

Р

МОСКОВСКИЙ ХИМИКО-ТЕХНОЛОГИЧЕСКИЙ ИНСТИТУТ
КАФЕДРА ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

РУКОВОДСТВО ПРОГРАММИСТА
~~~~~  
ЯЗЫК ПРОГРАММИРОВАНИЯ ВТ-МХТИ

1987

# СОДЕРЖАНИЕ

~~~~~

НАИМЕНОВАНИЕ РАЗДЕЛА	ЛИСТ
1. ЯЗЫК ПРОГРАММИРОВАНИЯ ВТ-МХТИ	
2. СПИСОК СЛУЖЕБНЫХ СЛОВ ОПЕРАТОРОВ	
3. ОПИСАНИЕ ОБЪЕКТОВ	
3.1. ОПИСАНИЕ ВЕЩЕСТВЕННЫХ И ЦЕЛЫХ ПЕРЕМЕННЫХ	
3.2. ОПИСАНИЕ СИМВОЛЬНЫХ ПЕРЕМЕННЫХ	
3.3. ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ПЕРЕМЕННЫХ	
3.4. ОПИСАНИЕ ОБЩЕЙ ОБЛАСТИ ПОДПРОГРАММ	
3.5. ОПИСАНИЕ ВНЕШНЕЙ ОБЩЕЙ ОБЛАСТИ	
4. ФУНКЦИИ	
4.1. СТАНДАРТНЫЕ ФУНКЦИИ	
4.2. ФУНКЦИИ ОТ ВЕЩЕСТВЕННЫХ МАССИВОВ	
4.3. СПЕЦИАЛЬНЫЕ ФУНКЦИИ ОТ МАССИВОВ	
4.4. ГЕНЕРАЦИЯ СЛУЧАЙНЫХ ЧИСЕЛ	
4.5. РЕШЕНИЕ УРАВНЕНИЙ	
5. ПАУЗА	
6. АДРЕС ПЕРЕМЕННОЙ	
7. ОПЕРАТОРЫ ПРИСВАИВАНИЯ	
7.1. СКАЛЯРНЫЕ ОПЕРАТОРЫ ПРИСВАИВАНИЯ	
7.2. ЗАДАНИЕ ЗНАЧЕНИЙ ЭЛЕМЕНТАМ МАССИВА	
7.3. ОПЕРАЦИИ С МАССИВАМИ И МАТРИЦАМИ	
7.4. СТРОКА ОПЕРАТОРОВ	
8. СТРОКА КОДОВ	
9. СТРУКТУРЫ	
9.1. УСЛОВНЫЙ "IF"	
9.2. ЦИКЛ ПОКА "WHILE"	
9.3. ЦИКЛ "DO"	
9.4. ВЫБОР "CASE"	
9.5. БЛОК "BEGIN"	
10. ОБРАБОТКА СИМВОЛЬНОЙ ИНФОРМАЦИИ	
10.1. СИМВОЛЬНЫЕ ПЕРЕМЕННЫЕ	
10.2. СИМВОЛЬНЫЕ КОНСТАНТЫ	
10.3. ОПЕРАЦИИ НАД СИМВОЛЬНЫМИ ПЕРЕМЕННЫМИ	
10.4. РЕАЛИЗАЦИЯ СИМВОЛЬНОГО ПОИСКА	
11. ВВОД/ВЫВОД ИНФОРМАЦИИ	
11.1. ОПЕРАТОРЫ ВВОДА/ВЫВОДА	
11.2. СПИСОК ОБЪЕКТОВ ВВОДА/ВЫВОДА	
11.3. ФОРМАТ ВВОДА/ВЫВОДА	
ПРИМЕРЫ ОПЕРАТОРОВ ВВОДА/ВЫВОДА	
ДИСПЛЕЙ И ПЕЧАТЬ	
11.4. ВВОД/ВЫВОД ЧИСЛОВОЙ ИНФОРМАЦИИ	
1. БЕСФОРМАТНЫЙ И ФОРМАТНЫЙ ВВОД/ВЫВОД	
2. ПЕРЕМЕННЫЙ ФОРМАТ ВЫВОДА	
3. СМЕНА ВВОДА/ВЫВОДА	
4. ФОРМАТИЗАЦИЯ ВВОДА/ВЫВОДА МАССИВОВ	
11.5. ВВОД/ВЫВОД СИМВОЛЬНОЙ ИНФОРМАЦИИ	
1. БЕСФОРМАТНЫЙ ВВОД/ВЫВОД	
2. ФОРМАТИЗИРОВАННЫЙ ВВОД/ВЫВОД	
3. ПОСТРОЕНИЕ СИМВОЛЬНЫХ ИЗОБРАЖЕНИЙ НА ЭКРАНЕ	
4. АСИНХРОННЫЙ ВВОД С КЛАВИАТУРЫ ДИСПЛЕЯ	

