

Колпачки на клавишах клавиатуры

Переименовой клавиши при перепрограммировании их функций.
Если предохранительный колпачок на клавише клавиатуры (речь о нем шла в совете 1В) сделать эластичным и написать на нем функцию клавиши, то станет более удобным ее перепрограммирование. Для некоторых ЭВМ выпускаются, кстати, пластиковые эластичные накладки на клавиатуру, повторяющие ее форму. Такой нехитрый прием позволяет легко переходить, например, на работу с другим алфавитом или даже с другим языком программирования, когда служебные слова языка вводятся нажатием всего лишь одной клавиши как, к примеру, на ЭВМ "Искра 226". В последнее время появились клавиатуры, надписи на которых формируются жидкими кристаллами, что позволяет легко менять их при переходе на другой регистр или при работе с графическими символами.

Кроме того, на дисплеях выделяются специальные окна для указания функций программируемых клавиш. Особое удобство представляют дисплеи с контактными экранами, работать с которыми можно и без клавиатуры, нажимая на экране нужные места.

СОВЕТ 74

Тайные особенности аппаратных и программных средств

Будь любопытен.

Еще раз сформулируем совет, о котором мы уже векользь где-то говорили.
Беря в руки описание конкретной версии языка программирования, имей в виду, что такие руководства для пользователей, как правило, пишутся не разработчиками трансляторов, а совсем другими людьми. Авторы описаний языков программирования могут и не знать всех тонкостей и деталей конкретного транслятора, так как не имели на руках оригинального руководства для пользователя, а просто работали с транслятором.

Будь любопытен. Испытай на своей машине операторы и конструкции, не описанные в руководстве, но имеющиеся в других версиях языка.

Конкретный пример.

Мало где написано, что числовые переменные в последних версиях Бейсика по совместительству выполняют функции булевых (логических) переменных. Испытай свою Бейсик-машину странной на первый взгляд конструкцией $A = B = C$ ("по-паскалевски" это будет выглядеть так: $A := B = C$). Если ЭВМ не выдаст сообщение о синтаксической ошиб-

ке, то переменная А (в этом случае булева) примет нулевое значение, если $B \neq C$, и ненулевое (обычно минус единица), если $B = C$.

СОВЕТ 75

Не пользуйся советом 74

Авторы в книге нередко дают взаимоисключающие советы. Поступить так или иначе, а может быть совсем по-третьему — решать читателю.
Не используй предыдущий совет в обычной практике программирования. Побереги его для различных программных курсов или для воплощения правила о том, что программа без тайны — не программа.

СОВЕТ 76

Нужна ли ЭВМ педаль?

Нужна. И не только педаль, а другие устройства ввода сенсорной информации помимо клавиатуры.

На клавиатуре ЭВМ есть клавиша, нажатие которой приостанавливает бег строк на экране дисплея (клавиши пробела, BREAK, CTRL + S и др.).

Если ты, читатель, имеешь склонность модернизировать не только программы компьютера, но и сам компьютер, то можно посоветовать тебе приделаться к ЭВМ "педаль тормоза". Неплохо, если функция этой ножной клавиши клавиатуры будет перепрограммируемая: приостановка бега строчек на дисплее, переход на верхний регистр клавиатуры и др. Ведь так часто нам при работе с компьютером не хватает не только головы, но и рук. Они при отладке программы заняты карандашом, черновиками программ, распечатками результатов расчета и т. п.

Еще лучше, если ты, читатель, приделаешь к ЭВМ не педаль тормоза, а такие полезные вещи, как мыш для ввода графической информации, игровой пульт, цифровую мини-клавиатуру и др., руководствуясь принципом "Мы не будем ждать милостей от отечественной электронной промышленности, сделай их — наша задача".

СОВЕТ 77

Как включают периферию

На рис. 3.21 была помещена полупрограмма-полушутка поиска простых чисел, когда ЭВМ сама включает и выключает свою периферию.

Программа на рис. 3.21 нереальна не только потому, что нет в трансляторе соответствующих операторов, но и потому, что есть строгие правила включения периферии — "Круги на воде". Сначала включают

центральной и до начала работы с ней — всю нужную периферию. Выключают машину в обратном порядке — от периферии к центру. Включение периферии в момент работы на машине может приводить к сбоям системы, к ее "зависанию".

