны, и сей-

час, проверяя текст, надо их обнаружить и исправить. Среди отсроченных ошибок могут быть как ошибки клавиатурного набора (сделанные, например, из-за топорливости ученика), так и ошибки грамматические (сделанные потому, что ученик не знает, как надо писать то или иное слово).

Слушая это дедушкино объяснение, Тим слегка покраснел: вспомнил, как дедушка Фёрстов пожурил его за ошибку в домашней тетради по русскому языку.

Существует всего три типа отсроченных ошибок:

- 1) *неверный символ* — его надо исправить, набрав вместо него верный символ;
- 2) *лишний символ* — его исправляют удалением;
- 3) *пропущенный символ* — его исправляют вставкой нужного символа на место ошибки внутри слова.

— Ну, если разных типов ошибок всего только три, то их легко запомнить и научиться исправлять, — сказала Дина.

А дедушка Фёрстов порадовал ребят.

— Ты права. И я покажу вам сейчас программу, которая научит вас исправлять ошибки.

Вопросы и упражнения

1. Чем отличаются оперативные и отсроченные ошибки?

2. Каждое из приведенных ниже слов набрано на клавиатуре с одной ошибкой:

фудбол, пинжак, мидведь, карабль, заец

К какому типу ошибок они относятся? Как эти ошибки можно исправить?

3. К какому типу относятся ошибки в словах:

гостинница, сонце, маленький, чувство, ужастно?

Правилка показывает мультик

— А я догадался, что за пиктограмма обозначает программу, которую ты хочешь нам показать! — заявил Тим. — Вот эта.



Правилка

И Тим показал на маленький квадратик, в котором буква “и” в слове “кит” была перечеркнута, а над ней написано исправление — буква “о”.

— Правильно догадался. Называется эта программа “Правилка”.

— Войдем в нее? — спросил Тим.

— Конечно.

Уверенное нажатие на клавишу выполнения, и на экране появилась заставка программы “Правилка”: на одной странице раскрытой книги изображена вредная старуха Шапокляк (из-за которой и случаются все ошибки), а на другой...

— А что на другой? — задал вопрос дедушка Фёрстов, показывая на экран.

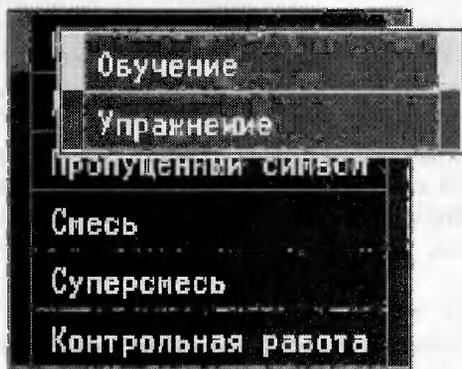
— Я думаю, что это меню программы “Правилка”, — не очень уверенно сказала Дина.

— Вот именно. В этом меню, как вы видите, шесть позиций. Прочитайте первые три из них — “Неверный символ”, “Лишний символ”, “Пропущенный символ”, — и вы заметите, что они совпадают с названиями тех трех типов ошибок, о которых мы уже говорили.

— Я начну с позиции “Неверный символ”, — решил Тим. — Мне этот тип ошибок кажется самым простым. Я вижу, курсор уже подсвечивает первую строку. Поэтому я сразу нажму клавишу выполнения.

Но все-таки, прежде чем нажать, вопросительно взглянул на дедушку. Тот согласно кивнул головой.

После такой долгой предварительной беседы ребята не терпелось поскорее увидеть “Правилку” в работе. Однако вместо этого они увидели поверх позиции “Неверный символ” еще одно меню. Правда, не такое большое. Всего из двух позиций — “Обучение” и “Упражнение”. Но не было никаких сомнений — это тоже было меню.



— Меню внутри меню!

— Что ж тут удивительного? — сказал дедушка Фёрстов. — Когда вы открываете книгу, вы видите меню ее основных разделов-глав. Каждая глава имеет свое меню — деление на разделы-параграфы. А в некоторых книгах параграфы делятся, в свою очередь, на разделы-пункты, и тогда получается, что и каждый параграф имеет свое меню. Иногда в таких случаях говорят, что меню параграфов представляет собой подменю главы. Мы можем сказать, что короткое меню — это подменю режима “Неверный символ” в “Правилке”. А большое меню из шести строк, с которого начинается программа, называют *основным меню*. Меню, включающее в себя несколько подменю, называют также *сложным меню*.

Двигаемся дальше!

— Значит, я вхожу в позицию “Обучение”? — сказал Тим, готовясь нажать клавишу выполнения.

— Прежде чем ты в нее войдешь, — успел прервать его дедушка Фёрстов, — я хочу вам рассказать, что вы сейчас увидите. Режим, в котором работает программа, когда тыходишь в позицию “Обучение”, называется

демонстрационным режимом. Так он называется потому, что программа, работая в этом режиме, показывает, демонстрирует, как нужно управлять программой, или, другими словами, как компьютер выполняет программу.

Такой режим очень полезен всякому, кто хочет познакомиться с программой, изучить, как она работает. Поэтому в подавляющем большинстве сложных программ или учебных программ всегда есть демонстрационный режим.

В демонстрационном режиме “Правилка” показывает, словно в мультфильме, примеры правильного исправления отсроченных ошибок. Никаких клавиш в этом режиме вам нажимать не придется. Надо просто внимательно смотреть на экран, запоминая и понимая, что делает компьютер.

Выслушав такой длинный рассказ дедушки, Тим с еще большим нетерпением вошел в позицию “Обучение”, чтобы посмотреть, как компьютер исправляет неверные символы. Главную идею “мультфильма”, который увидели Дина и Тим, можно записать так:

Для исправления неверного символа надо поставить курсор на этот символ и нажать клавишу с правильным символом.

Вопросы и упражнения

1. Что такое сложное меню? Приведите примеры сложных меню из повседневной жизни.
2. Для чего предназначается демонстрационный режим в программах? Почему так нужны демонстрационные режимы в сложных и учебных программах?
3. В чем состоит работа пользователя, который включил демонстрационный режим программы?

Упражнения “Правилки”

В окошке строкового редактора “Правилки” — слово с ошибкой:



Тим сразу делает исправление:



Столь же уверенно в следующем задании действует Дина. Слово



она уверенно исправляет:



И вдруг — неожиданная ситуация. Тим получил задание — исправить неверное слово



Конечно же, он знал, что это слово надо писать с буквой “с”, а не “з”. Но он немножко запутался в начале. Потом еще раз. Наконец, решил для простоты

стереть все слово (это ведь совсем нетрудно сделать, воспользовавшись, например, обратным пробелом) и написать заново все слово правильно.

Как же он удивился, когда компьютер, вместо того чтобы отметить, как обычно, “Нет ошибок”, сообщил: “Плохо исправлено”.

Дедушка Фёрстов и на этот раз пришел на помощь:

— Конечно, слово “хвост” у тебя сейчас написано верно. Но ведь ты сделал исправление совсем не по тому алгоритму, который тебе только что показал компьютер в демонстрационном режиме.

Когда Тим выполнил правильно пять заданий, программа высветила в середине экрана окошко с приятной надписью “Вам присвоено звание профессор” рядом с портретом толстенького и немолодого, но вполне симпатичного профессора.

Дина и Тим поменялись местами. В одном из заданий Дина допустила ошибку. Поэтому в конце серии из пяти заданий она получила звание Мудрого Кролика, а не профессора. Кролик, хоть и был назван мудрым, на экране выглядел веселым, даже немножко озорным, поэтому все (и даже Дина) невольно улыбнулись.

— Теперь мы умеем исправлять неверные символы, — сказала Дина.

— Можно посмотреть другие типы ошибок?

— Даже нужно!

Тогда Дина спустила подсветку-курсор на одну строку вниз в основном меню, остановившись в позиции “Лишний символ”.

— И сейчас опять я рекомендую начать с режима “Обучение”, — обратился дедушка к Дине. — Это важно, потому что в учебном мультике “Правилки” вы познакомитесь на этот раз с новой клавишей. Это клавиша удаления. На ней английская надпись *Delete*, что означает “удалить”. Именно она нужна для удаления лишнего символа.

В позиции “Лишний символ” дети посмотрели, как действует клавиша удаления, а затем почти без ошибок выполнили две серии упражнений из пяти заданий: одну серию — Дина, другую — Тим.

Вот какое правило усвоили ребята, поработав с “Правилкой” в режиме “Лишний символ”:

Для того чтобы удалить лишний символ, надо подвести к нему курсор и нажать клавишу удаления.

Демонстрационный мультфильм в позиции “Пропущенный символ” показал ребятам еще одну новую клавишу — клавишу вставки, на которой написано по-английски *Insert* — “вставить”.

Для того чтобы вставить пропущенный символ, надо поставить курсор на место вставки и нажать клавишу вставки. Тогда расположенная правее курсора часть строки сдвигается вправо, освобождая место для символа. На это место и надо поставить пропущенный символ.

Чтобы вставить все пропущенные символы в тексте

Струха у разбитого крыла ■

придется нажать на клавишу вставки три раза.

Перед тем как приступить к работе в режимах “Смесь”, “Суперсмесь” и “Контрольная”, дедушка Фёрстов предупредил:

— В каждом из предшествующих режимов мы начинали с “Обучения”. Обратите внимание, что, войдя в следующий режим — “Смесь”, вы не увидите привычного подменю с позициями “Обучение” и “Упражнение”. Как вы думаете, почему?

И, поскольку ребята призадумались, сам и подсказал:

— Да по той простой причине, что вы познакомились уже со всеми приемами исправления отсроченных ошибок. Во всех остальных режимах “Правилка” только предлагает вам упражнения. Самые разные. В режиме “Смесь” заранее невозможно предугадать, к какому из типов ошибок будет относиться очередное задание. А в “Суперсмеси” — и того хитрее: в одном задании, то есть в одном слове или одной фразе, окажутся несколько ошибок различных типов.

Но эти “страшные” предупреждения только раззадорили ребят: они с большим интересом набросились на задания новых режимов программы “Правилка”.

Вопросы

1. Сколько раз надо нажать на клавишу вставки, если в окошке “Правилки” появятся слова:

Сережий козлик ■

2. Сколько ошибок допущено в строчке известного стихотворения Н.А. Некрасова:

Однажды в студеную зимнюю пору ■

Сколько здесь неверных и лишних символов?

3. Дедушка Фёрстов назвал ошибки типа “неверный символ” самыми простыми. Согласны ли вы с ним? Почему?

Вечер четвертый

Побеседуем с компьютером

“Привет”

— Снова будем упражняться с “Правилкой”? — спросила Дина дедушку, как только он уселся рядом с внуками.

— Ну, если эта программа вам так понравилась, то, обещаю, мы к ней еще вернемся. Только позднее. А пока попробуем применить знания, которые вы получили от “Правилки”, в работе с другими программами.

— Какими?

— Сегодня мы поговорим с компьютером.

— Поговорим? — удивился Тим. — Как это?

— Очень просто: компьютер вам будет задавать вопросы, выписывая их на экране, а вы — отвечать на них, набирая слова ответа на клавиатуре.

— А-а! — протянул Тим понимающе.

А дедушка пояснил:

— Да. Именно так вы будете вести беседу с компьютером. Разговор двух собеседников называют диалогом. Поэтому и программы, в которых происходит обмен сообщениями — символами, словами или фразами — между человеком и компьютером, называют *диалоговыми*.

— Ну, так можно любую программу назвать диалоговой, — попытался возразить Тим. — Сначала программист дает компьютеру команду решить задачу; компьютер подумает, порешает и выдаст на экран результат, словно ответ на вопрос. Чем не диалоговая программа!