МАГНИТНАЯ ЛЕНТА

- 11.6. ВВОД/ВЫВОД ДАННЫХ
 - 1. БЕСФОРМАТНЫЙ ВВОД/ВЫВОД
 - 2. ФОРМАТИЗИРОВАННЫЙ ВВОД/ВЫВОД
- 11.7. ВВОД СЕГМЕНТОВ РАБОЧИХ ПРОГРАММ
- 12. ОПЕРАТОРЫ ПЕРЕХОДОВ
- 12.1. БЕЗУСЛОВНЫЕ ПЕРЕХОДЫ
- 1. ЯВНЫЙ ПЕРЕХОД НА ЧИСЛОВУЮ МЕТКУ
- 2. НЕЯВНЫЙ ПЕРЕХОД НА ЧИСЛОВУЮ МЕТКУ
- 3. ПЕРЕХОД НА СИМВОЛЬНУЮ МЕТКУ
- 4. ПЕРЕХОД ПО СПИСКУ С УПРАВЛЯЮЩЕЙ ПЕРЕМЕННОЙ ..
- 5. ПЕРЕХОД ПО СПИСКУ С КЛАВИАТУРЫ ДИСПЛЕЯ
- 12.2. УСЛОВНЫЕ ПЕРЕХОДЫ
- 1. ПЕРЕХОД ПО ЗНАКУ ВЫРАЖЕНИЯ
- 2. ПЕРЕХОД НА СИМВОЛЬНУЮ МЕТКУ ПО УСЛОВИЮ
- 13. ВСТРОЕННЫЕ ПОДПРОГРАММЫ
- 14. ОБРАБОТКА ВНЕШНИХ ПРЕРЫВАНИЙ
- 15. ПРОГРАММЫ, ПОДПРОГРАММЫ, ОПЕРАТОР-ФУНКЦИИ
- 15.1. ОФОРМЛЕНИЕ ПРОГРАММ
- 1. ЗАГОЛОВОК ПРОГРАММЫ
- 2. ВВОД ЗНАЧЕНИЙ ПАРАМЕТРОВ ПРОГРАММЫ
- 3. ОБЩИЕ ПРАВИЛА ПОСТРОЕНИЯ ОПИСАНИЙ ОБЪЕКТОВ
ПРОГРАММЫ
- 4. ОБРАЩЕНИЯ К ПРОГРАММАМ
- 5. ВЫХОД ИЗ ПРОГРАММЫ
- 6. ВЛОЖЕННЫЕ ПРОГРАММЫ
- 7. ПОДПРОГРАММЫ-ФУНКЦИИ
- 8. ОПЕРАТОР-ФУНКЦИИ
- 15.2. СЕГМЕНТИРОВАНИЕ ПРОГРАММ
- 1. НЕЗАВИСИМАЯ ПЕРЕДАЧА УПРАВЛЕНИЯ
- 2. ПЕРЕДАЧА УПРАВЛЕНИЯ С ОБЩИМ ПОЛЕМ ДАННЫХ
- 16. ПРИЛОЖЕНИЯ

1. ЯЗЫК ПРОГРАММИРОВАНИЯ ВТ-МХТИ.