СОВЕТ 78

Бережного бог бережет

Засидевшись у компьютера, не поленитесь встать и повесить на рубильник, его питающий, табличку "Не выключать!", дабы уходящие домой аккуратные коллеги ненароком не обесточили тебя и не пустили насмарку твой дневной труд.

СОВЕТ 79

Полуфабрикаты на диске

Старайся, чтобы программа в ОЗУ и ее копия на внешнем носителе как можно меньше отличались друг от друга.

Расширяя и отлаживая программу, не забывай периодически сбрасывать полуфабрикаты на внешний носитель памяти. Они могут пригодиться по крайней мере в трех случаях:

- а) см. предыдущий совет;
- б) при зависании машин (они имеют такую дурную привычку); их приходится в этом случае отключать, включать и снова загружать "систему" (операционные системы, трансляторы языка);
- в) для возвращения к ранним вариантам программы, если в последних запутался.

Снабжай полуфабрикаты программ подробными комментариями, иначе в них тоже можно запутаться.

С другой стороны, не забывай периодически чистить рабочий магнитный диск от устаревших дублей программ.

Наиболее важные программы и файлы данных всегда имей как минимум в двух экземплярах на двух дисках (магнитных лентах).

Не полагайся на аккуратность пользователей, системные программисты разработали средства автоматической пересадки полуфабрикатов развиваемой и отлаживаемой программы.

СОВЕТ 7А

Бейсик-строки с одним комментарием

Не делай переходов к строкам с одной ремаркой и не выкидывай ремарки при вводе в ЭВМ чужой программы.

Не делай условных и безусловных переходов к строкам программ, содержащим только комментарии (ремарки). Позже ты сам или

твой товарищ-программист будут вводить эту программу в машину с листинга, "выжимая из нее воду", опуская комментарии и тем самым допуская ошибку перехода к несуществующему номеру строки. Большинство Бейсик-машин реагируют на такую ситуацию аварийным остановом при выполнении программы, но есть машины, позволяющие управление на строку с номером, ближайшим большим номером отсутствующей строки.

Сравни две программы: на рис. 2.9 и на рис. 2.13. В первой (рис. 2.9) — поиск оптимальной точки на карте) вызов подпрограмм на строках 220—250, 260—290 и 300—430 ведется ссылками на первую "действующую" строку. Во второй (рис. 2.13 — испытание "Вечного календаря") вызов подпрограмм на строках 100—160 и 170—260 ведется ссылками на строки с комментариями. Пропуск строк с комментариями при вводе программы на рис. 2.13 чреват ошибкой, а при вводе программы на рис. 2.9 — нет.

СОВЕТ 7В

Персональный компьютер и компьютер с разделением времени работы процессора

Есть единственный способ узнать, персональный ли у тебя компьютер или это терминал большой машины.

На персональном компьютере время выполнения конкретной программы — величина постоянная, а на компьютере с разделением времени работы процессора оно зависит от загрузки соседних терминалов (дисплеев).

СОВЕТ 7С

Имитация часов на компьютере

При замере интервалов времени иногда можно обойтись и без встроенных часов компьютера.

Далеко не все персональные компьютеры имеют встроенные часы, а необходимость в них ощущается довольно часто при написании, например, программ для машин, работающих в реальном режиме времени (см. программу "Электронный секретарь" на рис. 4.2).

Но встроенные часы компьютера можно смоделировать, запустив цикл, время выполнения одного такта которого известно).

Такой прием работы компьютера в реальном режиме времени реализован в программе на рис. 7.1, позволяющей определить среднюю скорость реакции человека за несколько измерений.

Подобную программу мы уже разбирали (см. рис. 5.16), но в ней мы использовали встроенные часы компьютера.

центральной и до начала работы с ней — всю нужную периферию. Выключают машину в обратном порядке — от периферии к центру. Включение периферии в момент работы на машине может приводить к сбоям системы, к ее "зависанию".

СОВЕТ 78

Береженого бог бережет

Засидевшись у компьютера, не поленись встать и повесить на рубильник, его питающий, табличку "Не выключать!", дабы уходящие домой аккуратные коллеги ненароком не обесточили тебя и не пустили насмарку твой дневной труд.

