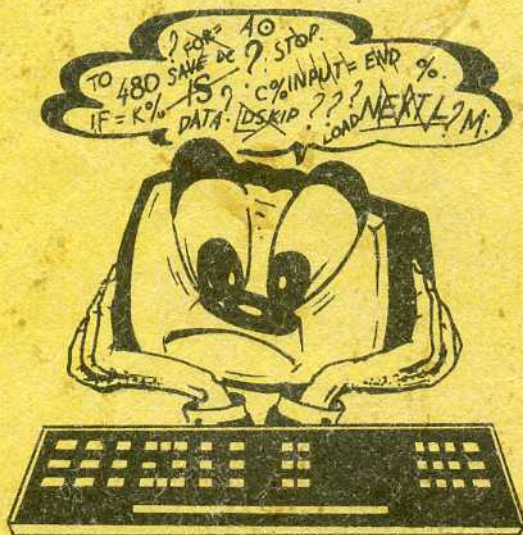


СПРАВОЧНОЕ ПОСОБИЕ ПО ЯЗЫКУ ПРОГРАММИРОВАНИЯ БЕЙСИК



Справочное пособие
по языку программирования
«БЕЙСИК»

(Бейсик — шпаргалка)

Ленинград
«Наука и быт»
1991 г.

Редактор *Е. В. Ильина*
Оформление художника *О. Коловенко*

Количество программ на самом популярном языке программирования Бейсик, публикуемых в различных источниках и ежегодно входящих в широкую практику, исчисляется в астрономических числах. Рядовому пользователю трудно оперативно разобраться не только в нюансах предлагаемого алгоритма, но и в особенностях его реализации, связанной со спецификой используемой операционной системы.

Мы предлагаем справочное пособие, которое должно стать для Вас ключом к быстрому пониманию выбранной Вами версии языка Бейсик, методическим руководством по адаптации чужой программы к своим специфическим нуждам. Справочное пособие незаменимо при переложении текста программы с одной версии языка Бейсик на другую.

В настоящее время известно более 30 версий языка Бейсик. Наибольшее распространение получили 4 версии, представляемые Вашему вниманию:

- 1 — Бейсик-ЕС
- 2 — PDP-11 (CM-3, CM-4)
- 3 — BASIC-A (IBM PC)
- 4 — BASIC MSX

В пособии сопоставляются версии Бейсика по трем разделам: операторы, функции и директивы. В описании их форматов приняты следующие обозначения:

- [] — необязательный элемент формата,
- { } — тут надо выбрать один элемент формата из предложенных,
- < > — название элемента формата,
- ... — повтор последнего элемента формата,
- x — арифметическая переменная; абсцисса,
- y — ордината,
- b — символьная, строковая переменная,
- n — номер оператора, строки,
- a или a() — массив или матрица, в скобки вписывается размерность,
- i — строка матрицы,
- j — столбец матрицы,
- k — параметр, коэффициент.