ЯЗЫК ВТ-МХТИ ДЛЯ МИКРО-ЭВМ СОЗДАН НА ОСНОВЕ КРИТИЧЕСКОГО АНАЛИЗА СУЩЕСТВУЮЩИХ В НАСТОЯЩЕЕ ВРЕМЯ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ ВЫСОКОГО УРОВНЯ. ИЗ НИХ ОТОБРАНЫ НАИБОЛЕЕ ЭФФЕКТИВНЫЕ СРЕДСТВА ДОПОЛНЕННЫЕ НЕКОТОРЫМИ ДРУГИМИ РАЗРАБОТКАМИ, УПРОЩАЮЩИМИ ИСПОЛЬЗОВАНИЕ МИКРО-ЭВМ ДЛЯ РЕШЕНИЯ РАЗНООБРАЗНЫХ ЗАДАЧ.

РЕАЛИЗАЦИЯ ОПЕРАЦИОННОЙ СИСТЕМЫ И КОМПИЛЯТОРА С ЯЗЫКА ВТ-МХТИ ВЫПОЛНЕНА ДЛЯ МИКРО-ЭВМ "ДЗ-28" И ПОЗВОЛЯЕТ ЭФФЕКТИВНО ПРИМЕНЯТЬ ЕЕ ДЛЯ РЕШЕНИЯ САМЫХ РАЗНООБРАЗНЫХ НАУЧНО-ТЕХНИЧЕСКИХ ЗАДАЧ, ВКЛЮЧАЯ СБОР И ОБРАБОТКУ ИНФОРМАЦИИ В АВТОМАТИЗИРОВАННЫХ СИСТЕМАХ НАУЧНЫХ ИССЛЕДОВАНИЙ И СИСТЕМАХ УПРАВЛЕНИЯ.

СРЕДСТВА ЯЗЫКА ВТ-МХТИ ПОЗВОЛЯЮТ С МИНИМАЛЬНОЙ АДАПТАЦИЕЙ ВОСПРОИЗВОДИТЬ ПРОГРАММЫ, ЗАПИСАННЫЕ НА ДРУГИХ ЯЗЫКАХ ПРОГРАММИРОВАНИЯ. РАЗРАБОТАННЫЙ ДЛЯ "ДЗ-28" КОМПИЛЯТОР ДАЕТ ВОЗМОЖНОСТЬ ПО ВЫБОРУ ПОЛЬЗОВАТЕЛЯ РАБОТАТЬ КАК В РУССКОЙ, ТАК И В АНГЛИЙСКОЙ НОТАЦИИ СЛУЖЕБНЫХ СЛОВ ЯЗЫКА.

ОПЕРАТОРЫ В ЯЗЫКЕ ВТ-МХТИ, КАК И В БОЛЬШИНСТВЕ ДРУГИХ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ ВЫСОКОГО УРОВНЯ, ОПРЕДЕЛЯЮТСЯ СЛУЖЕБНЫМИ СЛОВАМИ. ИСКЛЮЧЕНИЕ СОСТАВЛЯЕТ ЛИШЬ ОПЕРАТОР ПРИСВАИВАНИЯ, ДЛЯ КОТОРОГО В ПЕРВОЙ ПОЗИЦИИ СЛУЖЕБНОГО СЛОВА ИСПОЛЬЗУЕТСЯ СИМВОЛ "ПРОБЕЛ".

2. СПИСОК СЛУЖЕБНЫХ СЛОВ ОПЕРАТОРОВ.

В ПРИВЕДЕННОМ СПИСКЕ СЛУЖЕБНЫХ СЛОВ ЯЗЫКА СИМВОЛЫ, ВВОДИМЫЕ ПОЛЬЗОВАТЕЛЕМ, ВЫДЕЛЕНА СИМВОЛАМИ "[...]". ОСТАЛЬНАЯ ЧАСТЬ СЛОВА ВЫВОДИТСЯ КОМПИЛЯТОРОМ АВТОМАТИЧЕСКИ.