СОВЕТ 79

Полуфабрикаты на диске

Старайся, чтобы программа в ОЗУ и ее копия на внешнем носителе как можно меньше отличались друг от друга.

Расширяя и отлаживая программу, не забывай периодически сбрасывать полуфабрикаты на внешний носитель памяти. Они могут пригодиться по крайней мере в трех случаях:

- а) см. предыдущий совет;
- б) при зависании машин (они имеют такую дурную привычку); их приходится в этом случае отключать, включать и снова загружать "систему" (операционные системы, трансляторы языка);
- в) для возвращения к ранним вариантам программы, если в последних запутался.

Снабжай полуфабрикаты программ подробными комментариями, иначе в них тоже можно запутаться.

С другой стороны, не забывай периодически чистить рабочий магнитный диск от устаревших дублей программы.

Наиболее важные программы и файлы данных всегда имей как минимум в двух экземплярах на двух дисках (магнитных лентах).

Не полагайся на аккуратность пользователей, системные программы разработали средства автоматической перезаписи полуфабрикатов развиваемой и отлаживаемой программы.

СОВЕТ 7А

Бейсик-строки с одним комментарием

Не делай переходов к строкам с одной ремаркой и не выкидывай ремарки при вводе в ЭВМ чужой программы.

Не делай условных и безусловных переходов к строкам программы, содержащим только комментарии (ремарки). Позже ты сам или

той товарищ-программист будут вводить эту программу в машину с листинга, "выжимая из нее воду", опуская комментарии и тем самым допуская ошибку перехода к несуществующему номеру строки. Большинство Бейсик-машин реагируют на такую ситуацию аварийным остановом при выполнении программы, но есть машины, позволяющие управление на строку с номером, ближайшим большим номером отсутствующей строки.

Сравни две программы: на рис. 2.9 и на рис. 2.13. В первой (рис. 2.9 — поиск оптимальной точки на карте) вызов подпрограмм на строках 220—250, 260—290 и 300—430 ведется ссылками на первую "действующую" строку. Во второй (рис. 2.13 — испытание "Вечного календаря") вызов подпрограмм на строках 100—160 и 170—260 ведется ссылками на строки с комментариями. Пропуск строк с комментариями при вводе программы на рис. 2.13 чреват ошибкой, а при вводе программы на рис. 2.9 — нет.

СОВЕТ 7В

Персональный компьютер и компьютер с разделением времени работы процессора

Есть единственный способ узнать, персональный ли у тебя компьютер или это терминал большой машины.

На персональном компьютере время выполнения конкретной программы — величина постоянная, а на компьютере с разделением времени работы процессора оно зависит от загрузки соседних терминалов (диспеев).

СОВЕТ 7С

Имитация часов на компьютере

При замере интервалов времени иногда можно обойтись и без встроенных часов компьютера.

Далеко не все персональные компьютеры имеют встроенные часы, а необходимость в них ощущается довольно часто при написании, например, программ для машин, работающих в реальном режиме времени (см. программу "Электронный секретарь" на рис. 4.2).

Но встроенные часы компьютера можно смоделировать, запустив цикл, время выполнения одного такта которого известно).

Такой прием работы компьютера в реальном режиме времени реализован в программе на рис. 7.1, позволяющей определить среднюю скорость реакции человека за несколько измерений.

Подобную программу мы уже разбирали (см. рис. 5.16), но в ней мы использовали встроенные часы компьютера.

```

10 INPUT "ЧИСЛО ПОПЫТОК"; N; S=0
20 FOR I=1 TO N
30 ? "ГОТОВЫ?"
40 GET A: IF A=0 GOTO 40
50 J=INT(300*RND(1)): J=0
60 J=J+1: IF J=J0 GOTO 90
70 GET A: IF A=0 GOTO 60
80 J=1: ? "ШТРАФ! "; GOTO 120
90 ? "ПУСК!": J=0
100 J=J+1: GET A: IF A=0 GOTO 100
110 J=0.008*J
120 ? "РЕАКЦИЯ"; J; " СЕК": S=S+J
130 NEXT I СЛЕДУЮЩАЯ ПОПЫТКА
140 ? "СРЕДНЯЯ РЕАКЦИЯ-"; S/N; " СЕК "