Формат оператора	Описание оператора	Номер версии
BEEP	Звуковой сигнал в 800 Гц и 0,25 с	3 4
CALL <имя процедуры> [(k1[,k2,...])] <i>вызывает</i>	Вызов процедуры	1 2 3 4
CHAIN [MERGE] «<имя>» [, <точка входа>] [, ALL] [, DELETE <№ нач. оператора> — <№ кон. оператора>]	Уничтожение фрагмента программы	2 3
CIRCLE (x,y), <радиус> [, <код цвета> [, <1 угол>, <2 угол> [, k]]] <i>круг</i>	Построение дуги окружности или эллипса	3 4
CLOSE <имя файла>	Закрытие доступа к файлу	1 2 3 4
CLS <i>зачистка</i>	Очистка экрана с возвратом курсора в левый верхний угол	3 4
COLOR <фон>, <палитра>	Выбор цвета фона и палитры	3 4
COM x,a(),b	Занесение данного в общую память	3
COMMON x,a(),b	Занесение данного в общую память	2 3
COPY <имя вставки>	Ввод в программу библиотечной вставки — фрагмента исходной программы с относительной нумерацией строк	1
DATA [(k1)]p1[, [(k2)]p2...] <i>данные</i>	Занесение данных с коэффициентом кратности k	1 2 3 4
DEF FN <имя функции> (x) = <выражение>	Описание формулы нестандартной функции	1 2 3 4
DEF DBL x1[-x2][, x3[-x4]...] <i>определен</i>	Объявление вещественного типа с удвоенной точностью	3 4
DEF INT x1[-x2][, x3[-x4]...]	Объявление целого типа	3 4
DEF SEG = <адрес>	Занесение адреса начала текущей программной секции	3
DEF SGN x1[-x2][, x3[-x4]...]	Объявление вещественного типа	3 4
DEF STR x1[-x2][, x3[-x4]...]	Объявление символьного типа	3 4
DEF USRn = <адрес>	Объявление точки входа в подпрограмму n	3 4
DIM x1(n1[,n2,...]) [, x2(n3[,n4,...])]	Распределение памяти для массивов	1 2 3 4
DIM # <номер>, b() = <адрес>	Размещение виртуального массива в дисковой памяти	2
DRAW b(), x,y	Задание вектора из текущей точки	3 4
END	Завершение программы	1 2 3 4
ERASE <файл1> [, <файл2> ...]	Очистить файл	3 4
ERROR <код ошибки>	Формирование кода ошибки	1 2 3 4
FIELD [#] <номер файла>, <длина1> AS b1, <длина2> AS b2 <i>поле</i>	Определение полей файла с прямым доступом	3
FOR <счетчик циклов> = <нач. значение> TO <кон. значение>	Начало цикла FOR...NEXT	1 2 3 4
STEP <приращение>		
GET <имя файла>, <список переменных>	Чтение записи из файла	1 3
GET <номер файла>, <номер записи>	Чтение записи из файла	3 4
GOSUB <номер строки> [(x1[,x2,...])]	Передача управления в строку подпрограммы	1 2 3 4
GO TO <номер строки>	Передача управления в строку программы	1 2 3 4
IF <условие> GOSUB <номер строки>	Условный переход к подпрограмме	1

IF <условие> GO TO <номер строки>	Условный переход к строке программы	1 3 4
IF <условие> NEXT <управляющая переменная>	Условное окончание цикла	1
IF <условие> THEN <оператор>	Условное выполнение оператора	1 2 3 4
IF <условие> THEN <оператор> ELSE <оператор>	Условное выполнение оператора	3 4
INDATA <имя>,[<№ нач. оператора>],[<длина фрагмента>],[<указатель файла>]	Сборка архивных документов	1
INPUT x1[,x2,...]	Ввод переменных с терминала	1 2 3 4
INPUT #n[,x1,...]	Ввод числовых и текстовых значений	2 3 4
INTERVAL {ON OFF STOP}	Сигнал прерывания {сработает не состоится заморожен}	3
KEY n,b	Задание значения функциональной клавиши	3 4
KEY ON	Вывод 6 символов определения функциональной клавиши	3 4
KEY OFF	Стирание с дисплея	3 4
KEY LIST	Вывод определения функциональных клавиш полностью	3 4
KEY <значение> {ON OFF STOP}	Слежение за функциональной клавишей {разрешается запрещается приостанавливается}	3 4
KEY (<№ функц.клавиши>) b	Активизация функциональной клавиши с заданным номером от 1 до 10 как программируемой с указанным строковым значением	3 4
KILL <имя файла> <i>Subect</i>	Удаление файла с диска	3 4
[LET] x=[<выражение>] <i>ya bal</i>	Присвоение значения выражения заданной переменной	1 2 3 4
LINE [(x1,y1)-(x2,y2)][,<цвет>][,BLF]] <i>линия</i>	Формирование отрезка прямой или прямоугольника на дисплее, где BF — закрашен весь прямоугольник, B — только контур	3 4
LINE INPUT b1, b2	Ввод данных по запросу ЭВМ	3 4
LINE INPUT n,b1	—“—	3 4
LINPUT b1, b2	—“—	2
LINPUT #n,b	—“—	2
LOCATE [i,j][,{0 1}][,i _n ,i _k]	Установка курсора в позицию (i,j) и ограничение видимости курсора i _n — i _k	3 4
LPRINT [USING <шаблон>]; <список значений>[:]	Печать на LPT1:, аналогично оператору PRINT	3 4
LSET <поле> = b	Перемещение в буфер с прижимом влево	3
MAT INPUT x1[,x2,...]	Ввод значений элементов массива с терминала	1
MAT PRINT x1[,x2,...]	Печать элементов массива	1
MAT READ x1[,x2,...]	Присвоение элементам массива значений из блока данных	1
MAT a=CON	Роспись массива единицами	1
MAT a=IDN	Формирование единичной матрицы	1
MAT a1=INV(a2)	Обращение матрицы	1
MAT a1=TRN(a2)	Транспонирование матрицы	1
MAT a=ZER	Очистка массива (роспись нулями)	1
MAT a1=a2	Матричное присваивание	1