— По большому счету ты прав, Тим, в своих сомнениях. Но ведь никто не назовет диалогом лекцию, которую профессор читает студентам, хотя в начале лекции студенты и произносят “Здравствуйте!”, а в конце — “До свидания!”. Ведь в течение всей лекции говорит только один человек — профессор.

— Вот и разговор между командиром и солдатом тоже не похож на диалог, — заметил Тим. — Командир подробно объясняет свой приказ, а солдат в ответ говорит только одно слово: “Есть!” — и молча идет выполнять.

— Это как наш “Перевозчик”, — добавила Дина. — Мы ему говорим: “На левый берег...”, “На правый берег...”, “С козой...”, “С капустой...” — а он, ничего не говоря, все аккуратно исполняет.

— Ну, молодцы. Судя по вашим словам, вы легко сможете распознать диалоговую программу, если увидите, как она выполняется на компьютере. В диалоговой программе человек-пользователь и компьютер обмениваются не только словами приветствия и прощания, но еще и другой информацией, которая используется при выполнении программы.

— Так как же называется наша диалоговая программа? Где она? — нетерпеливо спросил Тим.

— Вот, смотри, — сказал дедушка Фёрстов, подводя мышкой стрелку курсора к пиктограмме, на которой изображен молодой человек, любезно снимающий шляпу,

— “Привет”, — прочитала Дина слово под пиктограммой.

— Да, программа так и называется — “Привет”, — сказал дедушка Фёрстов. — Ну-ка, Диночка, войди в эту программу.

— А я? — обиделся Тим.

— Как всегда, будем действовать по очереди. Следующий — ты, — успокоил его дедушка.

Как только Дина щелкнула левой кнопкой мыши, на экране возник молодой человек (“Это тот же, что изображен на пиктограмме”, — отметили ребята) и поздоровался:

— Привет!!! Как тебя зовут?

Дина обратилась к дедушке шепотом (наверно, она подумала, что молодой человек с экрана может ее подслушать):

— Мне надо набрать на клавишах: “Меня зовут Дина”?

— Нет, — ответил дедушка, — отвечай совсем кратко: пиши свое имя.

Девочка набрала свое имя: “Дина” — и задумалась.

Дедушка ее подбодрил:

— Ты правильно набрала ответ в окошке строкового редактора. Но для того, чтобы из этого окошка ответ попал в компьютер, то есть к твоему собеседнику, надо нажать клавишу выполнения.

Молодой человек с экрана, видимо, остался доволен полученным ответом, потому что он тут же задал следующий вопрос:

— Хорошо, Дина. Ты любишь играть в куклы?

Дина подняла вопросительный взгляд на дедушку, словно спрашивая, как записать ответ.

— Ну, понятно, что ты ответишь на такой вопрос. Конечно же, ты любишь играть в куклы. А записать ответ лучше всего коротким словом “да”. Можно с заглавной буквы — “Да”. Можно набрать все слово заглавными буквами — “ДА”. Ну а уж если ты хочешь, чтоб у “Привета” не возникало сомнений, ты можешь написать слово “конечно”.

Тим вмешался:

— Если бы я сейчас сидел у компьютера, я бы, конечно, написал “нет”.

Дина пропустила это замечание мимо ушей, она решила написать просто и скромно.

— “Да”, — набрала она на клавишах.

И тут же получила от “Привета” новый вопрос:

— Я думаю, Дина, что ты девочка, да?

Дина даже удивилась: “Какие тут могут быть сомнения?”. И хотя дедушка объяснил ей, что “Привет” не видит ее косички через компьютерный экран, а может только читать ее сообщения, набранные на клавиатуре, Дина все-таки изобразила возмущение и написала:

— Еще бы!

Компьютер ответил мгновенно на ярко-красном фоне:

Я не знаю этого слова!

Ни Тим, ни бабушка не удивились, что компьютеру не понравился ответ Дины. Да и она сама понимала, что в программе компьютера не предусмотрена реакция на такой необычный ответ пользователя. Поэтому ей пришлось ответить на тот же вопрос еще раз, на этот раз спокойнее:

— Конечно.

На экране — новый вопрос:

— Хорошо, девочка Дина. Назови мальчика из сказки.

Тут снова пришлось задуматься. В голове промелькнули сказочные персонажи: Буратино и Пьеро, Мальчик-с-пальчик и Мальчиш-Кибальчиш, Мальчиш-Плохиш и Незнайка.

— А какого выбрать, — решила она посоветоваться с братом, — хорошего или плохого?

— Какого хочешь. Я бы выбрал Мальчиша-Кибальчиша. Мне он очень нравится. Геройский парень.

Дина последовала совету брата, а когда в ответ на с трудом набранное на клавиатуре имя мальчика с двумя заглавными буквами последовал новый вопрос программы о том, какой же он, этот Мальчиш-Кибальчиш, она недолго размышляя написала:

— Геройский.

Диалог на этом не кончился. Программа продолжала задавать вопросы о сказочном персонаже, который Дине не нравится, — как его зовут, какой он.

И вот наконец в самом конце беседы программа “Привет” исчерпала свои вопросы и, словно в награду за откровенный разговор, предложила Дине послушать сказку:

Однажды
геройский Мальчиш-Кибальчиш
рванул вместе с Кучей Мусора
командовать парадом.
Вот добрались они до места.
Смотрят, а там совсем
зловредная Баба Яга,
— Три тысячи чертей!
Так подумали
Мальчиш-Кибальчиш и Куча Мусора,
Они стали хохотать
очень сильно. И пропал недруг!
Занялись друзья делом,
а сказке конец.
Привет!!!

Ребята дружно посмеялись над этой компьютерной сказкой, а поскольку в нижней строке экрана зажглось приглашение к новой сказке, то Тим не мешкая воскликнул:

— Теперь я!

Новый диалог, новые вопросы и ответы, новая такая же смешная сказка.

Дина с удовольствием читала сказку брата. А в конце вопрос бабушке:

— Бабушка, а как же работает эта программа, если каждый раз она сочиняет разные сказки?

Бабушка уже приготовился отвечать, но тут раздался звонок в дверь — это пришли Артем и Наташа — одноклассники Дины и Тима.

— Мы возвращались домой из школы, с занятий театрального кружка. Но на улице такой дождь. Просто настоящий ливень. Вот мы и решили зайти к вам, немножко согреться.

— Вы правильно сделали, что зашли. Дина, Тим! Проводите гостей на кухню, напоите чаем, а потом все вместе возвращайтесь в мой кабинет.

Вопросы и упражнения

1. Чем отличается диалог людей от диалога компьютера с человеком?

2. Составьте два диалога с программой “Привет”, повторив во втором из них все те же слова, которые были набраны в первом. Сравните две получившиеся сказки: одинаковы они или нет? Какое можно предложить объяснение этому?

3. Чем оказалась полезна “Правилка” детям, которые работают с программой “Привет”?

Как работает программа “Привет”

Георгий Аркадьевич не пошел на кухню. Дети там хозяйничали самостоятельно. И пока Наташа с Артемом увлеченно делились с приятелями впечатлениями о новом спектакле школьного театрального кружка, он достал из углового шкафа несколько коробок, приклеил к ним какие-то ярлыки и уселся писать карточки, которые затем стал раскладывать по коробкам. Дети вернулись очень скоро: хотя чай был вкусный, все же хотелось поскорее вернуться к компьютеру. Однако, на удивление внуков и их гостей, бабушка посадил детей не около экрана, а вокруг стоящего посередине комнаты стола и поставил на стол четыре коробки.

— В этих коробках, — начал объяснять бабушка Фёдоров, — лежат карточки, на которых написаны разные члены предложения. — И он показал детям коробки, на которых было написано крупными буквами: “Подлежащие”, “Обстоятельства”, “Сказуемые”, “Дополнения”.

— Вытаскивая не глядя (то есть в случайном порядке) из каждой коробки по одной карточке, вы построите таким образом фразу. Попробуйте.

Дина достала карточку из коробки “Подлежащие”, Наташа — из коробки “Обстоятельства”, Артем залез в “Сказуемые”, а Тим — в “Дополнения”.

Когда они выложили свои карточки на стол, вот что у них получилось:

Артем устало кусает Дину

Нельзя было не улыбнуться, читая эту фразу.

— Забавная игра, — оценили Артем и Наташа.

— А мне понятно, почему дедушка показал нам ее: ведь точно так же, как эта игра, работает программа “Привет”.

И, обращаясь к деду, спросила:

— Это вместо объяснения, как работает “Привет”?

Дедушка улыбнулся и подтвердил ее догадку:

— В коробке “Подлежащие” лежат всего две карточки — имена мальчиков (в именительном падеже), в коробке “Дополнения” тоже две карточки — имена девочек (в винительном падеже). Тут выбор небольшой. А вот в коробке “Обстоятельства” карточек много: тут слова “больно”, “быстро”, “весело”, “грустно”, “устало”, “старательно”, “сильно”, “отлично”, “плохо”, “осторожно”. И сказуемых тоже много: “догоняет”, “кусает”, “провожает”, “щекочет”, “обижает”, “защищает”, “развлекает”. Впрочем, количество карточек можно увеличить, тогда игра станет еще забавнее.

— Все понятно, — сказал Артем. — Карточки из коробок мы вынимаем случайно, поэтому часто получаются забавные фразы.

И спросил в свою очередь:

— А что это тут Дина сказала про программу “Привет”?

Дедушка ответил:

— Ну, это пускай она вам с Наташей сама покажет. А Тим поможет.

Вопросы и упражнения

1. Объясните сходство игры с карточками дедушки Фёрстова и программы “Привет”. Почему игра может служить объяснением этой программы?

2. Могут ли в результате дедушкиной игры получиться правильные предложения или все они будут неправдоподобно смешными?

3. Постройте сами такую игру, придумав свои наборы сказуемых и обстоятельств, и пригласите для игры в нее несколько мальчиков (имена которых будут подлежащими) и девочек (с именами-дополнениями).

Сказка по выбору

Наигравшись вдоволь с “Приветом”, Наташа и Артем уже начали было прощаться, но тут Дина обратилась к дедушке:

— Деда, а может, у тебя для нас что-нибудь еще припасено на сегодня? Тогда и ребята с нами вместе тоже посмотрели бы твои программы.

— Что ж, извольте. Я и в самом деле хотел вам показать еще одну программу, которая так же, как и “Привет”, использует ваши умения работать со строчковым редактором. Кстати, в этой программе вы тоже увидите сказочные сюжеты. Она так и называется — “Конструктор Сказок”.

— Мы будем сами строить сказки?

— Программа “Конструктор Сказок” действительно написана специально для того, чтобы дети могли создавать сказки и другие истории, которые можно было бы применять и для развлечений, и для обучения.

Но это не такая уж простая программа. Я хотел подробно рассказать вам об этой программе, начнем изучать ее возможности попозже, примерно через год. А сегодня просто научимся читать тексты, построенные с помощью “Конструктора”.

— Компьютерная сказка! Как интересно! — воскликнула Наташа.

Дедушка Фёрстов подвел курсор к пиктограмме сказки, на которой был изображен старичок-сказочник.

— Этот старичок тоже любит рассказывать сказки. Только, в отличие от своего молодого приятеля “Привета”, он не ждет, когда вы ему ответите на кучу вопросов, а сразу знакомит с персонажами своих историй. Вы спросите: каких? Вы сами сможете их выбрать. Сказочник рассказывает истории по вашему выбору.