```

Рис. 7.1. Бейсик-программа проверки скорости реакции человека: имитация таймера

Как проводится замер скорости реакции по программе на рис. 7.1? Человек поудобнее усаживается у компьютера и в ответ на запрос машины "Готовы?" (строка 30) нажимает любую цифровую клавишу. После этого через некоторый интервал времени (он разный в различных замерах) на дисплее появится сообщение "Пуск!" (строка 90). Тут человек должен как можно быстрее (скорость реакции!) нажать любую клавишу. Если же он ее нажмет раньше появления команды "Пуск!" (не выдержали нервы? попытки перехитрить машину?), то ЭВМ накажет его секундным штрафом (строка 80).

Оператор GET существенно обогащает возможности диалога человека с компьютером. Работая по программе с оператором GET, компьютер может делать два дела одновременно — вести счет и как бы прислушиваться к человеку, передавая управление программой на новые участки, как только тот нажмет оговоренную или любую клавишу клавиатуры. Так и вежливый, внимательный человек, ведя свой монолог, прислушивается к собеседнику, давая ему возможность при первой необходимости вставить свою фразу или задать уточняющий вопрос.

Эрац-часы, встроенные в программу, требуют тарировки. Число 0,008 на строке 110 программы говорит о том, что один такт цикла "до" на строке 100 выполняется машиной за 0,008 с.

Не пытайся смоделировать встроенные часы на компьютере, работающем в режиме разделения времени процессора между многими пользователями, — вот и пригодился тебе предыдущий, казалось бы, ни на что не нужный совет.

Непрономерованный совет. Не отмахивайся от советов книги, даже если они тебе с первого раза покажутся непрактичными и даже вредными. Вдруг они на что-нибудь пригодятся! Даже на то, чтобы поискать более оптимальное решение.

Два способа тасовки карт

Очень много игр, в которые играет человек, начинаются с бросания жребия: перемешивания в мешке фишек лото, перемешивания на столе костяшек домино, бросания игральные костей, тасовки карт. Программирование подобных игр начинается с обучения компьютера тасовке карт, например для генерации целых неповторяющихся случайных чисел в заданном диапазоне.

Имея под рукой генератор псевдослучайных чисел и функцию выделения целой части числа, несложно получить ряд целых случайных чисел. Стожнее добиться, чтобы эти числа не повторялись.

Программы на рис. 7.2 и 7.3 позволяют получать такие ряды чисел, но делают это по разным алгоритмам.

По программе на рис. 7.2 создается массив A, вещественный по форме, но булевый по содержанию. Его элементы хранят информацию о том, выходило ли число, совпадающее с номером элемента массива, или нет. Если, например, $A(5) = 0$, то это значит, что числа пять еще не было. Если же $A(9)$ не равно нулю, то девятка уже выпала. Поэтому в теле цикла с параметром на строке 2 записан цикл "до" — генерация целого случайного числа повторяется до тех пор, пока машина не убедится, что такого числа еще не было.

Алгоритм генерации целых случайных неповторяющихся чисел от 1 до N по программе на рис. 7.3 иной. Сначала машина заполняет двумерный массив A следующим образом. Элементы первого столбца принимают значения номера строки I, а элементы второго столбца — значения случайных вещественных чисел в интервале от 0 до 1 (строка 7). Затем массив A сортируется в порядке убывания значений элементов второго столбца (строки 8–12). Это влечет за собой переименование значений элементов первого столбца — целых чисел в диапазоне от 1 до N.

В программе на рис. 7.3 реализованы три способа инициализации генератора случайных чисел:

использование в качестве базы генератора псевдослучайных чисел показаний встроенного секундомера ЭВМ (строка 1);

```

INPUT "КОЛ-ВО ЧИСЕЛ"; N
DIM A(1 TO N) AS INTEGER
FOR I=1 TO N
DO: B=1+INT(N*RND)
LOOP WHILE A(B)=1
A(B)=I: PRINT I; B
NEXT