РУССКАЯ НОТАЦИЯ	АНГЛИЙСКАЯ НОТАЦИЯ	НАЗНАЧЕНИЕ
ПРОГРАММА [ИМЯ...]	PROGRAM [ИМЯ...]	- ЗАГОЛОВОК ПРОГРАММЫ
[ПРОБЕЛ...]	[ПРОБЕЛ...]	- СТРОКА ОПЕРАТОРОВ
[П]ОГРАММА [:]	[P]ROGRAM [:]	- ОПИСАТЕЛЬ ИМЕН ПРОГРАММ
[Н]АЧ	[B]EGIN	- НАЧАЛО БЛОКА
[К]ОН	[E]ND	- КОНЕЦ ПРОГРАММЫ, БЛОКА
[В]СЕ	[E]ND	- КОНЕЦ ВЫБОРА, УСЛОВИЯ
[К]Ц	[E]ND	- КОНЕЦ ЦИКЛА
[Д]ЛЯ	[D]O	- НАЧАЛО ЦИКЛА "DO"
[П]ОКА	[W]HILE	- НАЧАЛО ЦИКЛА "WHILE"
[Е]СЛИ	[I]F	- УСЛОВИЕ
[И]НАЧЕ	[E]LSE	- АЛЬТЕРНАТИВА УСЛОВИЯ
[В]ЫХОД	[R]ETURN	- ВЫХОД ИЗ ПРОГРАММЫ
[В]ЫБРАТЬ	[C]ASE	- ВЫБОР АЛЬТЕРНАТИВЫ
-	[C]ALL	- ПЕРЕХОД К ПРОГРАММЕ
[В]ВОД	[R]EAD	- ВВОД
[В]ЫВОД	[W]RITE	- ВЫВОД
[Ф]ОРМАТ	[F]ORMAT	- ФОРМАТ
[С]ЕД	[G]OTO	- ПЕРЕХОД
[М]ЕТКА [ИМЯ]	[L]ABEL [ИМЯ]	- ОПИСАТЕЛЬ СИМВОЛЬНОЙ МЕТКИ
[М]ЕТКА [:]	[L]ABEL [:]	- ОПИСАТЕЛЬ ИМЕН МЕТОК
[В]ЕЩЬ:	[R]EAL :	- ОПИСАТЕЛЬ ВЕЩЕСТВЕННОГО ТИПА
[Ц]ЕЛ:	[I]NTEGER:	- ОПИСАТЕЛЬ ЦЕЛОГО ТИПА
[С]ИМВОЛ:	[C]HAR:	- ОПИСАТЕЛЬ СИМВОЛЬНОГО ТИПА

[O]БЩ	[CO]ММОН	- ОПИСАТЕЛЬ ОБЩЕЙ ОБЛАСТИ
[O]БЩ[#]	[CO]ММОН[#]	- ОПИСАТЕЛЬ РАСШИРЕН. ПАМЯТИ
[З]НАЧ:	[DA]ТА:	- ЗАДАНИЕ ЗНАЧЕНИЙ МАССИВУ
[Т],с	[Т],s	- ПАУЗА В СЕК.
[[...]]	[[...]]	- СТРОКА КОДОВ ЭВМ
[А]ДР:	[А]DR:	- АДРЕС ПЕРЕМЕННОЙ

ИМЕЮТСЯ НЕСУЩЕСТВЕННЫЕ ОТЛИЧИЯ В СОСТАВЕ РУССКОЙ И АНГЛИЙСКОЙ НОТАЦИИ СЛУЖЕБНЫХ СЛОВ ЯЗЫКА. ТАК, В ТЕКСТАХ ПРОГРАММ НА РУССКОМ ЯЗЫКЕ ОБРАЩЕНИЕ К ПРОГРАММЕ ЗАДАЕТСЯ УКАЗАНИЕМ ЕЕ ИМЕНИ В СТРОКЕ ОПЕРАТОРОВ. В АНГЛИЙСКОЙ НОТАЦИИ НАРЯДУ С АНАЛОГИЧНЫМ СПОСОБОМ СОХРАНЕНЫ ВЫЗОВЫ "CALL" ДЛЯ ЗАПИСИ ПРОГРАММ НА FORTRAN. В ЛИСТИНГЕ ПРОГРАММ, ЗАПИСАННЫХ В АНГЛИЙСКОЙ НОТАЦИИ, ПРИ ПЕРЕХОДЕ НА РУССКУЮ НОТАЦИЮ ВЫЗОВЫ "CALL" АВТОМАТИЧЕСКИ ОПУСКАЮТСЯ.