Компьютерный сказочник и вправду напоминает умудренного рассказчика, который время от времени останавливается, чтобы передохнуть, оглядеть внимательно своих слушателей, поинтересоваться их впечатлениями. Слушая этого симпатичного старичка, вы словно листаете страницы сказки с картинками. Самое интересное в работе с программой “Конструктор Сказок” состоит в том, что экранные страницы можно листать не обязательно строго по порядку, как всегда читают обыкновенную книгу, страницу за страницей. Да у компьютерной сказки вы и не увидите эти привычные номера. Можно продолжать чтение каждый раз так, как этого хочет читатель. Внимательно следуя рекомендациям компьютерного сказочника, можно самостоятельно создать сюжет повествования.

Дедушка уступил Наташе свое кресло. Она решительно щелкнула по пиктограмме со сказочником.

В середине экрана появилась картинка — сказочный деревянный домик на лесной полянке. Над ней — слова негоропливого рассказа старого сказочника:

В одном теремочке в некотором лесочке жили-были вместе...

А под картинкой — продолжение.

Если вы хотите, чтобы там жили кот Котофей, пес Барбос и петух Кукарек, наберите слово ТЕРЕМОК. Если же хотите узнать другую сказку, наберите: ЗВЕРИ.

Внимательно прочитав страницу-экран и посмотрев картинку, Наташа сказала:

— Симпатичные животные: и кот, и пес, и петух. Мне они нравятся.

— Ну, тогда пиши ТЕРЕМОК, — заявил Тим. — Тут ведь надо выбрать, куда идти дальше.

Наташа так и сделала, быстро набрав на клавишах слово ТЕРЕМОК; там, где стоял курсор, — в самой нижней строке экрана она нажала клавишу выполнения. Но программа не только не перевернула страничку, но еще вдобавок зажгла неприятное сообщение:

Неверно! Нажмите пробел

— Это почему же? — огорчилась Наташа.

— А ты обратила внимание, как написано слово ТЕРЕМОК в нижней части страницы, под картинкой?

— А, понятно, — сказала Наташа. — Тут написано заглавными буквами. Значит, так я и должна была набирать. А я набрала это слово малыми буквами. Как же поправить?

Дедушка Фёрстов помог:

— Чтобы набираемая буква выглядела на экране заглавной, надо нажимать ее вместе с клавишей **[Shift]**.

Он показал ее на клавиатуре.

— Значит, если мне надо нажимать подряд несколько заглавных букв, как в нашем примере со словом ТЕРЕМОК, я должна одним пальцем всегда держать нажатой эту клавишу **[Shift]**? — спросила Наташа.

— Вовсе нет. Чтобы не нажимать на **[Shift]** все время, достаточно один раз, нажать клавишу, которая находится над ней. Это **[Caps]**. Перевести с английского это название можно словами “запереть за замок (зафиксировать) регистр заглавных букв”. А когда потребуется снова вернуться к малым буквам, надо будет повторить нажатие этой клавиши еще раз.

Исправив свою ошибку и установив заглавные буквы, Наташа перешла к следующей выбранной ей странице. Сказка льется складно:

Кот Котофей, пес Барбос и петух Кукарек жили дружно и ладно. Но вот однажды собрался кот на охоту за мышами в...

А под картинкой снова следует продолжение, управляющее сказкой:

Куда же он пошел? Это зависит от вас. Если хотите, чтобы он пошел в лес, наберите слово ЛЕС. Если нужно, чтобы Котофей пошел в огород, наберите ОГОРОД.

Посоветовавшись с друзьями, Наташа выбрала дальнейшее продолжение истории. Так, от одной страницы к другой, прочитала она всю сказку.

Тим, который всегда ищет свои дороги, и на этот раз произнес:

— А я бы выбрал другой путь по этой сказке.

Вопросы

1. Обратили ли вы внимание на то, что два сказителя — “Привет” и “Сказочник” — по-разному рассказывают свои сказки? В чем, по вашему мнению, состоит различие их сказок?

2. Что общего можно увидеть в двух программах — “Привет” и “Конструктор Сказок”?

3. Пока не нажата клавиша выполнения, в набираемый текст можно вносить любые исправления. Почему?

4. Что произойдет, если после неправильно набранного слова будет нажата клавиша выполнения?

ИНФОРМАТИКА А



Газета “Информатика” и Роботландский сетевой университет
объявляют о первом этапе совместной акции

“Подписчикам везде у нас дорога... и скидка!”

Данный купон дает право на скидку в размере 10% при приобретении:

- интерактивного учебника-лаборатории “HTML-конструирование” (см. № 21, 22/2000);
- программного пакета “Зимние вечера” (см. № 1/2001).

В дальнейшем наши подписчики получают скидки и на другие продукты Роботландского сетевого университета.

Для получения скидки необходимо выслать заявку на приобретение того или иного продукта по адресу: 152025, г. Переславль-Залесский, ул. Октябрьская, д. 43, кв. 112, Дуванову Александру Александровичу.

В письмо необходимо вложить оригинал данного купона или ксерокопию купона вместе с ксерокопией подписной квитанции на “Информатику”. Для быстрого получения программ рекомендуется дополнительно отправить электронное письмо с заявкой по адресу: kurs@robotland.botik.ru. В электронном письме требуется указать дату отправки бумажного письма с купоном или ксерокопиями.



Данное предложение действительно до 30 апреля 2001 года.

Книга для учителя

Обращение к читателям

I. К школьному учителю

Хотя эта книга, предназначенная в первую очередь учителям, которые ведут своих школьников по курсу “Информатика для начинающих”, не является описанием Роботландии, все же опирается она практически целиком на роботландские разработки, и потому экскурс в историю этой выдуманной, но востребованной страны представляется обоснованным.

Без малого пятнадцать лет тому назад в маленьком городке Центральной России Переславле-Залесском удалось собрать интересную группу разработчиков-программистов. Это были молодые люди (пожалуй, только автора этих строк уже в то время было несколько некорректным называть “молодой”), приехавшие в Переславль из далекого Зауралья — Сибири, Дальнего Востока. Впрочем, их объединение в творческий коллектив можно было объяснить не географическими и не возрастными параметрами, а прежде всего единым тяготением к новым в те времена проблемам школьной информатики. Все они (может быть, в разной степени) были “заражены” ершовскими идеями раннего информатического образования и, получив широкое поле и определенную свободу деятельности, со свойственным молодежи энтузиазмом окунулись в методические, программистские и технические проблемы использования компьютеров в переславльских школах. В те времена считалось естественным начинать исследование возможностей применений компьютеров в той или иной проблемной области с выбора (или, часто, создания) языка программирования, адекватного этой предметной области. Неудивительно, что и молодежь переславльской группы не могла пройти мимо обсуждения языковых проблем. Однако очень скоро стало ясно, что для обучения младших школьников не годятся ни “простой” Бейсик, ни гибкий Алгол, ни “древний” Фортран, ни пространственный Кобол, ни “студенческий” Паскаль, ни даже всеми высоко почитавшийся молодежный Лого. Языковая система программирования, даже такая богатая, как Лого, будучи объемной и единственной, приходит в противоречие с дидактическим принципом многообразия форм обучения. Многообразие же операционных средств, порождаемых системой исполнителей, практически неисчерпаемо.

Здесь не обсуждается актуальность (необходимость) раннего информатического образования: можно полагать, что учитель, взявший на себя заботу обучения младшей начальной школы элементам информатики, убежден в этом. Впрочем, доказательства этой актуальности неоднократно приводились отечественными и зарубежными исследователями, в том числе и “роботландцами”. Исходная посылка такого доказательства состоит в

том, что основной целью изучения информатики в школе является не овладение ремеслом написания программ, а формирование стиля мышления, адекватного требованиям современного информационного общества. В условиях раннего информатического образования наглядность и многообразие системы программных исполнителей представляют собой дополнительные весомые аргументы в сравнении с высокоуровневыми языковыми системами программирования.

Эпоха зарождения Роботландии была временем, когда на “Агатах” уже работали программные исполнители Звенигородского, а на “Ямахах” замышлялись исполнители Кушниренко. Неудивительно, что переславльские программисты сделали учебные программные исполнители пристальным объектом изучения, творчества и педагогической практики. На успехе задуманной идеи не могли не отразиться и технологические (сейчас сказали бы, “объектно-ориентированные”) аспекты проектирования исполнителей. С одной стороны, каждый исполнитель замышлялся, опробовался и внедрялся ради некоторого, подчас единственного, умения или навыка, который и формировался этим исполнителем. С другой стороны, не будучи столь громоздким, как языковая система программирования, каждый исполнитель оперативно проектировался и очень быстро получал законченную программную реализацию. (Вспоминается случай, когда обсужденная ранним утром идея очередного программного исполнителя к обеду превратилась в законченную программу, а вечером обсуждалась с детьми на кружке.)

Примеров “узкой специализации” роботландских исполнителей можно привести много. Так, “Перевозчик” предназначен для первого знакомства с понятием алгоритма. “Правилка” очень полезна для формирования навыков корректировки ошибок, неизбежных при клавиатурном наборе. Исполнитель “Квадратик” нужен для освоения общей схемы знакомства с исполнителями... Разумеется, многие исполнители не ограничиваются единственной (или строго очерченной) педагогической задачей. Вот пример: исполнитель “Конюх”. С одной стороны, он расширяет коллекцию алгоритмических этюдов, с другой стороны, работает на расширение клавиатурных навыков, демонстрируя необходимость манипулирования обоими регистрами — кириллицей и латинью. Наконец, “Конюх” пропедевтически вводит младшим школьникам важное понятие координат, которое в “классической” школьной методике математики появляется существенно позднее (заметьте: все перечисленные здесь темы, использующие эти программные исполнители Роботландии, существенно опираются на возрастные особенности контингента обучаемых — школьников начальной школы).

Идея использования программных исполнителей в обучении младших школьников основам информатики проста и естественна. Удачей же коллектива следовало

считать то, что каждый разработчик-роботландец, сам будучи педагогом — кто по диплому, кто по призванию, кто в силу хорошо понимаемых родительских обязанностей, — создавал элементы будущей системы не для забавы или профессионального самоутверждения, а для практического внедрения в педагогическую предметную область, то есть для детей. Это обеспечило сооружение не аморфного набора исполнителей, а построение их продуманной системы. Коллектив много дискутировал, обсуждая место каждого исполнителя в разрабатываемой системе, их последовательность, их группирования в темы и методические линии. Конструктивное участие в таких дискуссиях, наряду с программистами и педагогами, принимали психологи и эргономисты.

В содержании курса раннего информатического образования, базирующегося на системе исполнителей, выделяются следующие основные направления:

1. Информационное (теоретическое) направление, на котором формируется современное мировоззрение молодого человека.

Понятие информации ложится в основу единой информационной картины мира. Рассматриваются различные способы представления информации, виды информационных процессов — хранение, передача, обмен.

2. Компьютерное (практическое, технологическое) направление, на котором формируются важнейшие навыки общения с компьютером, а также представление о компьютере как универсальной информационной (а не только вычислительной) машине.

Дети знакомятся с разнообразными применениями компьютеров в повседневной жизни. Среди информационных технологий, которые осваивают учащиеся начальной школы, выделяются компьютерные телекоммуникации, и прежде всего электронная почта.

3. Алгоритмическое направление, на котором ученик изучает ряд важнейших понятий и методов информатики, учится описывать, конструировать и анализировать алгоритмы.

Алгоритмы предстают перед детьми как модели различных видов деятельности.

4. Исследовательское направление, которое одной из своих важнейших задач считает сформировать у ребенка творческое отношение не только к изучаемому предмету, но и ко всей своей (пока — учебной) деятельности.

Программный инструментарий, который дети осваивают в курсе раннего обучения информатике, становится базой для проведения простейших самостоятельных исследований, для воспитания навыков творческого, литературного, художественного, музыкального творчества.