```

Рис. 7.2. Бейсик-программа тасовки карт: использование булевого массива

```

1 P=TIMER: RANDOMIZE P
2 ? "PRESS ANY KEY TO CONTINUE - НАЖМИ ЛИБУЮ КЛАВИШУ"
3 DO: A=RND: K=INKEY: LOOP WHILE K=""
4 P=VAL(RIGHT$(TIMER,2)): FOR I=-1 TO P: A=RND: NEXT
5 CLEAR P 1,4: СТИРАНИЕ НАЧАЛА ПРОГРАММЫ
6 INPUT "КОЛ-ВО ЧИСЕЛ": N: DIM A(1 TO N, 1 TO 2)
7 FOR I=1 TO N: A(I,1)=I: A(I,2)=RND: NEXT
8 DO: PR=0
9 FOR I=2 TO N
10 IF A(I-1,2)>A(I,2) THEN
    FOR J=1 TO 2: SWAP A(I-1,J),A(I,J): NEXT: PR=1
    END IF
11 NEXT
12 LOOP UNTIL PR=0
13 FOR I=1 TO N: ? I A(I,1): NEXT

```

Рис. 7.3. Бейсик-программа тасовки карт: использование отсортированного двумерного массива

холостая прокрутка генератора до тех пор, пока человек не нажмет какую-либо клавишу клавиатуры (строки 2 и 3);

холостая прокрутка генератора случайное число раз, зависящее от показания встроенных часов компьютера (строка 4).

Непрономерованный совет. Оформлйй сообщения машины двуязычно (строка 2 в программе на рис. 7.3). Это, во-первых, будет приучать отечественного пользователя к работе с импортными программными продуктами, а во-вторых, повысит шансы программы перешагнуть государственные границы.

Если нет уверенности, что резервируемый массив (строка 6 на рис. 7.3) уместится в памяти машины, то "отработанную" часть программы можно стереть (строка 5).

Программа на рис. 7.2 работает по алгоритму, похожему на вытаскивание карт из середины колоды. Согласно программе на рис. 7.3 колода карт сначала тасуется, а затем с нее слистаются отдельные карты.

Программа на рис. 7.2 проще программы на рис. 7.3, но имеет один существенный недостаток — у нее переменная (убывающая) скорость генерации целых случайных неповторяющихся чисел.

СОВЕТ 7Е

Как принимают решения с помощью ЭВМ

Очень часто, решая ту или иную проблему, мы оказываемся в шкуре бурдианова осла. Задача может иметь два альтернативных решения, как две охапки сена слева и справа от упомянутого животного. Дово-

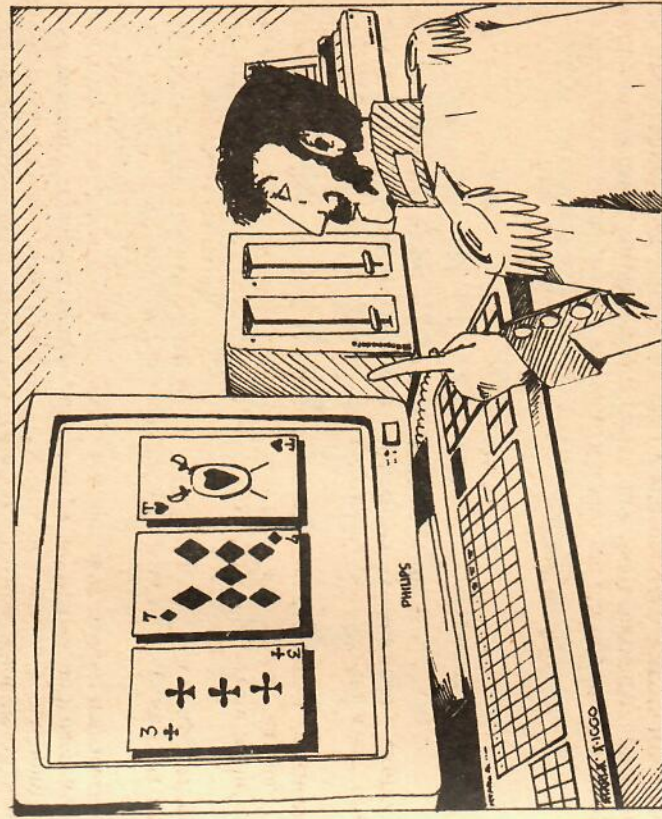
ды "за" и "против" принятия каждого уравновешены. Как в этом случае поступить? Ехать или не ехать в командировку? Строить или не строить запроецированный завод? Понстине, гамлетовские вопросы задают нам жизнь!