MAT a1=a2+a3	Сложение матриц (векторов)	1
MAT a1=a2-a3	Вычитание матриц (векторов)	1
MAT a1=a2*a3	Умножение матриц	1
MAT a1=(k)*a2	Умножение матрицы на скаляр	1
MAXFILES = < количество файлов >	Ограничение количества одновременно открытых файлов в пределах данной задачи	4
MOTOR { 0 <число> ≠ 0 }	Включение — выключение двигателя	3 4
NEXT x	Конец циклического участка с передачей управления переменной	1 2 3 4
ON ERROR GO TO <метка>	Переход на оператор условия в ситуации ошибки	3 4
ON <выражение> GOSUB n1,n2...	Переключатель подпрограмм	2 3 4
ON <выражение> GO TO n1,n2...	Переключатель строк	1 2 3 4
ON INTERVAL = <нач.значение> GOSUB n	Занесение начального значения и пуск таймера	4
ON KEY GOSUB n	Обработка прерываний	3 4
ON PEN GOSUB n	Привязка точки входа в подпрограмму обработки прерываний	3
ON SPRITE GOSUB <точка входа>	Указание точки входа для обработки прерываний	4
ON STRIG GOSUB n	Указание точки входа для изменения цвета	3 4
OPEN { <имя файла> <устройство> [FOR <режим>] AS [#] { <номер файла> <номер устройства> } LEN = <длина>	Присвоение файлу или устройству с заданным именем указанного номера и задание режима доступа к нему	2 3 4
OPEN <режим> [#] { <номер файла> <номер устройства> <имя файла> <устройство> [, <длина записи>]	Присвоение имени указанному физическому файлу или устройству в одном из режимов: =0 организуется последовательный вывод, =1 — последовательный ввод, =R — произвольный доступ	1 2 3 4
OPEN'COM <адаптер>: <протокол>'AS [#]<номер устройства> [LEN = <длина буфера>]	Присвоение заданного номера файла или устройства адаптеру	2 3 4
OPTION BASE { 0 1 }	Фиксация начала отсчета индексов (базы) в массивах от 0 или от 1	3
OUT x,y	Запись однобайтного числа у в порт вывода с адресом x	3 4
OUTDATA <имя>[, <№ нач. оператора>],[<длина>],[<указатель временного файла>]	Сохранение временного файла в архиве	1
OVERLAY '<имя>', n	Дозагрузка программы с заданным именем	2
PAINT (i,j) [<цвет-заполнитель>[, <цвет-ограничитель>]]	Закрашивание цветом-заполнителем области экрана, начиная с заданной точки, кроме точек, закрашенных цветом-ограничителем	3 4
PEN OFF	Игнорирование сигналов светового пера	3
PEN ON	Разрешение на переход	3
PEN STOP	Временная задержка обработки прерываний	3
POKE <адрес>,x	Занесение значения аргумента по адресу	3 4