3. ОПИСАНИЕ ОБЪЕКТОВ.

~~~~~

ИМЕНА ОБЪЕКТОВ ПРОГРАММЫ НА ЯЗЫКЕ ВТ-МХТИ РАЗЛИЧАЮТСЯ ПО ПЕРВЫМ ЧЕТЫРЕМ СИМВОЛАМ И НАЧИНАЮТСЯ С БУКВЫ, ЗА КОТОРОЙ МОЖЕТ СЛЕДОВАТЬ ПРОИЗВОЛЬНОЕ ЧИСЛО БУКВ И ЦИФР. ДЛЯ КАЖДОГО ОБЪЕКТА ЗАДАЕТСЯ ЕГО ТИП, ОПРЕДЕЛЯЕМЫЙ СООТВЕТСТВУЮЩИМ ОПЕРАТОРОМ ОПИСАНИЯ ТИПА. ДОПУСКАЮТСЯ ЧЕТЫРЕ ОСНОВНЫХ ТИПА ОБЪЕКТОВ ПРОГРАММЫ:

- |                    |            |
|--------------------|------------|
| 1. ВЕЩЕСТВЕННЫЙ    | - REAL:    |
| 2. ЦЕЛЫЙ           | - INTEGER: |
| 3. СИМВОЛЬНЫЙ      | - CHAR:    |
| 4. ФУНКЦИОНАЛЬНЫЕ: | - LABEL:   |
|                    | - PROGRAM: |

ЛЮБЫЕ ОПИСАНИЯ МОГУТ РАСПОЛАГАТЬСЯ В ЛЮБОМ МЕСТЕ ПРОГРАММЫ. ИСКЛЮЧЕНИЕМ ЯВЛЯЕТСЯ ОПИСАНИЕ "PROGRAM:...", КОТОРОЕ МОЖЕТ БЫТЬ ПОМЕЩЕНО ТОЛЬКО НА НУЛЕВОМ УРОВНЕ ОПЕРАТОРОВ (Т.Е. ВНЕ БЛОКОВ "BEGIN", СТРУКТУРНЫХ ОПЕРАТОРОВ: "IF", "DO", "WHILE", ВЛОЖЕННЫХ ПРОГРАММ). ОБЛАСТЬ ДЕЙСТВИЯ ОПИСАНИЙ РАСПРОСТРАНЯЕТСЯ НА ВСЕ ОПЕРАТОРЫ НИЖЕСЛЕДУЮЩЕЙ ЧАСТИ ПРОГРАММЫ, В КОТОРОЙ ОПИСАННЫЕ ОБЪЕКТЫ ЯВЛЯЮТСЯ ГЛОБАЛЬНЫМИ ПО ОТНОШЕНИЮ К СТРУКТУРАМ (БЛОКАМ, ВЛОЖЕННЫМ ПРОГРАММАМ), ИМЕЮЩИМ СОБСТВЕННЫЕ (ЛОКАЛЬНЫЕ) ОПИСАНИЯ.

#### 3.1. ОПИСАНИЕ ВЕЩЕСТВЕННЫХ И ЦЕЛЫХ ПЕРЕМЕННЫХ.

~~~~~

INTEGER:D,DATA,D10,Поле,N,Nota,ind
 REAL:Время,S,Длина,CLOCK

ОПИСАНИЕ МАССИВА ПЕРЕМЕННЫХ С ИНДЕКСАМИ (КОЛИЧЕСТВО КОТОРЫХ МОЖЕТ БЫТЬ ОТ 1 ДО 3) СОСТОИТ ИЗ ИМЕНИ, ЗА КОТОРЫМ В КВАДРАТНЫХ СКОБКАХ "[]" СЛЕДУЕТ СПИСОК ГРАНИЦ ИНДЕКСОВ. ЗНАЧЕНИЯ ГРАНИЦ ИНДЕКСОВ ЗАДАЮТСЯ ЛИБО ЦЕЛЫМ ЧИСЛОМ (ДО 255), ЛИБО ЦЕЛОЙ ПЕРЕМЕННОЙ (ПРИ ЭТОМ ЗНАЧЕНИЕ ИНДЕКСА МОЖЕТ БЫТЬ БОЛЬШЕ 255). ЗНАЧЕНИЕ ПЕРЕМЕННОЙ-ИНДЕКСА К МОМЕНТУ ОПИСАНИЯ МАССИВА ДОЛЖНО БЫТЬ ОПРЕДЕЛЕНО (ОПЕРАТОРАМИ ПРИСВАИВАНИЯ ИЛИ ВВОДА).

```

READ ind;N;Д
D10:=Д+1
Поле:=200
INTEGER:ТАБЛ[200],SIMP[N,3]
REAL:X[D10,Поле,3]