Некоторые в таких ситуациях бросают монету, другие загадывают "Мужчина или женщина?" и смотрят в окно, ожидая, кто первый появится, а по улице, как назло, за целый час кошка только и пробежит... Есть проверенный столетиями метод принятия подобных решений. Достаточно разложить пасьянс. Сошелся — решение принято, все сомнения прочь. Можно еще подыскать доводы в его пользу, и оно будет твердо воплощаться в жизнь. Пасьянс психологически нас на это настраивает.

Но принять решение подобным образом бывает иногда трудно, так как не всегда под рукой есть колода карт, да и не совсем удобно раскладывать их на рабочем месте. Но это можно сделать и на экране дисплея ЭВМ.

Программа на рис. 7.4 позволяет разложить старинный пасьянс "Турецкий шпак" по таким правилам.

Из одной колоды карт в 52 листа выкладывают картинкой вверх пять рядов по 10 карт в каждом (рис. 7.5). Последние две карты кла-



```

H=0;M=4;K=13;DIM M(K),K(K),B(5,9),N(9),K(M*K),J(1)
DATA 50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100
CLS:PRINT"ТУРЕЦКИЙ ПЛАТОК.НАИМИЙТЕ НОМЕРА СТОЛБЦОВ.S-S STOP"
FOR I=1 TO K:READ A:K(I)=CHR$(A):NEXT I:СЧИТЫВАНИЕ КАРТ
FOR I=1 TO M:READ A:M(I)=CHR$(A):NEXT I:СЧИТЫВАНИЕ МАСТИ
FOR J=0 TO 9:ПЕЧАТАТЬ НОМЕРА СТОЛБЦОВ КАРТ
PRINT " ";J;:IF J<=1 THEN N(J)=5 ELSE N(J)=4
NEXT:PRINT:PRINT
FOR I=0 TO 5:FOR J=0 TO 9:РАСКЛАДКА ПАСЬЯНСА
DO:V=1+INT(M*K*RND(1)):LOOP UNTIL K(V)=0
K(V)=1:N=INT(V-K*INT((V-1)/K)):B(I,J)=K(N)
PRINT " ";K(N);M(I+INT((V-1)/K)):;IF I=5 AND J=1 THEN J=9
NEXT:PRINT:NEXT:ПАСЬЯНС РАЗЛОЖЕН,ДАЛЕЕ РАСПУСКАЕМ ПЛАТОК
DO:DO:PLAY "BC":ОЧЕРЕДНОЙ ХОД
FOR I=0 TO 1:СНЯТИЕ ДВУХ КАРТ (ПОПЫТКА)
DO:DO:AP=INKEY$:LOOP UNTIL AP="S" OR AP="0" AND AP<="9"
J(I)=VAL(AP):IF AP="S" THEN PLAY "CDEFGAB":END
LOOP WHILE N(J(I))=-1
NEXT:ДАЛЕЕ ПРОВЕРКА,ОДНОГО ЛИ ДОСТОИНСТВА СНИМАЕМЫЕ КАРТЫ
LOOP WHILE J(0)=J(1) OR B(N(J(0)),J(0))<>B(N(J(1)),J(1))
FOR I=0 TO 1:СНИМАЮТСЯ ДВЕ КАРТЫ
LOCATE N(J(I))+4,4+J(I)+2:PRINT " ";N(J(I))=N(J(I))-1
NEXT
H=N+2:LOCATE 10,10:PRINT USING"СНЯТО # КАРТ";H:PLAY"CD00"
LOOP UNTIL H=52:ДАЛЕЕ ТУРЕЦКОЕ РОНДО В.МОЦАРТА
PLAY"120 ML L16 02 BAG#A>C4 DC<B>CE4""ТАРА-ПАРА-ПААМ - 2-РАЗА
PLAY"03 FED#E#BAG#A#>C4 L8 AB>C<BAG#AE#FDC<B4A4"
PLAY"03 AB>C#4<AB>C#<BAG#F#6#ABG#EAB>C#4<AB>C#<BAG#F#6#E#A.P16"
PLAY"03 AB>C#4<AB>C#<BAG#F#6#ABG#EAB>C#4<AB>C#<BAG#F#6#E#A4"