Роботландия как курс раннего обучения информатике представляет собой не хронологическую последовательность этих направлений, а связку тем из разных направлений, переплетающихся практически на каждом занятии.

Для курса “Зимние вечера. Информатика для начинающих” отобрана не вся Роботландия, а только ее часть. Базовая, начальная, но все-таки часть. Хотя в курсе постоянно, начиная с первых занятий, употребляется слово “программа” (и даже изредка слово “программирование”), все же о программировании в классическом понимании этого слова здесь речи нет. Ключевые слова информационного и алгоритмического направлений — это “алгоритм”, “исполнитель”, “команда”, “система команда”..., а технологического и исследовательского — “редактор”, “файл”, “каталог”, “электронная почта”... Вместе с тем книги (учебник и книга для учителя) имеют право на самостоятельное существование, поскольку рассказывают о законченной системе понятий, целенаправленно работающих на формирование стиля мышления ребенка.

Читатели (и дети, и педагоги), заинтересовавшиеся идеями этих книг, без труда смогут (и будут готовы) продолжать освоение информатики по многочисленным уже книгам, и в первую очередь по методическим материалам Роботландского сетевого университета в его последующих курсах.

II. К руководителю команды Роботландского сетевого университета

Наше время с его информационно-технологическими возможностями сделало актуальной новую форму образования — дистанционное обучение. Наряду с традиционными формами образования — очной, вечерней, заочной — и даже настойчиво вытесняя две последних, дистанционное обучение позволяет совместить такие контрастирующие подходы, как массовость обучения, с одной стороны, так и индивидуальная гибкость учебного процесса, с другой. Правда, дистанционное обучение в школе пока еще делает свои первые шаги; основные результаты этой новой информационной учебной технологии достигнуты главным образом в высшей школе.

В общеобразовательной средней школе дистанционное обучение активно захватывает сферу дополнительного образования (кружки, внеклассные соревнования, летние лагеря и т.п.). Переход от дополнительного образования к базовому для дистанционного обучения — это пока только лозунг, перспектива (хотя, безусловно, совсем близкая).

Дистанционные влияния не обошли стороной и Роботландию. Уже более трех лет работает такая зарекомендовавшая себя форма дистанционного обучения, как Роботландский сетевой университет. Более двух тысяч школьников со всех уголков России и наши ближайшие соседи из стран СНГ стали слушателями этого дистанционного учебного центра. Но — парадокс! — хотя Роботландия всегда ориентировалась на потребности начальной школы, на курсах Роботландского университета до последнего учебного года занимался главным образом “взрослый” контингент — ребята из 5-х классов, а зачастую и старшеклассники.

Впрочем, в составе университета всегда был (и по-прежнему остается) курс, подробно рассказывающий о методике раннего обучения информатике. Он ориентирован на индивидуальных (а не коллективных!) взрослых слушателей — учителей, которые, повышая свою квалификацию, готовят себя к предстоящей преподавательской деятельности, к обучению малышей основам информатики. Сколь бы ни был успешен такой курс, он лишен главного качества коллективных курсов — совместного обучения детей и педагогов в сети.

Для того чтобы ликвидировать эту методическую нишу, для того чтобы показать, что дистанционное обучение учащихся начальной школы — это не декларация, а реальность, курс для самых юных слушателей включен отныне в систему курсов Роботландского сетевого университета. В нем реализуются все методические решения, которые были найдены в других курсах, органически ориентированных на коллективного слушателя — команду детей, возглавляемую, как правило, школьным учителем:

- соревновательная форма учебного процесса (турниры, конкурсы, вернисажи по завершении отдельных тем);
- практическая направленность педагогических результатов (сборники программ и пакетов, сделанных детьми и предназначенных для использования на компьютеризированных школьных уроках);
- перекрестные проверки (действенный методический прием, превосходящий по своей результативности широко пропагандируемую рефлексию).

Курс, о котором идет речь, назван “Зимние вечера. Информатика для начинающих”.

Такому курсу необходим учебник для учащихся (для младших школьников), излагающий содержание обучения, и книга для учителя, рассказывающая о методике проведения занятий.

Такое предназначение методических материалов определило структурирование учебника. Темы рассредоточены по отдельным занятиям-вечерам. С учетом строгой дисциплины дистанционного обучения, каждому такому “вечеру” отводится в календарном плане неделя (календарный план дистанционного курса “Зимние вечера. Информатика для начинающих” приведен в приложении 1). Хотя детализация учебного времени внутри темы осуществляется руководителем команды, можно утверждать на основе учительского опыта, тщательно анализируемого в дистанционном центре в конце каждого учебного года, что такая недельная порция материалов “поглощается” либо одним плотным двухчасовым занятием, либо двумя часовыми встречами в течение рекомендуемой недели.

Тема завершается конкурсом или турниром, и на проведение такого коллективного соревнования приходится планировать отдельное занятие — “вечер”.

Перекрестная проверка — это тоже чаще всего отдельный “вечер”. Ни один учитель — слушатель Роботландского университета ни на одном из курсов не назвал такой “вечер” потерянным.

И турниры, и перекрестные проверки — эти характерные для дистанционного обучения школьников формы контроля могут с меньшей интенсивностью нагружать традиционный “очный” курс в школе или дома, но для обеспечения оперативного контроля в сети такие занятия становятся не дополнительной, а органической частью учебного процесса.

Хотя, следуя многочисленным пожеланиям руководителей команд учащихся (то есть школьных учителей), кураторы и преподаватели и стремятся приблизить структуру курсов, их содержание, порционирование материала, методику преподавания к требованиям урока и школьной программы, все же в полном объеме такая задача сегодня неразрешима, особенно в начальной школе. Прежде всего это связано с тем, что пока не существует государственного стандарта раннего информатического образования школьников. Как следствие, в российских школах существует почти столько же реализаций курса основ информатики в начальной школе, сколько учителей-энтузиастов инициативно и творчески взялись за эту непростую экспериментальную работу. В таких условиях трудно говорить не только о едином расписании, но даже о едином содержании.

Дистанционный курс основ информатики для начинающих может внести определенную лепту в планируемый в перспективе стандарт, однако сегодня приходится считаться с тем, что в силу нынешних условий он вынужденно относится к формам дополнительного образования и, следовательно, может фиксироваться у завуча школы как факультатив или кружок.

Трудности в привязке занятий дистанционного обучения к текущему школьному учебному процессу добавляет огромное разнообразие технического обеспечения школьных кабинетов информатики и особенно учебных программ, программных пакетов и прикладных программных систем, используемых в школе. Эти трудности в курсе “Зимние вечера. Информатика для начинающих” решаются, во-первых, строгой ориентацией на наиболее типовые школьные компьютеры сегодняшнего дня — ПК на платформе IBM, а во-вторых, на специально для этого курса подготовленное программное обеспечение — пакет “Зимние вечера”, который почти полностью сформирован из программных продуктов Роботландии. Только во втором семестре часть занятий проводится на “стандартном” прикладном программном обеспечении, которым чаще всего располагает даже не самый “богатый” программными средствами школьный кабинет информатики — типовая почтовая система The Bat!, простой графический редактор Paint.

Курс выдвигает достаточно строгие возрастные критерии к контингенту обучаемых: это должны быть в основном школьники третьего и четвертого классов. Не следует понимать это ограничение как отступление от роботландцев от собственной концепции раннего обучения информатике, по которой утверждается необходимость начинать знакомство детей с фундаментальными понятиями информатики на пороге школы,

в первом классе. Школьник первого класса, безусловно, способен усвоить начала новой научной дисциплины, однако темп усвоения учебного материала перво-классниками настолько невысок, что в тесные рамки двух десятков занятий, предусматриваемых для дистанционного курса, не удалось бы вместить содержание, затрагивающее телекоммуникационные возможности компьютера, без чего дистанционная форма обучения попросту теряет смысл.

III. К родителям и гувернерам

Домашний компьютер становится в наши дни заурядным явлением: количество персональных компьютеров, установленных в российских семьях, уже сейчас измеряется миллионами. Как правило, домашний компьютер — это мощная машина, превосходящая по своим возможностям средний компьютер школьного кабинета информатики; большинство домашних компьютеров имеет выход в глобальную информационную сеть.

В подавляющем большинстве “компьютеризованных” семей есть дети. Миллионная армия детей, имеющих расширенные возможности общения с компьютерами, это одновременно и огромный ресурс системы образования, и ее головная боль. Не учитывать эти ресурсы, заботы и проблемы сегодня становится невозможным.

Возможные направления педагогического использования домашних компьютеров многообразны. Их классификация и оценка их социальной значимости, безусловно, заслуживают пристального внимания и отдельного рассмотрения. Здесь эти вопросы не обсуждаются. Важен принципиальный тезис: в любой области педагогического применения домашних компьютеров — от восполнения пропущенных навыков до экстерната, от формирования углубленных профессиональных навыков до освоения начальных понятий информатики, от обучения детей, лишенных возможности посещать школу, до индивидуального творчества — необходимо тесное и постоянное сопряжение деятельности и интересов школьного учителя информатики и куратора домашних компьютерных проблем юного ученика. Куратор должен понимать и разделять концепцию школьной программы, направленной на формирование стиля мышления ребенка, и иметь желание, умение и время помочь своему подопечному; учитель же должен знать технические характеристики и уровень программного обеспечения домашнего компьютера школьника и главное уровень методической подготовленности куратора.

Читатель без труда обнаружит третью ориентацию этой книги, которая предназначена для такого куратора. Лучше всего, если роль куратора сможет выполнять кто-либо из родителей, владеющих навыками общения с персональным компьютером и выхода в Интернет. Один из главных персонажей книги для школьника — дедушка Фёрстов, который свои долгие зимние вечера заинтересованно посвящает своим люби-

мым внукам. Когда-то дедушка Фёрстов сам учительствовал, преподавая информатику в школе, а сейчас пенсионер Фёрстов, отложив ненадолго свои заметки и статьи на педагогические и компьютерные темы, передает свои знания Дине и Тиму. Выбрав такого методически подготовленного персонажа, удалось уйти от, вообще говоря, непростой проблемы сопряжения интересов гувернера и школьного учителя.

Взрослому, который ведет ребенка по дистанционному курсу у своего домашнего компьютера, нельзя не учитывать специфику домашних условий. С одной стороны, дома ребенок располагает несравненно большими ресурсами личного времени, внимания персонального куратора и технических ресурсов. (Уместно заметить, что в возрастной ориентировке курса “Зимние вечера” отчетливо говорится о третьеклассниках, занимающихся в типовой общеобразовательной школе, тогда как ученики дедушки Фёрстова — второклассники. Это не случайно.) С другой стороны, “домашний” ребенок лишен столь необходимого ему активного общения со сверстниками. Это важно, поскольку методика курса существенным образом опирается на коллективные и соревновательные формы учебного процесса.

В книге предложен своеобразный компромисс: дети занимаются с дедушкой дома, но их двое — брат и сестра. Ясно, что такая ситуация вряд ли может считаться типовой (хотя и не является стопроцентной выдумкой автора: в Роботландии разработчики учебных программных систем брали за правило сначала испытать новую разработку в семье, на собственных детях, затем на кружке или в летнем компьютерном лагере и лишь после этого в массовой школе). Когда куратор вынужден сопровождать по курсу не братьев и сестер, а единственного ребенка, педагогическая задача куратора — родителя, дедушки, гувернера — в действительности даже сложнее, чем у дедушки Фёрстова.