PRESENT (x,y), <номер цвета>	Занесение точки в буферную память дисплея	3 4
PRINT <элемент1> [<разделитель1> <элемент2> ...]	Вывод печати элементов списка на дисплей	1 2 3 4
PRINT USING <формат>, <список>	Вывод данных на терминал	2 3 4
PRINT #n, <список>	Вывод списка на указанное устройство	2 3 4
PRINT #n USING <формат>, <список>	Вывод данных на указанную МЛ или МД	3 4
PSET (x,y), <номер цвета>	Занесение точки в буферную память	3 4
PUT (x,y), b[, <режим>]	Отображение информации на дисплей в заданном режиме	1
PUT # <номер файла> [, <номер записи>]	Вывод записи на диск	3
PUT SPRITE <маска> (x,y), <темный цвет>, <элемент>	Перенос закодированного изображения из элемента массива на маску с заданным темным цветом в точку привязки	4
RANDOMIZE	Перенастройка датчика случайных чисел	2 3
READ x1[, x2...]	Выборка очередного значения из блока данных и присвоение его очередной переменной из данного списка	1 2 3 4
REM <произвольный текст>	Комментарий	1 2 3 4
RESET [<выражение>], [x]	Переустановка указателя	1
RESTORE	Возврат указателя блока данных в исходное состояние	1 2 3 4
RESTORE n	Возврат указателя в строку под данным номером	1
RESTORE #	Вернуть указатель числового блока данных в исходное положение	2
RESUME [{n NEXT}]	Продолжение выполнения программы после обработки оператором ON ERROR GO TO с заданного места	3 4
RETURN	Возврат из подпрограммы	1 2 3 4
RSET <поле> = b	Перемещение данных в буфер с прижимом вправо	3
SCREEN [<режим отображения>] [, <режим цветности>] [, <№ активной страницы>] [, <№ видимой страницы>]]]	Установка режимов отображения и цветности, фиксация номеров видимой и активной страниц	3 4
SOUND <w>, <t>	Регулировка частоты сигнала 37Гц <w> <32767Гц и длительности 0 <t> <65535 маш.единиц	3 4
SPRITE {ON OFF STOP}	{Разрешение Блокировка Замораживание} сигнала прерывания	4
STOP	Останов программы, после которого ее можно продолжить либо со следующего оператора, либо с указанного	1 2 3 4
STRIG (n) {ON OFF STOP}	{Разрешить Запретить Заморозить} событие № 0 ≤ n ≤ 4	3 4
SWAP x1, x2	Обмен значениями между переменными	3 4
TROFF	Отключение трассировки	3 4

TRON	Включение трассировки	3 4
WAIT x,y,z	Ожидание события, при совершении которого в байте с адресом x окажется кодовая комбинация y&z	3 4
WEND	Последний оператор тела цикла, осуществляющий проверку заданного условия	3 4
WHILE <выражение>	Первый оператор тела цикла, осуществляющий проверку заданного условия	3 4
WIDTH [<устройство>,<ширина>	Установка длины выводной строки на дисплее или указанном устройстве	3
WRITE [#<номер файла>,<список значений>	Вывод на экран списка значений или в указанный номер файла	1 2 3 4
?	Вместо слова <PRINT>	3 4
,	Вместо слова <REM>	3 4

Формат функции	Описание функции	Номер версии
ABORT(I)	Прекращение выполнения программы с сохранением ее текста в памяти I=0 или со стиранием I=1	2
ABS(x)	Модуль аргумента, x	1 2 3 4
ASC(b)	Машинный формат числа, представленный в символьном виде текстовым выражением	2 3 4
ATN(x)	$\arctg x$	1 2 3 4
BIN(b())	Преобразование целого числа от 0 до 255 в восьмиразрядный двоичный код	2
BIN\$(I)	Преобразование восьмиразрядного двоичного кода в целое число	4
CDBL(x)	Преобразование заданного числа в число с удвоенной точностью	3 4
CHR\$(I)	Символ ASCII, которому соответствует значение I	2 3 4
CINT(x)	Целое число, полученное округлением вещественного числа с обычной точностью	3 4
COS(x)	$\cos x$	
CSNG(x)	Вещественное число с обычной точностью, полученной округлением вещественного числа с двойной точностью	2
CVD (<8-символьное текстовое значение>)	Машинный эквивалент вещественного числа с двойной точностью, представленного 8-символьным текстовым значением	3
CVI (<двусимвольное текстовое значение>)	Машинный эквивалент целого числа, представленного двусимвольным текстовым значением	3
CVS (<4-символьное текстовое значение>)	Машинный эквивалент вещественного числа с обычной точностью, представленного 4-символьным текстовым значением	3
DATE\$(0)	Текущая дата в виде символьного значения «ДД-ММ-ГГ»	3