```

ИНДЕКСЫ ЭЛЕМЕНТОВ МАССИВА ИЗМЕНЯЮТСЯ ОТ 1 ДО СООТВЕТСТВУЮЩЕЙ ВЕРХНЕЙ ГРАНИЦЫ. КОЛИЧЕСТВО ЭЛЕМЕНТОВ В МАССИВЕ РАВНО ПРОИЗВЕДЕНИЮ ВСЕХ ВЕРХНИХ ГРАНИЦ. НЕСКОЛЬКО ОПЕРАТОРОВ ОПИСАНИЯ ТИПА МОГУТ РАСПОЛОГАТЬСЯ В ОДНОЙ СТРОКЕ. РАЗДЕЛИТЕЛЕМ ОПИСАНИЙ ЯВЛЯЕТСЯ СИМВОЛ ";".

```

REAL:Z,A[3]; INTEGER:КОНТ,DAMP[ind,8]; REAL:G,B,C

```

ОПИСАНИЯ, РАСПОЛОЖЕННЫЕ В ОДНОЙ СТРОКЕ ВСЛЕД ЗА ПЕРВЫМ, МОГУТ ЗАДАВАТЬСЯ ПЕРВОЙ БУКВОЙ СЛУЖЕБНОГО СЛОВА. ПРЕДЫДУЩИЙ ПРИМЕР МОЖЕТ БЫТЬ ЗАПИСАН ТАК:

```

REAL:Z,A[3]; I:КОНТ,DAMP[ind,8]; R:G,B,C

```

PP

3.2. ОПИСАНИЯ СИМВОЛЬНЫХ ПЕРЕМЕННЫХ.

~~~~~

```

INTEGER:ind
CHAR:FIO(30),ИМЯ(ind)

```

В КРУГЛЫХ СКОБКАХ УКАЗЫВАЕТСЯ РАЗМЕР СИМВОЛЬНОЙ ПЕРЕМЕННОЙ В БАЙТАХ (СИМВОЛАХ). В ВТ-МХТИ В КАЧЕСТВЕ СИМВОЛЬНЫХ МОГУТ ИСПОЛЬЗОВАТЬСЯ ТАКЖЕ ЦЕЛЫЕ, ВЕЩЕСТВЕННЫЕ ПЕРЕМЕННЫЕ И МАССИВЫ. ПРИ ЭТОМ РАЗМЕР СИМВОЛЬНОЙ ПЕРЕМЕННОЙ В БАЙТАХ ДЛЯ:

|                         |                            |
|-------------------------|----------------------------|
| ЦЕЛОЙ ПЕРЕМЕННОЙ        | - 2 БАЙТА;                 |
| ВЕЩЕСТВЕННОЙ ПЕРЕМЕННОЙ | - 8 БАЙТ;                  |
| ЦЕЛОГО МАССИВА          | - 2*(РАЗМЕР МАССИВА) БАЙТ; |
| ВЕЩЕСТВЕННОГО МАССИВА   | - 8*(РАЗМЕР МАССИВА) БАЙТ. |

ОПИСАНИЯ СИМВОЛЬНЫХ ПЕРЕМЕННЫХ ТАКЖЕ МОГУТ РАСПОЛАГАТЬСЯ В ОДНОЙ СТРОКЕ С ДРУГИМИ ОПИСАНИЯМИ.

```

REAL:Z,A[3]; I:КОНТ,DAMP[ind,8]; C:Поле(КОНТ)

```

ЧИСЛОМ МОЖЕТ ЗАДАВАТЬСЯ РАЗМЕР СИМВОЛЬНОЙ ПЕРЕМЕННОЙ НЕ БОЛЕЕ 255. ЦЕЛАЯ ПЕРЕМЕННАЯ МОЖЕТ ЗАДАВАТЬ И БОЛЬШИЙ РАЗМЕР.

### 3.3. ОПИСАНИЯ ФУНКЦИОНАЛЬНЫХ ПЕРЕМЕННЫХ.