Впрочем, расставляя приоритеты в ориентирах книги, автор, без сомнения, отдает предпочтение дистанционному обучению перед семейным (разумеется, термины “дистанционное” и “семейное” обучение отнюдь не альтернативны). Например, дедушка Фёрстов вполне мог бы при организации перекрестных проверок удовлетвориться тем, что его внуки проверят турнирные задачи друг у друга. Тем не менее он умышленно не отказывается от возможности вывести свою молодежь в телекоммуникационное общение с их далекими сверстниками.

Легких путей и четких рекомендаций, работающих на все случаи жизни, в педагогике не бывает. И любой читатель этой книги (либо школьный учитель, взявшийся преподавать элементы информатики в начальной школе, либо дедушка, отдающий свое время и время своего компьютера внуку) должен искать в книге не безусловные предписания, а информацию для размышления о собственном педагогическом творчестве.

Как работать с книгой

Книга рассказывает о годовом курсе “Зимние вечера. Информатика для начинающих”. Занятия (желательно двухчасовые) планируются один раз в неделю. С учетом каникул — осенних, зимних и весенних — и несколько сокращенного учебного года (с середины октября по начало мая), по сравнению с традиционным школьным, получается 25 занятий. Эти 25 занятий несколько условно распределены на два семестра, а внутри семестров еще и по разделам/темам.

В каждом занятии учителю важно выделить содержание, методику (цели занятия, его план, способы контроля, комментарии и обсуждения), а также программное сопровождение урока. Содержание определяется книгой для учащегося, где каждому занятию отведен один параграф, достаточно подробно описывающий это занятие (исключение составляют конкурсные и контрольно-проверочные занятия). Методические рекомендации в виде кратких тезисов распределены по параграфам-занятиям в книге для учителя. В тех же параграфах сообщаются сведения, которые необходимы учителю для его работы с используемыми на занятии программными средствами.

Для удобства совмещения учебника и книги для учителя пункты в планах занятий поименованы названиями пунктов ученических параграфов.

Руководитель команды в принципе имеет три варианта предъявления детям учебной информации.

Первый — распечатка текстов занятий из учебника в достаточном количестве экземпляров и распределение их между детьми для последующего обсуждения. Такой вариант, приемлемый в старших классах, не годится для третьеклассников.

Второй вариант — совместное прочтение учебного материала с экрана компьютера или с его твердой копии.

Наконец, третий вариант — единственно возможный для детей младшего школьного возраста — это подробный разговор учителя с детьми на тему занятия из учебника “Зимние вечера”. При этом речь может (и должна!) идти не о прямом пересказе диалогов дедушки Фёрстова с его внуками, а о творчески подготовленной беседе со своими учащимися на заданную учебником тему с использованием примеров из учебника, а также контрольных вопросов и упражнений, которыми заканчивается каждый пункт параграфа.

При такой форме занятия текст учебника может рассматриваться как дополнительный, раздаточный материал, если у учителя есть возможности (техника, ресурсы, время) для распечатки страниц учебника. В этом случае выданные ученикам страницы можно эффективно использовать для закрепления материала темы и для домашних заданий. Тем ученикам, которые имеют доступ к домашним компьютерам, можно переписать файлы изученных на уроках заданий, с тем чтобы ребенок смог дома перечитать текст, просмотреть картинки, выполнить упражнения.

В этом последнем случае учитель должен быть уверен, что домашний куратор поможет ученику в технической обработке учебного материала — чтении с дискеты, разархивировании файлов, выводу их на экран, антивирусной проверке, установке необходимых параметров чтения с экрана (выбор и размещение файла, установка масштаба изображения и проч.). Даже если учебная информация из школьного кабинета информатики на домашний компьютер передается не на дискете, а по электронной почте, требования к подготовленности домашнего куратора по приему и предварительной обработке полученной информации остаются.

Готовясь к очередному занятию, учитель должен сам поработать с каждой запускаемой на уроке программой во всех ее режимах, отметить особенности их работы. Такой учительской самоподготовке может помочь пользовательская документация, которая для многих программ раскрывается щелчком по пиктограмме с названием “Описание”. В файл описания программы легко войти с рабочего стола, поскольку запускаемая программа и ее описание расположены, как правило, по соседству и имеют одинаковые пиктограммы. В ряде случаев пиктограммы описаний не выведены на рабочее поле, но тем не менее текстовые файлы описаний легко обнаруживаются в соответствующих каталогах. Имена этих файлов имеют расширение DOC (обратите внимание на то, что это текстовые файлы MS-DOS, а не файлы Word).

К пиктограммам описаний (файлам с документацией) можно не привлекать внимание учащихся. Впрочем, и скрывать эту информацию от них не надо. Дело в том, что в “Зимних вечерах” материал насыщен, дети достаточно сильно нагружены и без того, чтобы изучать (тем более самостоятельно) пользовательскую документацию к программам. А главное, эти тексты — не детские.

В ходе каждого занятия предусматривается работа детей на компьютере. Идеально, если каждому ребенку может быть предоставлено индивидуальное рабочее место. Впрочем, третьеклассники с удовольствием работают у экрана парами, если учитель продуманно подошел к комплектации пар, учитывая психологическую совместимость детей. Даже работая в парах, каждый ребенок должен вынести с занятия сформированные или закрепленные навыки управления программой.

Многие программы “Зимних вечеров” в соответствии с традициями Роботландии выполнены как открытые программы. Это значит, что они могут рассматриваться как программные оболочки, которые наполняются обновляемыми к каждому уроку данными. Тем не менее в большинстве случаев (кроме специально оговоренных) здесь возможности открытых программ-оболочек умышленно усечены. “Зимние вечера” — это курс введения в информатику, курс очень динамичный. Поэтому целесообразно не отвлекать учеников и учителя на концептуально не-

простые идеи программного конструирования, которые находят свое воплощение в методически следующих за “Зимними вечерами” курсах Роботландского сетевого университета.

Однако, если учитель располагает достаточным временем и заинтересован в учебных применениях открытых программ, он сможет использовать для этого в большинстве случаев документацию к соответствующим программам, а в остальных ситуациях ему будет рекомендовано обращаться к программным системам “Роботландия” или “Хиты Роботландии”, куда эти программы включены со всем необходимым инструментарием, позволяющим использовать их как открытые программы.

Спецификой дистанционных курсов являются контролируемые занятия, в которых взаимная (перекрестная) проверка становится неотъемлемой частью учебно-воспитательного процесса. Именно здесь проявля-

ются столь характерные для дистанционного обучения дисциплинарные требования к срокам, синхронизации и оформлению. Учитель должен приложить немало усилий не только для того, чтобы его ученики решили задачу, но и для того, чтобы они оформили свою работу в строгом соответствии с заданиями. В частности, во многих задачах контролируемых занятий-соревнований в не меньшей степени, чем собственно верное решение, оценивается обоснование найденного решения. Важно, чтобы дети решили и оформили все задачи самостоятельно. Конечно, учитель может и должен помочь ребенку в исправлении грубых орфографических ошибок или даже в клавиатурном наборе (впрочем, трудно поверить в то, что ребенок уступит учителю, даже уважаемому, право набрать на клавишах сочиненный им текст), но цель занятий — научить ребенка делать, а не сделать за него.

Занятие 1

Про программы, пиктограммы и курсор

Тема занятия. Выбор и запуск программ.

Цель занятия. Познакомить с основными элементами управления программой — рабочее поле, пиктограмма, курсор.

Программное обеспечение. Программы “Корешки и вершки”, “Гости”.

План

1. Беседа о технике безопасности
2. Рабочее поле и пиктограммы
3. Многообразие форм курсора.
“Корешки и вершки”
4. Многовариантная программа “Гости”
5. Координаты
6. Конструирование задачи

Комментарии

1. Беседа о технике безопасности

Всякому систематическому общению с техническим устройством предшествует инструктаж по технике безопасности. Правда, когда обучаемый контингент — дети, а техническое устройство — персональный компьютер, инструктаж превращается в душевную беседу. Учитель говорит всего о двух (но важнейших!) требованиях к поведению детей около компьютера — чистота и дисциплина, убедительно аргументируя эти требования.

2. Рабочее поле и пиктограммы

Программы, используемые в курсе “Зимние вечера”, работают как в двухпанельной программной оболочке DOS, так и в оконной среде Windows. Ничто не мешает учителю в нужный момент вызвать Norton Commander (или Far, или другую имеющуюся на ма-

шине программу-оболочку MS-DOS), чтобы поработать в среде DOS (если пользователь к ней привык). Однако детям целесообразно сразу представить рабочее поле более современной оконной среды Windows: большинство школьных (и тем более домашних) компьютеров располагают такими средствами.

Готовя компьютер к занятию с младшими школьниками, полезно позаботиться о презентабельности рабочего поля. Зачастую пользователь, злоупотребляя возможностями Windows, размещает на рабочем столе большое число ярлыков. При индивидуальном использовании компьютера насыщенность экрана несущественна и может определяться привычкой или персональными предпочтениями. Новичкам-малышам лучше предложить обстановку, в которой будет меньше предпосылок для отвлечения их внимания. Целесообразно убрать (возможно, на время) те пиктограммы, которые на занятиях не понадобятся.

С другой стороны, рекомендуется вынести на рабочее поле пиктограмму WINT.ICO, которая поставляется вместе с программным пакетом “Зимние вечера”. (Для пользователя, умеющего манипулировать ярлыками рабочего стола, этой информации, вообще говоря, достаточно. Тем, кто будет делать установку пиктограммы впервые, рекомендуется воспользоваться приводимой ниже инструкцией (с которой читатель, возможно, уже познакомился в файле READ.ME, хранящемся в корне пакета “Зимние вечера”):

Инструкция по установке ярлыка-пиктограммы пакета “Зимние вечера” в рабочем поле Windows

Внимание: если в указаниях явно не оговорено, какой из кнопок надо щелкать, используйте левую кнопку мыши.

А. Запишите файлы-архивы SOFT.EXE и SOFT.R00 в некотором удобном вам каталоге и разверните их, запустив файл SOFT.EXE.

Для примера, который используется дальше, примем, что вы создали для размещения пакет-каталог WINTER на диске С:

В. Создайте ярлык

- В.1. Для этого в свободном месте рабочего поля, на которое не вызвана ни одна из программ, надо щелкнуть правой кнопкой мыши.
- В.2. В выпавшем меню щелкните в позиции Создать.
- В.3. Выпадает подменю, в котором теперь надо щелкнуть в позиции Ярлык.
- В.4. Выпадает окно, которое называется Создание ярлыка. Под его единственным окном — кнопка Обзор, а в окне — вертикальная черточка курсора.
- В.5. Щелчок по надписи Обзор выводит на экран новое окно, которое называется Обзор. В верхней части этого окна — два поля. В большом поле окна видны каталоги файловой системы вашего компьютера в таком же виде, как они представлены в программе Проводник: в такой системе, двигаясь от каталога к каталогу, легко найти нужный объект — требуемый файл.

В соответствии с нашим примером вам предстоит пройти путь:

C:\WINTER\GO.BAT

- В.5.1. Верхнее однострочное поле окна Обзор называется Папка. Здесь видно имя текущего каталога. Установите в этом поле указание на диск С:

Если в исходном состоянии (при открытии окна Обзор) указатель на диск С: не записан в верхнем окне, следует поставить курсор на кнопку, находящуюся справа от окна (со стрелкой, загнутой кверху), и нажать на нее (возможно, несколько раз; каждое нажатие — это возврат по дереву файловой системы на один уровень выше), чтобы в конце концов в поле Папка появилась надпись C:\

- В.5.2. Найдите среди каталогов диска С: тот, который вам нужен, — WINTER и щелкните левой кнопкой по этой надписи. Название каталога станет подсвеченным, теперь надо щелкнуть стрелкой курсора по кнопке Открыть. Тогда в окне файловой системы вы увидите содержимое каталога WINTER.