DATE\$(n)	Текущая дата дня с номером относительно 01.01.1970	3
EOF(n)	= 1, если вместо очередной n записи в файле считан признак конца файла	3 4
EXP(x)	e^x	1 2 3 4
FIX(x)	Выделение целой части аргумента	3
FMT($\begin{Bmatrix} I \\ F \\ E \end{Bmatrix}$ <число знаков>)	Установка формата вывода числовой информации	1
FRE(x)	Выдача адреса первой свободной ячейки	3 4
HEX\$(x)	Преобразование целого десятичного числа в 16-ричное	3 4
INP(x)	Однobaйтовое целое число, считанное из порта ввода с адресом x	3 4
INPUT\$(x,y)	Строка из x байтов, считанных из файла с номером y	3 4
INSTR([J],b1,b2)	Поиск вхождения одного текстового выражения b1 в другое b2	3 4
INT(x)	Ближайшее целое, x	1 2 3 4
LEFT\$(b,n)	Выделение в левой подстроке n символов	3 4
LEN(b)	Определение длины текстового выражения	2 3 4
LOC(n)	Определение порядкового номера текущей записи	3 4
LOF(<номер файла>)	Длина файла или число свободных байтов в соответствующем буфере ввода	4
LOG(x)	$\ln x$	1 2 3 4
LOG10(x)	$\lg x$	2
LPOS(x)	Номер текущей позиции печатающего механизма АЦПУ; аргумент фиктивный	3 4
MID\$(b,<номер>,<количество>)	Выделение в центральной подстроке указанного числа символов	3 4
MKD\$(x)	8-мисимвольный текстовый эквивалент вещественного числа с двоичной точностью	3
MKI\$(x)	Двусимвольный текстовый эквивалент целого числа	3
MKS\$(x)	4-символьный текстовый эквивалент вещественного числа	3
OCT('<восьмеричная строка>')	Числовой эквивалент 8-ричной строки	2
OCT\$(x)	Текстовый эквивалент целого десятичного числа, переведенного в 8-ричную систему	3 4
PEEK(<адрес байта>)	Выдача содержимого байта ОП в виде целого числа от 0 до 255	3 4
PEN(<номер состояния светового пера, 0+9>)	Опрос состояния светового пера в указанной точке	3
PGR(<номер позиции>,<символ>)	Занесение символа в позицию свободной строки	1
PLAY(<номер канала>)	Определение, генерирует ли канал звук	4
POINT(<строка>,<столбец>)	Определение состояния заданной точки экрана	3 4
POKE <адрес>,x	Занесение целого числа в байт ОП	3 4
POS(x)	Опрос позиции курсора в строке	1