```

Рис. 7.4. Бейсик-программа раскладки пасьянса "Турецкий платок"

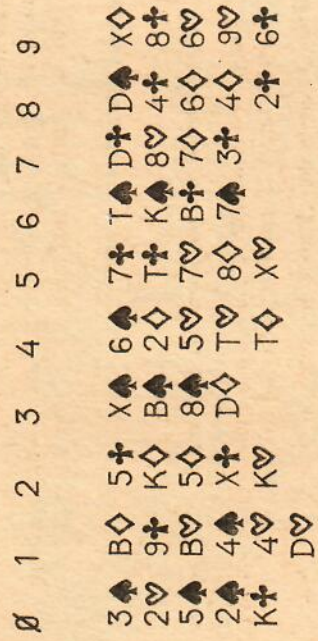


Рис. 7.5. Образец раскладки пасьянса "Турецкий платок": снято две карты

дут в шестой неполный ряд на любое место, как правило, к первому (у нас он будет нулевой) и второму (первому) столбцам. Требуется распустить этот турецкий платок, снимая за один ход по две нижние карты одного достоинства (тройки, дамы и т. д.) из разных столбцов.

На рис. 7.5 показана раскладка пасьянса после снятия двух пар карт. По правилам игры сейчас можно снять двух королей либо двух дам. Для этого нужно нажать на клавиатуре две клавиши в любом порядке — 0 и 2 или 1 и 3, что соответствует номерам столбцов, которые эти карты замыкают. Изображения снимаемых карт на экране сотрутся, игру можно продолжить — откроются двойки.

Если удастся снять все 52 карты, то машина в награду сыграет нацело "Турецкое рондо" Моцарта. Это означает, что в командировку все-таки ехать нужно, а завод построить стоит.

Если же при игре станет ясно, что пасьянс разобрать невозможно, то остановить игру поможет клавиша "S" (STOP).

В таком случае играющий услышит лишь музыкальную гамму. С принятием положительных решений по поставленным проблемам пока нужно повременить.

Последний пронумерованный совет. Имей в матобеспечении компьютера игры, головоломки, пасьянсы и прочее и, работая над сложной программой, периодически отвлекайся, чередуя дело с потехой.

Кстати, раскладка пасьянса, показанная на рис. 7.5, сходится. Как легко это понять человеку и как трудно написать программу для ЭВМ, "понимающую" это.

Последнее задание читателю. Составь программу — перевертыш программы на рис. 7.4: человек сообщает машине начальную конфигурацию карт в раскладке пасьянса "Турецкий платок", а машина отвечает, раскладывается он или нет.

Составление этой программы будет одним из свидетельств того, что ты, читатель, теряешь статус начинающего программиста.

СОВЕТ 7F

Timeo Danaos et dona ferentes

Суть совета сформулирована устами жреца Лаокоона, заклинявшего троянцев не втаскивать в город деревянного коня:

"Либо ахейцы за досками этими скрыты,
Либо враги возвели громаду эту, чтобы нашим
Стенам грозить, дома наблюдать и в город проникнуть.
Тевкры, не верьте коню: обман в нем некий таится.
Чем бы он ни был, страшусь и дары приносящих данайцев."

(Вергилий, "Энеида", пер. С. Ошерова)

Большинство программ, иллюстрирующих советы книги, написаны на Бейсике. В последнее время критика в адрес этого самого популярного языка программирования сильно изменилась. Прежде его ругали за неструктурность, за необязательность и неряшливость в описании переменных, за медленность работы интерпретаторов и, наконец, за то, что он прививает начинающим плохой стиль программирования, и исправит который потом практически невозможно. Сейчас же люди, знакомые с последними версиями этого языка (Турбо- и Квик-Бейсиком), если и ругают его, то за то, что он беззастенчиво "крадет" у других языков (у Паскаля, Ады, Модулы, Си и др.) их полезные качества. Так, программы, написанные на Квик-Бейсике, мало чем отличаются от Паскаль-программ. Но, приближаясь к Паскалю, Квик-Бейсик оставил такое "родимое пятно", как необязательность описания типов переменных: в Бейсик-программу можно "нырять" сразу, переложив на плечи машины заботу об определении статусов переменных и об их начальных нулевых значениях.