Если ваш файл лежит достаточно глубоко в дереве файловой системы, вам может понадобиться сделать несколько таких шагов — перемещений от каталога к входящему в него подкаталогу.

В последнем нужном вам каталоге вы обнаруживаете файл GO.BAT. Щелчок по надписи GO.BAT, и вы возвращаетесь в поле Создание ярлыка, где в окне обзора высвечивается теперь полный путь к файлу, запускающему пакет “Зимние вечера”:

C:\WINTER\GO.BAT

- В.6. Теперь нажимайте кнопку Далее в нижней части поля.

Тогда появится новое поле, которое называется Создание названия программы. В однострочном окне, над которым написано “Укажите название ярлыка”, следует написать имя пакета — “Зимние вечера”, затем нажать кнопку Готов в нижней части поля.

Вспомогательные поля убираются с экрана, а на экране — прежнее состояние рабочего поля, на котором видно новый, только что созданный ярлык, правда, с “чужой” пиктограммой.

С. Сменить значок

- С.1. Щелкните по значку правой кнопкой мыши и войдите в позицию Свойства.
- С.2. Появляется поле, называемое Свойства: “Зимние вечера”. В верхней строке этого поля — несколько позиций-закладок. Вам следует щелкнуть левой кнопкой по позиции Программы. В появившемся после этого поле надо найти кнопку Сменить значок и щелкнуть по ней.
- С.3. Следующее выпадающее поле называется Смена значка. Из трех кнопок, находящихся в нижней части поля, надо выбрать кнопку Обзор...
- С.4. Теперь, действуя таким же способом, как был указан в п. В, надо добраться в нашем каталоге до названия нужного значка-файла — WINT.ICO.
- С.5. Открытие этого файла возвращает вас в поле Изменение значка. В этот момент вы можете видеть в этом поле требующийся значок. Сейчас надо нажать клавишу “Ok”, она возвратит вас в поле Свойства: “Вечера”.

Еще один щелчок на “Ok” — и вы снова в рабочем поле Windows, и через одну-две секунды вы можете наблюдать созданный вами ярлык пакета “Зимние вечера”.

Д. Перемещение пиктограммы

Если вам действительно не нравится расположение новой пиктограммы в рабочем поле, вы можете ее переместить.

Нажмите на кнопку (любую) мыши и, не отпуская ее, поведите мышь в другую точку рабочего поля, где вы хотите разместить пиктограмму. Установите ее на нужное место, отпустите кнопку. Пиктограмма установлена.

Е. Запустите программу

Остается только проверить, как осуществляется запуск программы. Поставьте курсор на пиктограмму “Зимние вечера” и щелкните левой кнопкой мыши.

Внимание: на некоторых машинах в этой ситуации надо делать не одинарный, а двойной щелчок. Это зависит от настроек системы.

Одна из возможных неприятностей:

Если при пробном запуске программы обнаруживается аварийное сообщение, надо щелкнуть правой кнопкой и выбрать позицию Свойства. В поле Свойства выбрать позицию Программы. В появившемся окне выберите на сей раз кнопку Дополнительно.

Убедитесь, что в маленьком квадратном окошке, около которого видна надпись Режим MS-DOS, указана меточка-галочка. Щелкнув по этой галочке левой кнопкой, вы погасите ее, а строкой выше будет ярко гореть надпись Автоматически предлагать переход в режим MS-DOS. Маленькое квадратное окошко около нее — пустое. Щелкните по нему левой кнопкой — в нем появится метка-галочка. Возвращайтесь в рабочее поле (“Ok”, “Ok”), теперь программа должна нормально запускаться.

Имеет смысл выбрать на время занятий подходящие экранные обои — спокойную, не раздражающую картинку. Лучше вообще иметь чистый фон, чем дополнительный отвлекающий раздражитель.

Войдя в Панель управления, выполните необходимые настройки, чтобы установить крупные значки (пиктограммы-ярлыки) и крупные символы. Это облегчит детям ориентацию в рабочем поле.

Завершив системные установки, надо опробовать запуск пакета “Зимние вечера”, а затем пройти по страницам этого пакета. О навигации по пакету можно прочитать в документации к программе Монитор (запуск — щелчок по пиктограмме Описание Монитора на заглавной, конфигурационной странице пакета). Для подготовки к первым занятиям достаточно ознакомиться пока с “навигационным”, четвертым разделом описания.

3. Многообразие форм курсора.

“Корешки и вершки”

Продемонстрировав (но не акцентируя внимание) включение компьютера, учитель умышленно отказывается от обсуждения операций выключения. Здесь детская самостоятельность чревата неприятностями. Позднее они освоят стандартные для Роботландии способы выхода из программ и даже операции выключения компьютера. Сегодня учитель занимается этим сам. Поэтому специфику управления двумя программами сегодняшнего занятия дети не увидят. Программы “Корешки и вершки” и “Гости” не являются разработками Роботландии, они выполнены Ассоциацией “Компьютер и детство”, которая с этими программами участвовала в созданном Роботландией проекте “Информационная

культура”. (В частности, обе программы рекомендуется запускать на занятии однократно. При необходимости неоднократного последовательного обращения к программам следует после каждого завершения программ “Корешки и вершки” или “Гости” выходить из пакета “Зимние вечера” на рабочее поле, после этого можно сразу же возвращаться в программу.)

Учитель знакомится с описаниями обеих программ на странице “Про курсор” пакета “Зимние вечера”, обратив внимание на выход из программ — **[Ctrl] + [Q]** и отключение звука **[Ctrl] + [S]**.)

При демонстрации программ возможности звукового их сопровождения — поздравительная мелодия в конце выполнения программ — служат положительным эмоциональным воздействием. Но в последующих запусках, и в частности при работе со вторым и третьим вариантом, звук лучше отключить: многократные звуковые повторы и тем более наложения звучаний нескольких компьютеров создают раздражающий диссонанс. В таких ситуациях без наушников работать не рекомендуется: надо отключать звук.

4. Многовариантная программа “Гости”

Программа “Гости” привлекает детей не арифметическим содержанием задач (они весьма просты для третьеклассников; упоминавшийся курс “Информационная культура”, например, фиксирует место этой программы в самом начале второго класса как элемент повторения предшествовавшего материала), а увлекательным ребенком многоплановым интерфейсом, в котором следует обратить внимание на многообразие форм курсора. В задаче “Гости” курсор не только отличается от веночка “Корешков и вершков”, но и на каждом этапе решения задачи он принимает новый образ: сначала это симпатичный зверек, затем яркая стрелочка, потом цветная прямоугольная рамка (разного цвета в разных вариантах) и, наконец, окантованный квадрат с вопросительным знаком для операндов арифметических операций.

Неожиданные повороты сюжета и интерфейса, помноженные на небольшой пока опыт клавиатурного набора, приводят к тому, что содержательно несложная задача может занимать значительную по времени часть занятия. Учитывая, что первые два варианта программы очень схожи (они отличаются лишь числом элементов городского пейзажа), учитель может при дефиците времени отказаться от одного из этих вариантов — либо пропустить первый вариант, начиная с чуть более сложного второго, либо после первого варианта непосредственно переходить к третьему — и полудить при этом дополнительный ресурс времени.

5. Координаты

Ключевое понятие, вводимое программой “Гости”, — координаты. Для третьеклассников это слово звучит впервые: традиционный школьный курс математики вводит координаты существенно позднее.

Роль этапа занятия, отведенного программе “Гости”, — пропедевтика этого фундаментального математического понятия. Наглядность клетчатой среды, перенесенной на прямоугольные квадраты игрушечного города, представляет для этой цели эффективный дидактический инструмент.

6. Конструирование задачи

Третий вариант программы “Гости” существенно отличается от двух предыдущих тем, что ребенок сначала сам конструирует среду для решения задачи — выбирает домики для их строительства в городе, представляет их по своему усмотрению, выделяет контур искомого домика — и лишь затем приступает к ре-

шению задачи. Это тоже пропедевтика — самое первое использование идей программного конструирования.

Когда позднее, год или два спустя, учащиеся смогут в рамках Роботландского сетевого университета приступить к изучению курсов “Компьютерное конструирование-1” и “Компьютерное конструирование-2”, учитель будет иметь основание напомнить детям, что еще в “Зимних вечерах” подобные задачи уже встречались.

Такая пропедевтическая роль третьего варианта программы “Гости” заставляет считать ее обязательной в курсе. Рекомендуется строить занятие так, чтобы на рассмотрении третьего варианта программы не пришлось экономить время.

Занятие 2

Что такое алгоритм

Тема занятия. Интуитивное определение алгоритма.

Цель занятия. Дать представление о формализованной записи плана решения.

Программное обеспечение. Программа “Перевозчик”.

План

1. Нужен четкий план
2. Переезд через реку
3. Как задача о перевозе решается на компьютере

Комментарии

1. Нужен четкий план

Основной вывод, который должен быть сделан из обсуждения неформально описываемого рецепта манной каши, — такого рода описания неточны и приводят либо к двусмысленному трактованию, либо к непониманию задачи ее исполнителем. Дина как исполнитель плана его понимает, а исполнитель Тим — не понимает. Массовость — одно из основных качеств алгоритма — достигается тогда, когда его предписания записаны в виде четкой последовательности элементарных этапов, однозначно задаваемых введенными соглашениями — правилами записи. Формальность (и доступность) трактовки предписаний делает алгоритм доступным широкому кругу исполнителей.

Вопросы и упражнения первого пункта занятия (или подобные им задания, предложенные учителем) могут стать предметом обсуждения.

2. Переезд через реку

В формулировке задачи перевоза через реку волка, козы и капусты важно соблюдение ограничений. В учебнике примеры нарушений рождаются в ходе обсуждения рекомендаций Дины и Тима. Хорошо, если такого рода примеры-нарушения генерируются на вашем занятии. Если дети молчат, можно рассмотреть примеры из параграфа про “Перевозчика”.

Притча про “Перевозчика” буквально просится быть сюжетом инсценировки. Правда, дедушке Фёрстову

трудно воспользоваться этим методическим приемом. Но если на вашем занятии присутствует больше детей, чем слушателей у дедушки Фёрстова, то инсценировка (особенно с подготовленными масками или надеваемыми на голову бумажными лентами-обручами, на которых написано крупными буквами: “Волк”, “Коза”, “Капуста”) даст должный эффект. Более того, может оказаться, что желающих поучаствовать в такой инсценировке будет много. Этим можно воспользоваться для того, чтобы несколько позднее провести демонстрацию второго способа решения обсуждаемой задачи.

Здесь следует проявить осторожность в пропедевтических целях. Хотя слово “исполнитель” было бы очень легко воспринято детьми — как участниками инсценировки, так и зрителями, — произносить его на этом занятии не надо. Дело в том, что участники инсценировки, которых режиссер спектакля или зрители могли бы назвать исполнителями ролей волка, козы и капусты, с точки зрения информатики вовсе не исполнители, а элементы среды.

Исполнителем же является тот школьник, который, выслушав команду учителя (или, еще лучше, ученика — автора созданного им алгоритма), берет за руку одного из бессловесных персонажей инсценировки и “перевозит” его через реку (переводит через проход между партами, через нарисованную мелом на полу рекреации “реку”). Впрочем, в этой сценке молчат не только “перевозимые” персонажи, но и исполнитель. Однако понятие исполнителя — это предмет следующих занятий-вечеров. Сейчас важно позаботиться, чтобы инсценировка не внесла терминологическую путаницу в эти предстоящие обсуждения.