SYS(7, $\begin{Bmatrix} 0 \\ 1 \end{Bmatrix}$)	Разрешает (0) или запрещает (1) использование нижнего регистра на вводе	2
SYS(8)	Разрешает выполнить команде BYE из программы	2
TAB(n)	Сдвиг на n позиций вправо	1 2 3 4
TAN(x)	tg x	1 2 3 4
TIMES	Выдача системного времени	3 4
TTYSET(255%,n)	Установка длины (n-1) выводной строки на терминале	2
VAL(b)	Перевод строкового значения в число	2 3 4
VARPTR(x)	Определение адреса переменной в памяти	3 4
VARPTR(<номер файла>)	Определение адреса блока управления файлом в памяти	3 4
VARPTR\$(x)	Определение типа переменной и ее адреса в памяти	3 4
Формат директивы	Описание директивы	Номер версии
APPEND	Добавить в хвост	2
AUTO [<нач.значение>][,<шаг>]	Автоматическая нумерация строк с начальным значением и заданным шагом	3 4
BLOAD <имя>[,<адрес>]	Загрузка из архива на МД	3 4
BSAVE <имя>,<адрес>,<длина>	Запоминание файла	3 4
CLEAR	Очистка поля программы и всех областей с внутренними и транзитными переменными	2 3 4
CLEAR N	Очистка поля внутренних переменных	2 3 4
CLEAR P [n1][n2]	Очистка поля программы или удаление заданного фрагмента с сохранением значений всех переменных	2 3 4
CLEAR V	Очистка полей внутренних и общих переменных с сохранением текста текущей программы	2 3 4
CLOAD <имя>	Загрузка из архива в МЛ	4
CLOAD? <имя>	Контрольное чтение с МЛ	4
COMPILE ['<имя>']	Компиляция указанной или текущей программы	2
CONT	Продолжение работы программы после останова	3 4
CSAVE <имя> [V]	Запись программы на МЛ	4
DEL [n1][n2]	Удаление текущей программы в указанных пределах	2
DELETE [n1][$\begin{Bmatrix} - \\ \cdot \end{Bmatrix}$][n2]	Удаление текущей программы в указанных пределах	3 4
EDIT [<номер строки>]	Вызов строки на дисплей для редактирования	3
FILE "[<имя дискеты>:] {<имя файла> <начало имени файла>??}. * {<приставка к имени>}, *}	Вывод всех имен файлов указанного типа	3
KILL	Уничтожить файл из архива	3

LENGTH	Определение количества ячеек памяти, занятых программой, и числа свободных ячеек в буфере пользователя	2
LIST [n1][{-}[n2]]	Вывод фрагмента программы	2 3 4
LISTNH n1-n2	Вывод текста программы без стандартного заголовка	2
LOAD'<имя программы>'	Загрузка программы	3 4
MOTOR	Дистанционное управление двигателем магнитофона	3 4
NAME <старое имя> AS <новое имя>	Переименование архивного файла	3
NEW <имя программы>	Изначальный ввод программы с терминала	2 3 4
OLD <имя программы>	Загрузка архивной программы в ОП	2
PROG	Возврат в режим обычного ввода исходной программы из режима SUB	1
RENAME [<старое имя>,<новое имя>,<имя устройства>]	Переименование архивного файла	2
RENUM <новый нач.номер>,<номер перенумерации>,<шаг>	Перенумерация операторов в программе	3 4
REPLACE <имя>	Замещение архивного файла текущей программой	2
RESEQ n0,n1-n2	Перенумерация операторов в программе	2
RUN '<имя>','п	Вызов и выполнение программы, начиная с указанной строки	2 3 4
RUNNH n	Пуск текущей программы с указанной строки без печати стандартного заголовка	2
SAVE <спецификация файла>	Запись текущей программы в архив	2 3 4
SCR	Очистка памяти с уничтожением имени текущей программы	2
SUB	Режим, позволяющий корректировку вставок, находящихся в архивной программе	1 2
SYSTEM	Выход из Бейсик-системы	3 4
UNSAVE	Убрать файл из архива	2

Баласанян В. Э. и др. Программирование на микроЭВМ «Искра 226». — М., Финансы и статистика, 1987. — 264 с.

Рассмотрен диалоговый вариант бейсика применительно к описанным возможностям ЭВМ «Искра 226».

Геворкян Г. Х., Семенов В. Н. Бейсик — это просто. — М., Радио и связь, 1989. — 144 с.

Описание бейсика дано в популярной форме. Изложены особенности разработки алгоритмов и программ. Приведены программы, которые дают представление о широких возможностях языка.

Дьяконов В. П. Применение персональных ЭВМ и программирование на языке Бейсик. — М., Радио и связь, 1989. — 288 с. 3 4

Описаны версии языка Бейсик современных персональных ЭВМ ДВК-2М, ДВК-3, Электроника ДЗ-28, ЕС-1840, Искра 1030, IBM PC, MSX, ZX-Spectrum и др. Даны основы программирования на Бейсике, рекомендации по эксплуатации ПЭВМ и их периферийного оборудования.