Приверженцы Бейсика считают такую особенность языка достоинством, а приверженцы Паскаля — недостатком. Но скорее всего правы и те, и другие, если вспомнить, что наши недостатки — это очень часто продолжение наших достоинств. При написании коротких программ (а они преобладают в инженерной практике) описание переменных часто бывает лишним и только раздражает программистов-дилетантов, составляющих основной круг пользователей Бейсика. Но отсутствие описания переменных чревато грубыми ошибками. На рис. 7.6 помещена программа "Вечный календарь" — аналог программы на рис. 0.2, но переписанной на Квик-Бейсике. Мало кто заметит ошибку в конце строки 7, где в имени переменной, хранящей номер дня, буква А заме-

```

1 *ОПРЕДЕЛЕНИЕ ДНЯ НЕДЕЛИ ПО DATE
2 INPUT "Г,ММ,ДД ";YEAR,MONTH,DAY
3 MONTH=MONTH+1
4 IF MONTH<4 THEN
5     MONTH=MONTH+12:YEAR=YEAR-1
6     END IF
7 DAY=INT(365.25*YEAR)+INT(30.6001*MONTH)+DAY
8 SELECT CASE DAY-7*INT(DAY/7)
CASE 0:PRINT "ВОСКРЕСЕНЬЕ"
CASE 1:PRINT "ПОНЕДЕЛЬНИК"
CASE 2:PRINT "ВТОРНИК"
CASE 3:PRINT "СРЕДА"
CASE 4:PRINT "ЧЕТВЕРГ"
CASE 5:PRINT "ПЯТНИЦА"
CASE 6:PRINT "СУББОТА"
16 END SELECT

```

Рис. 7.6. Бейсик-программа "Вечный календарь"

нена на букву Е, что, кстати, более соответствует английской транскрипции слова "день": дей, а не дай. Эту ошибку пропустит и компилятор Квик-Бейсика, сочтя по принципу умолчания DEU за новую переменную и присвоив ей нулевое значение. Паскаль же такую ошибку не пропустит, выдав сообщение — "ссылка на переменную, не объявленную в заголовке программы".

Квик-Бейсик, руководствуясь принципом умолчания, может окрестить и другие "медвежьи услуги". Так, например, при работе по программе "Нелинейная аппроксимация" (см. рис. 0.14) может возникнуть ситуация возведения нуля в нулевую степень, когда один из аргументов исходной табличной зависимости равен нулю. В такой ситуации Бейсик "Искры 226" выдаст сообщение о некорректной операции, Квик-Бейсик же по принципу умолчания решит, что неопределенность 0° стремится к единице. В программе на рис. 0.14 это так и есть на самом деле, но в других ситуациях такая услужливость чревата ошибками.

ПОСЛЕСЛОВИЕ

Его мы даем в духе книги — в виде ряда взаимоисключающих советов, взятых из различных книг по программированию.

Помни! Практически невозможно научиться хорошо программировать студентов, ориентированных первоначально на Бейсик: как потенциальные программисты они умственно оболванены без надежды на исправление.

Не спешите навешивать ярлыки на языки программирования, тем более на живые, новые диалекты которых появляются чуть ли не каждый месяц. Прежде чем выслушать критику в адрес какого-либо языка, спроси у собеседника, не имеет ли он в виду версию двадцатилетней давности.

Осторожно! Занятие программированием может лишить тебя будущего. Не думай, что, научившись программировать на Бейсике, ты чего-то добьешься в жизни.

Программирование — вполне уважаемая профессия, а чтение популярных книг по программированию и решение программистских задач ничем не опаснее чтения популярных книг по математике и решения математических задач.

Если ты хочешь получить от программирования пользу, не программируй!

Даже если ты большой начальник или собираешься им стать, знакомство с искусством программирования не будет лишним. Оно поможет тебе, во-первых, более четко и квалифицированно формулировать задания подчиненным на разработку или поиск программных продуктов, а во-вторых, заставит тебя более справедливо оценивать их труд.