В “спектакле” “Перевозчик” импровизации не поощряются. Учитель отмечает все отклонения от правил ввода (записи, произнесения) команд и правил их выполнения.

Инсценировка или иная модель выполнения алгоритма (дедушка использует с этой целью различные предметы; вполне подходит для этого магнитная доска) обязательно сопровождается формальной записью команд на доске или экране проектора. О зафиксированной последовательности команд можно говорить как об алгоритме в одной из форм его представления.

3. Как задача о перевозе решается на компьютере

После инсценировки дети в нетерпении начать записывать алгоритм на компьютере. Однако (возможно, сдерживая такой порыв) надо потратить достаточно времени на обсуждения элементов интерфейса: правила записи в окно ввода команд, объяснение элементов стандартной информационной строки.

Учитель ведет подробный рассказ о справочном режиме, вызываемом по нажатию клавиши **[F1]**, подчеркивая общность элементов интерфейса для большинства роботландских программ, которые позднее придется изучать: справочное окно, лента протяжки, бегунок. Еще до составления алгоритма на компьютере дети должны прочитать справку и вернуться в задачу (по **[Esc]**).

Требует подробного рассказа указание информационной строки о способе выхода из программы — аккорд **[Ctrl] + [Break]**. Учитель вводит понятие клавишного аккорда, говорит об особенностях набора компьютерных аккордов (в отличие, например, от фортепианных). Однако выполнение аккорда учени-

ками откладывается на небольшое время. Дети самостоятельно наберут аккорд завершения программы при переходе от одного решения задачи ко второму.

До начала упражнения важно предусмотреть возможные аварийные сообщения. Еще не начав нажимать на клавиши, ребенок должен уметь привести примеры ситуаций, в которых может появиться сообщение “Не могу”.

Задача о перевозе имеет два решения. На каждом рабочем месте должны быть воспроизведены оба решения. Если за компьютером сидят два ученика, то учитель может организовать их деятельность в зависимости от темперамента учеников: либо они набирают по очереди отдельные команды алгоритма, либо сначала один ученик полностью воспроизводит на экране одно решение, а затем его напарник другое.

Если “Перевозчику” отводится одно занятие, то, кроме воспроизведения алгоритма на экране, надо побеспокоиться о том, чтобы запись алгоритма осталась и в тетрадах учеников. Другой вариант фиксации алгоритма в тетради — домашнее задание: учитель формулирует задание, а в начале следующего занятия собирает тетради для проверки.

Занятие 3

Строковый редактор

Тема занятия. Корректировка ошибок клавиатурного набора.

Цель занятия. Сформировать навыки исправления ошибок.

Программное обеспечение. программный тренажер “Правилка”.

План

1. Какие бывают ошибки
2. Правилка показывает мультик
3. Упражнения “Правилки”

Комментарии

1. Какие бывают ошибки

Классификация ошибок на оперативные и отсроченные, конечно, условная. Она вызвана прежде всего используемым программным обеспечением — программой-тренажером “Правилка”, в которой не предусмотрены практические упражнения с использованием клавиши обратного пробела **[BackSpace]**.

Впрочем, в свое время при разработке технического задания на реализацию программы “Правилка” подробно обсуждались ограничения на предметную область ее использования. В частности, простота освоения малышами клавиши стирания, отмеченная в ходе многочисленных наблюдений за поведением детей у компьютера, и привела к тому, что в упомянутом техзадании клавиша обратного пробела была “вынесена за скобки”. Для оправдания этого решения и понадобилась обсуждаемая в начале занятия условная классификация.

В адрес “Правилки” (точнее, ее режима “Пропущенного символа”) иногда звучит еще одно критическое замечание: использованный в “Правилке” строковый ре-

дактор использует только режим замены и игнорирует режим вставки, столь распространенный в современных профессиональных текстовых редакторах. Дело в том, что “Правилка” — это первый шаг в освоении текстового редактирования. Как всегда, на первых шагах важно сформировать не автоматизм практических навыков (в случае “Правилки” — навыков клавиатурного набора), а понимание основных механизмов редактирования. Такая “ручная” вставка пропущенного символа, которую требует “Правилка”, и есть тот самый первичный навык, который учит пониманию механизма. “Правилка” работает на первом витке дидактической спирали темы “Редактирование текстовой информации”. Возврат в этой теме на новом витке спирали (например, при освоении текстовых редакторов в составе современных информационных технологий в старших классах) к автоматизированным или переключаемым приемам вставки опирается на прочные навыки корректировки ошибок, сформированные “Правилкой” у малышей.

2. “Правилка” показывает мультик

Наступает время знакомства с “Правилкой”. Сразу же после щелчка по пиктограмме “Правилка” появляется возможность протянуть мостик между прошлым занятием и нынешним: единство стандартной информационной строки у обеих программ. Идеальный вариант — обнаружение этой параллели детьми (пусть даже вследствие провокационных “наводок” учителя). Но и при отсутствии детской инициативы учитель привлекает внимание учеников к этому важному обстоятельству: единство интерфейса в роботландских программах позволяет одинаково реагировать на технические, интерфейсные детали управления программы, сосредотачиваться, не отвлекаясь второстепенными деталями, на содержании обсуждаемой на уроке задачи и экономить учебное время.

Два первых момента встречи с программой “Правилка” комментируются учителем по-разному.

Самое первое сообщение на экране

Введите имя тестов на диске: PRAV.TST

следует попросту проигнорировать — нажать клавишу выполнения. После этого компьютер переходит к основной части программы.

Что происходит на самом деле? Компьютер запрашивает имя файла, из которого “Правилке” предстоит читать задания, предлагаемые детям. По умолчанию, если не поступило специальных указаний, “Правилка” читает эти задания-тесты из встроенного в пакет файла с именем PRAV.TST. В этом случае нажатие на клавишу выполнения при первом в программе “Правилка” запросе означает, что пользователь соглашается с таким предложением компьютера.

Однако у пользователя может, вообще говоря, быть и другое решение о продолжении программы. Если он предварительно заготовил свой файл с тестами, которые ему представляются более интересными в методическом плане для предстоящего занятия, то на вопрос об имени тестового файла он перед нажатием клавиши выполнения должен набрать имя этого файла вместо PRAV.TST.

В учебнике ничего не говорится о способе создания тестового файла. Здесь мы тоже не останавливаемся на этом. Построение тестового файла — это типовая (хотя и очень простая) задача программного конструирования, и ее место, несомненно, на следующем курсе Роботландского сетевого университета. Тем не менее учитель, который желает заменить PRAV.TST для обогащения набора упражнений или просто для удовлетворения собственной естественной любознательности, имеет возможность обратиться за необходимой информацией в описание программы “Правилка” — щелкнуть по пиктограмме “Описание”, расположенной под пиктограммой “Правилка”.

Рекомендация. Если вы действительно хотите так поступить, то дополните каталог REDSTR файлами из каталога MICRON (выйдя для этого, например, в Far).

Второй момент, заслуживающий внимательного обсуждения на начальном этапе “Правилки”, — сложное меню. Для учителя он не составляет труда, ученику сложное меню необходимо, чтобы увидеть два действия на входе в каждую из трех основных (первых) позиций меню “Правилки”. Впрочем, сложное меню важно не столько как элемент конкретной программы “Правилка”, сколько как качество универсального интерфейсного механизма многочисленных и разнообразных программных средств.

Позиция “Обучение”, с которой рекомендуется начинать работу в каждой из позиций основного меню “Правилки”, — это своеобразный справочный режим.

Главное его отличие, например от справки “Перевозчика”, — динамика изображения. Это дает основание называть работу в этом режиме “мультиком”.

Дети стремятся все записанные в режиме “Обучения” указания тут же выполнить на клавиатуре. Учителю предстоит усмирить этот пыл: сейчас надо только смотреть, а уже посмотрев, перейти (по клавише перехода ) в режим “Упражнения”. Однако смотреть надо внимательно (об этом учитель обязательно предупреждает детей), потому что все действия, предстоящие в режиме “Упражнения”, надо делать в строгом соответствии с теми правилами, которые продемонстрированы в позиции “Обучение”.

3. Упражнения “Правилки”

Две типовые реакции программы на нажатие клавиши выполнения — “Нет ошибок” и “Есть ошибки” — понятны детям без комментариев. Зато их, несомненно, удивит сообщение “Плохо исправлено”. Для объяснения следует повторить важность наблюдений за демонстрационными действиями “Правилки” в режиме “Обучения”. Учитель замечает: в результате получилось безошибочное слово, но само исправление сделано вовсе не так, как этому научила “Правилка”.

Даже одна погрешность такого рода должна спровоцировать общую для всей группы приостановку упражнений. В ходе этой приостановки можно поговорить не только об алгоритмах корректировки ошибок, но и о понятии тренажера. Почему “Правилка” — тренажер? Какие качества человека тренирует “Правилка”? Какие еще бывают тренажеры? Для чего они служат? Даже домашнее задание на темы о тренажерах и их роли в обучении можно сформулировать в этот момент.

Далее занятия продолжаются в свободном и индивидуальном темпе. Асинхронность этого этапа занятия неизбежна.

Когда упражнения в трех первых позициях основного меню завершены и наступает время приступить к позициям “Смесь”, “Суперсмесь” и “Контрольная работа”, учитель предупреждает, что в трех последующих режимах не будет подрежима “Обучения”. Дети уже готовы ответить самостоятельно на следующий за этим предупреждением вопрос “почему?”, но даже если ответ и не получен, надо объяснить детям:

— Существует всего три типа ошибок — “Неверный символ”, “Лишний символ” и “Пропущенный символ”. Никаких других ошибок при клавиатурном наборе не бывает. Если вы внимательно посмотрели три первые позиции основного меню “Правилки”, это значит, что вы уже все знаете и режим “Обучения” вам попросту не нужен.

Занятие 4

Побеседуем с компьютером

Тема занятия. Диалоговые программы.

Цель занятия. Закрепить навыки работы со строковым редактором.

Программное обеспечение. Программы “Привет” и “Сказка”.

План

1. “Привет”.
2. Как работает программа “Привет”.
3. Сказка по выбору.

1. Комментарии

У программы “Привет” двойное предназначение — закрепить навыки строкового редактирования, полученные в работе с тренажером “Правилка”, и сформировать представление о диалоговой программе.

Хотя на предстоящем занятии учитель пытался показать в “Правилке” элементы диалога, именно диалоговые возможности “Привета” изумляют ребят, дают им пищу для многочисленных вопросов. Еще бы: компьютер предлагает не задание, а адресованный ученику конкретный вопрос. На вопрос надо отвечать не нажатием какой-либо управляющей клавиши, а писать такой же “человеческий” ответ, как в любом диалоге. Более того, оказывается, что, кроме простого “да”, можно отвечать “Да”, “ДА” и даже “конечно”. Еще более интересной кажется детям возможность ответить компьютеру словами-характеристиками, называя Буратино деревенным, Карабаса — злым, Двоймовочку — маленькой.

Работа с программой “Привет” неизбежно вызывает веселое оживление в группе, находящееся на грани допустимых шумов. Не следует стремиться наводить дисциплину и порядок. Эмоциональная обстановка, создаваемая общением с программой “Привет”, как нельзя лучше иллюстрирует главный лозунг всего курса — “Компьютер — твой друг!”.

Диалог с “Приветом” рекомендуется провести по меньшей мере дважды. Это особенно актуально, если детей в группе больше, чем компьютеров. В этом случае повторное выполнение программы необходимо уже для того, чтобы ученики в парах могли сменить свои роли.