Дьяконов В. П. Справочник по алгоритмам и программам на языке Бейсик для персональных ЭВМ: Справочник. — М., Наука. Гл. ред. физ.-мат. лит., 1987. — 240 с. 3

Даны краткие сведения о современных отечественных и зарубежных микроЭВМ индивидуального пользования. Описаны основные версии бейсика. Изложены основы программирования на бейсике. В справочник включена обширная библиотека прикладных программ на бейсике.

Кетков Ю. Л. Диалог на языке бейсик для мини- и микро-ЭВМ. — М., Наука. Гл. ред. физ.-мат. лит., 1988. — 368 с. 1 2 3 4

Большое внимание в книге уделяется систематизация базовых средств языка и их последующему развитию в различных версиях. Вторая часть книги представляет собой справочное руководство по бейсику.

Кетков Ю. Л. Программирование на БЕЙСИКе. — М., Статистика 1 1978. — 158 с.

Дано описание бейсика, сравнение его с фортраном. Указаны особенности реализации бейсика на ЭВМ. Приведены примеры программ для решения практических задач.

Константинов В. С., Рожнова Н. С. Работа с наборами данных на микроЭВМ на базе языка Бейсик. — М., Финансы и статистика, 1989. — 222 с.

Описывается специфика организации на микроЭВМ «Искра 226» различных структур данных и методов доступа к ним.

Содержатся практические рекомендации по созданию программного обеспечения экономических задач.

Корчак А. Е. Язык программирования бейсик для микроЭВМ. — М., МЦНТИ, МНИИПУ, 1988. — 132 с. 3 4

В книге проведен сравнительный анализ версий языка бейсик с точки зрения функциональных возможностей, особенностей реализации, областей применения. Подробно рассматриваются версии языка бейсик, широко применяемые на 8- и 16-разрядных микроЭВМ.

Куликов В. Д. Курс программирования. — Л., Изд-во ЛГУ, 1982. — с. 73—99.

Учебное пособие по основам программирования. Рассчитано на начинающих с уклоном в математические методы решения задач геологии, геохимии и геофизики.

Кучура Н. А. и др. Персональные ЭВМ Единой системы: Бейсик. — 3 4

М., Финансы и статистика, 1988.— 207 с.

Даны основы бейсика, приведены примеры программ. Раскрываются возможности языка по управлению цветом, обработке графической информации и воспроизведению мелодий.

Моррил Г. Бейсик для ПК ИБМ: Пер. с англ.— М., Финансы и статистика, 1987.— 207 с.

Изложены основные понятия бейсика. Методическое построение данной книги позволяет начинающему пользователю быстро включиться в работу за пультом IBM PC.

Погорелый С. Д., Слободянюк Т. Ф. Программное обеспечение микропроцессорных систем: Справочник.— Киев, Техника, 1985.— сс. 197—234.

Дано описание программирования на различных языках в микропроцессорных системах. Бейсик представлен как интерактивный алгоритмический язык Бейсик-80 с описанием основ языка, порядка работы с файлами и редактированием программ.

Программирование на языке Бейсик—плюс для СМ-4/ В. П. Семик, Б. Р. Манцибович, Д. П. Непочатых и др.— М., Финансы и статистика, 1982.— 246 с.

Описаны средства языка, способы и методика использования их при решении различных задач.

Программное обеспечение микроЭВМ: Практ. пособие для инж.— пед. работников системы проф.-техн. образования: В 11 кн./Под ред. В. Ф. Шаньгина. Кн. 5./Диалоговый язык БЕЙСИК. В. Ф. Шаньгин, А. Я. Пьянзин.— М., Высшая школа, 1987.— 111 с.

Описан диалоговый язык бейсик, рассмотрены различные интерпретирующие системы, позволяющие проводить ввод и редактирование программ, трансляцию и исполнение, средства отладки программ на микроЭВМ.

Пул Л. Работа на персональном компьютере: Пер. с англ.— М., Мир, 1986.— 383 с.