2. Как работает программа “Привет”

После обсуждения ученических гипотез о машинном конструировании диалогов-сказок учитель может предложить детям игру, аналогичную той, в которую играет с детьми дедушка Ферстов. Можно предположить, что вы окажетесь в более выигрышном положении, чем главный персонаж учебника, и у вас окажется больше игроков-учеников, чем у дедушки Ферстова внуков. Ведь не случайно дедушка смог сколько-нибудь эффективно провести эту игру только потому, что в этот момент “подвернулись” гости — Наташа и Артем.

Сочетание двух механизмов — случайная выборка и готовые шаблоны-образцы — служит объяснением как последней игры в предложении, так и только что рассмотренных перед этим диалогов с программой “Привет”.

3. Сказка по выбору

Сказка, “рассказанная” программой “Привет”, получилась небольшой. Она целиком размещается на экране. Компьютер мог бы рассказать и длинную сказку, но тогда ее надо непременно разделить на порции-страницы, каждая из которых занимает экран: прочитал одну порцию, нажимаешь клавишу, появляется следующая порция и т.д. Компьютерная сказка, при которой каждой странице соответствует порция информации на экране, конечно, возможна, но малоинтересна, потому что ничего нового по сравнению с обычной, книжной сказкой здесь не появляется. Разница лишь в том, что вместо листания страниц вы нажимаете клавишу на клавиатуре.

Тут полезно вспомнить часто встречающиеся в сказках ситуации, когда герой оказывается на развилке дорог. Он размышляет, куда идти дальше. От выбора пути зависит судьба героя и, значит, дальнейшее течение сказки. В книжной сказке персонаж после некоторых размышлений выбирает одну из дорог, а читатель переворачивает очередную страницу, так и не узнав подробности собы-

тий, которые могли бы развиваться при другом выборе дороги на развилке. Если такую сказку станет “рассказывать” компьютер, то, воспользовавшись его умением вести диалог, можно поручить человеку, разговаривающему с компьютером, выбор развилки на дорогах, или выбор одного из нескольких возможных решений, или вообще одно из возможных продолжений сказки.

Одна из таких компьютерных сказок представляет собой сюжет очередного упражнения. На странице “Строковый редактор” ученики выбирают пиктограмму с портретом дедушки-сказочника, который готов начать свой рассказ. Как только на экранах появляется первая страница сказки, учитель поясняет структуру кадра:

— На каждой странице сказки вы увидите три части. Первая часть — это текст в верхней части экрана. Он называется повествованием. Средняя часть экрана занята иллюстрацией к сказке. Текст в нижней части экрана называется управлением. В нем выделены цветом два (а иногда и три) разных слова. Набрать на клавиатуре одно из этих слов означает выбрать путь дальнейшего продолжения сказки. Так, если в первом кадре вы наберете на клавишах слово ТЕРЕМОК, то ваша сказка будет посвящена коту, собаке и петуху, если же будет набрано слово ЗВЕРИ, то героями сказки станут заяц, гусь и медвежонок.

— А если написать не ТЕРЕМОК и не ЗВЕРИ, а что-нибудь другое?

— В этом случае, а также в случае неверно написанного требуемого слова (например, вы написали слово ТИРЕМОК вместо правильного ТЕРЕМОК) компьютер признается, что не понимает вас, и попросит ввести все-таки одно из тех слов, которые нужны для управления дальнейшим ходом сказки.

После этого дети продолжают читать сказку самостоятельно. Учитель не настаивает на каком-либо конкретном пути по дереву сказки. Получается столько сказок, сколько машин в классе. Это, правда, не способствует спокойной и размеренной обстановке на занятии. Но ею приходится жертвовать во имя укрепления авторитета компьютера как дружелюбного и интересного собеседника.

Закончив чтение сказки, можно запустить программу еще раз. Речь идет о повторении той же самой программы, но поскольку на каждом кадре начиная с самого первого можно на этот раз выбирать другие продолжения, то и сказка окажется другой. Это особенно важно, если за компьютером сидят двое учеников, спорящих о вариантах выбора. Тогда при первом прочтении сказки приоритет выбора получает один из учеников, а при повторном чтении — другой.

Конструируя сказку по своему выбору, дети увлечены сюжетом. Учитель же не должен забывать, что одно из назначений программы “Конструктор Сказок” состоит в закреплении навыков работы со строковым редактором. В частности, учитель еще на первых кадрах сказки обращает внимание детей на регистр, в котором следует работать: ключевые слова сказки (выделенные слова, управляющие выбором кадров) следует набирать в точности так, как они написаны в управляющем тексте, то есть заглавными буквами. Следовательно, надо внимательно следить за индикацией регистра в стандартной информационной строке и уметь устанавливать и фиксировать регистр заглавных букв. Навыки работы со строковым редактором потребуются и при корректировке возможных ошибок: исправлять неверно набранные ключевые слова следует по тем правилам, которые дети освоили при изучении программы “Правилка”.

Подписка круглый год на любой срок!

Дорогие читатели педагогических изданий «Первое сентября», напоминаем, что теперь подписка на эти периодические издания осуществляется круглый год и на любой срок от одного месяца до года.

Предлагаем на выбор четыре способа осуществить подписку.

- 1. Во всех отделениях почтовой связи РФ.** Спрашивайте каталог агентства «Роспечать» на первое полугодие 2001 года. Наши издания расположены в алфавитном порядке в разделе «Газеты». По этому каталогу можно осуществить подписку со следующего месяца и на любой срок до конца года с доставкой в школу, на дом или до почтового отделения.
- 2. Издательская подписка.** Заполните предложенный ниже купон и отправьте по адресу: **121165, Москва, ул. Киевская, д. 24, редакция газеты «Первое сентября».** В ответ вам будет прислан счет и заполненное извещение для оплаты в любом отделении Сбербанка РФ. Во многих случаях издательская подписка оказывается дешевле, так как редакция берет на себя стоимость доставки издания до почтового ящика. В счете будет указана только каталожная цена подписки (газета «Первое сентября» – 24 руб. за месяц, любое приложение – 40 руб. за месяц).
- 3. По телефону (095) 249-4758, с 10 до 17 часов по московскому времени.** Вежливый оператор примет заказ и направит по вашему адресу счет, который можно будет оплатить в любом отделении Сбербанка РФ и получить издания с доставкой. И в этом случае редакция берет на себя стоимость доставки по адресу. В счете будет указана только каталожная цена подписки.
- 4. Через ИНТЕРНЕТ.** Заказ можно осуществить в интерактивном режиме непосредственно на сайте объединения педагогических изданий «Первое сентября» www.1september.ru, заказав либо сами издания с доставкой на дом по каталожной цене, либо электронные версии. Последние обойдутся в два раза дешевле.

КУПОН ДЛЯ ИЗДАТЕЛЬСКОЙ ПОДПИСКИ:

1. Укажите в купоне вашу фамилию, имя, отчество, индекс и почтовый адрес.
 2. В столбце «С МЕСЯЦА», напротив интересующих вас газет, укажите месяц и год начала подписки.
 3. В столбце «КОЛ-ВО МЕСЯЦЕВ» укажите число месяцев, на которое оформляется подписка.
 4. В столбце «КОЛ-ВО ЭКЗ.» укажите желаемое количество экземпляров.
- Например, подписка на два экземпляра «Информатика» с февраля 2001 года на 6 месяцев выглядит так:

Информатика	февраль 2001	6	2
-------------	--------------	---	---

4. Вырежьте и вышлите купон в конверте с пометкой «Издательская подписка» по адресу: 121165, Москва, ул. Киевская, д. 24, «Первое сентября».
 5. Вам будет выслана заполненная персонально для вас квитанция Сбербанка РФ, которую вы сможете оплатить в любом отделении банка. Организациям будут высылаться для безналичной оплаты счет и счет-фактура.
- Внимание! Организаций! Для получения счета необходимо выслать вместе с купоном реквизиты платящей организации.**

Телефон для справок (095) 249-4758, с 10 до 17 часов по московскому времени.

КУПОН Подписка ЗАПОЛНЯЕТСЯ ПЕЧАТНЫМИ БУКВАМИ!

Первое сентября
Объединение педагогических изданий

ФАМИЛИЯ

ИМЯ

ОТЧЕСТВО

ИНДЕКС

АДРЕС

Название	С МЕСЯЦА	КОЛ-ВО МЕСЯЦЕВ	КОЛ-ВО ЭКЗ.
Газета «Первое сентября»			
Английский язык			
Библиотека в школе			
Биология			
Воскресная школа			
География			
Дошкольное образование			
Здоровье детей			
Информатика			
Искусство			
История			
Литература			
Математика			
Начальная школа			
Немецкий язык			
Русский язык			
Спорт в школе			
Управление школой			
Физика			
Французский язык			
Химия			
Школьный психолог			

Подписка производится только на территории РФ

00-1-00

Гл. редактор
 С.Л. Островский
Зам. гл. редактора
 И.Н. Фалина
Редакция:
 Е.В. Андреева
 Н.Л. Беленькая
 Л.Н. Картвелишвили
 Н.П. Медведева
Дизайн и верстка:
 Н.И. Пронская
Корректоры:
 Е.Л. Володина,
 С.М. Подберезина

©ИНФОРМАТИКА 2001
 выходит четыре раза в месяц.
 При перепечатке ссылка
 на ИНФОРМАТИКУ обязательна,
 рукописи не возвращаются

**Адрес редакции
 и издателя:**
 121165, Киевская, 24
 тел. 249-48-96
 Отдел рекламы
 тел. 249-98-70

Учредитель: ООО "Чистые пруды"
 Зарегистрировано в Комитете РФ по печати
 20.03.1997.
 Регистрационный номер 015873.
 Отпечатано в типографии ОАО ПО "Пресса-1".
 125865, ГСП, Москва, ул. "Правды", 24.
 Тираж 6000 экз.
 Срок подписания в печать по графику 20.12.2000.
 Номер подписан 20.12.2000.
 Заказ № **18494** Цена свободная

ИНДЕКС ПОДПИСКИ
для индивидуальных подписчиков 32291
комплекта приложений 32744

Тел.: (095)249-31-38, 249-33-86. Факс (095)249-31-84

Internet: inf@1september.ru
WWW: http://www.1september.ru

Номера "Информатики" (в ограниченном количестве) бесплатно распространяются
 в офисе ведущей российской антивирусной фирмы "ДиалогНаука".
Адрес: Москва, ул. Вавилова, д. 40, офис 103. Тел. 137-01-50.

Комплект
 приложений,
 главный редактор —
 А.С. Соловейчик

Английский язык (Е.В. Громушкина), индекс подписки — 32025; **Библиотека в школе** (О.К. Громова), индекс подписки — 33376; **Биология** (Н.Г. Иванова), индекс подписки — 32026; **Воскресная школа** (монах Киприан (Яценко), индекс подписки — 32742; **География** (О.Н. Коротова), индекс подписки — 32291; **Здоровье детей** (А.У. Лекманов), индекс подписки — 32033; **Информатика** (С.Л. Островский), индекс подписки — 32291; **Искусство** (Н.Х. Исмаилова), индекс подписки — 32584; **История** (А.Ю. Головатенко), индекс подписки — 32028; **Литература** (Г.Г. Красухин), индекс подписки — 32029; **Математика** (И.А. Соловейчик), индекс подписки — 32030; **Начальная школа** (М.В. Соловейчик), индекс подписки — 32292; **Русский язык** (Л.А. Гончар), индекс подписки — 32383; **Управление школой** (А.И. Адамский), индекс подписки — 32652; **Физика** (Н.Д. Козлова), индекс подписки — 32032; **Французский язык** (Л.А. Гончар), индекс подписки — 33371; **Химия** (О.Г. Громушкина), индекс подписки — 32034; **Школьный психолог** (Н.Д. Козлова), индекс подписки — 32898.