Приведены примеры программирования на бейсике. Описаны способы ввода цифровой и графической информации с помощью дисплея, а также вывода информации на печатающие и громкоговорящие устройства. Много внимания уделено запуску, редактированию и завершению программ.

Симпсон Г. Профессиональная работа на персональном компьютере./Пер. с англ.— М., Финансы и статистика, 1988.— 286 с.

Книга открывает пользователю-непрофессионалу важные элементы технологии и искусства программирования. Проиллюстрирована примерами программ на бейсике.

Трэктон К. Программы на Бейсике для инженерно-технических расчетов: Пер. с англ.— М., Радио и связь, 1985.— 96 с.

Даны около 30 небольших программ на бейсике, предназначенных для решения распространенных инженерно-технических задач. Книга рассчитана на инженерно-технических работников.

Уолш Б. Программирование на Бейсике: Пер. с англ.— М., Радио и связь, 1988.— 336 с.

Книга предназначена как для начинающих, так и для опытных программистов. Приведено большое число детально разработанных примеров и задач с решениями.

Уорт Т. Программирование на языке БЕЙСИК: Пер. с англ.— М., Машиностроение, 1981.— 225 с.

Дано описание бейсика, основы программирования на нем. Книга предназначена для инженеров, занимающихся программированием и решением практических вычислительных задач.

Фокс А., Фокс Д. БЕЙСИК для всех: Курс программирования на языке

БЕЙСИК для начинающих: Пер. с англ.— М., Энергоатомиздат, 1987.— 184 с.

Отличительная особенность данной книги — ориентация на читателей, не имеющих предварительной подготовки по программированию, на пользователей персональных компьютеров, а также занимательный и доступный способ изложения материала.

Хелмс Г. Л. Языки программирования: Краткое руководство: Пер. с англ.— М., Радио и связь, 1985.— сс. 18—49.

Одна из целей книги — помочь программисту понять логику работы программы на незнакомом для него языке без систематического изучения.

Этюды о персональных компьютерах. Пер. с венгерского.— М., Знание, 1988.— 160 с.

Книга адресована начинающим программистам и представляет собой увлекательный самоучитель.

34

ОГЛАВЛЕНИЕ

Введение	3
Описание операторов	4
Описание функций	8
Описание директив	11
Аннотированный список литературы по бейсику	13

Альманах «Наука и быт» готовит к изданию следующие выпуски:

ДЕФИЦИТ — ЭТО ПРОСТО /КУЛИНАРНЫЕ СЕКРЕТЫ СОВЕТСКОЙ ДОМОХОЗЯЙКИ/

«Кулинария — одна из древнейших областей знания, изучение которой человек начал с момента своего зарождения. Опыт, накопленный веками, в наши дни породил пищевую промышленность, приучившую нас не заботиться о приготовлении многих стандартных блюд. Однако сегодня плоды прогресса часто — недоступный дефицит.

Отложите напрасные поиски и приготовьте сами:...»

ЭКОЛОГИЧЕСКИ ЧИСТЫЕ МЕТОДЫ БОРЬБЫ С ДОМАШНИМИ НАСЕКОМЫМИ

В книге даются практические рекомендации по применению безвредных для человека, достаточно простых и доступных методов борьбы с домашними насекомыми. Приводится большое количество незаслуженно забытых народных средств. В книге даются описания и рисунки основных вредных бытовых насекомых.

Сдано в набор 11.10.91.	Подписано к печати 11.10.91.
Формат бумаги 60 × 88 ¹ / ₁₆ .	Отпечатано с готовых позитивов. Печать офсетная.
Печ. л. 1.	Заказ 1150. Тираж 50 000.

Альманах «Наука и быт» ставит своей целью преодоление разрыва между достижениями науки и нашим бытом, мировым опытом бизнеса и нашей безграмотной деловой практикой, достижениями дизайна и убожеством повседневности.

Приглашаем всех заинтересованных лиц к сотрудничеству.

Участие в нашем альманахе — прекрасная реклама для Вас!

Наш адрес: 194156 г. Ленинград, а/я 44.



Выпуск № 2