~~~~~

В ВТ-МХТИ В ФУНКЦИОНАЛЬНОМ ОПИСАНИИ УКАЗЫВАЮТСЯ ИМЕНА: ИСПОЛЬЗУЕМЫХ В ДАННОЙ ПРОГРАММЕ СИМВОЛЬНЫХ МЕТОК И ДРУГИХ ПРОГРАММ. ОПИСАНИЕ ОБЯЗАТЕЛЬНО ДОЛЖНО ПРЕДШЕСТВОВАТЬ ИСПОЛЬЗОВАНИЮ САМИХ ФУНКЦИОНАЛЬНЫХ ПЕРЕМЕННЫХ (СМ. РАЗДЕЛЫ: 12, 15)

```

LABEL:ГРУП,ВЫВ
PROGRAM:ПОИСК,СПУСК
LABEL:ПЛЮС,INVER; P:ТЕМА,SALL

```

ИМЯ СИМВОЛЬНОЙ МЕТКИ ПОСЛЕ ОПИСАНИЯ СТАНОВИТСЯ ГЛОБАЛЬНЫМ ОБЪЕКТОМ ДЛЯ ВСЕЙ НИЖЕСЛЕДУЮЩЕЙ ЧАСТИ ПРОГРАММЫ НЕЗАВИСИМО ОТ ТОГО, НА КАКОМ УРОВНЕ (В БЛОКЕ, СТРУКТУРНОМ ОПЕРАТОРЕ, ВЛОЖЕННОМ БЛОКЕ И Т.П.) УКАЗАНО ЭТО ОПИСАНИЕ.

ФУНКЦИОНАЛЬНЫЕ ПЕРЕМЕННЫЕ ИМЕЮТ ФИЗИЧЕСКИЕ АДРЕСА В РАБОЧЕЙ ПРОГРАММЕ И ПРИ НЕОБХОДИМОСТИ МОГУТ ИСПОЛЬЗОВАТЬСЯ В ПРОГРАММЕ КАК ПРОСТЫЕ ПЕРЕМЕННЫЕ ЦЕЛОГО ТИПА.

ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ПЕРЕМЕННЫХ ТАКЖЕ МОЖЕТ ЗАПИСЫВАТЬСЯ В ОДНОЙ СТРОКЕ С ДРУГИМИ ВИДАМИ ОПИСАНИЙ.

3.4. ОБЩАЯ ОБЛАСТЬ ПОДПРОГРАММ.

~~~~~

ОБЩАЯ ОБЛАСТЬ ПОДПРОГРАММ ЗАДАЕТСЯ ДЛЯ РЕАЛИЗАЦИИ ПРЯМОГО ОБМЕНА ИНФОРМАЦИЕЙ МЕЖДУ ВЫЗЫВАЮЩЕЙ И ВЫЗЫВАЕМОЙ ПРОГРАММАМИ. ОПИСАНИЕ ОБЩЕЙ ОБЛАСТИ ФОРМИРУЕТСЯ В ВЫЗЫВАЮЩЕЙ (ОСНОВНОЙ) ПРОГРАММЕ ОПЕРАТОРОМ:

```
INTEGER:ТАБЛ[200];R:X[2,3,5]
COMMON I:N,FLAG,ТАБЛ#,QR[N,4];R:X#,WW[10],S;P:ПРОГ,ПОИС;C:ФИО#
```

С ОБЯЗАТЕЛЬНЫМ ОПИСАНИЕМ ТИПА ВСЕХ ВКЛЮЧАЕМЫХ В ОБЩУЮ ОБЛАСТЬ ПЕРЕМЕННЫХ. ПРИ ВКЛЮЧЕНИИ В ОБЩУЮ ОБЛАСТЬ МАССИВА, РАНЕЕ ОПИСАННОГО В ПРОГРАММЕ, ПОСЛЕ ЕГО ИМЕНИ УКАЗЫВАЕТСЯ СИМВОЛ "#" БЕЗ УКАЗАНИЯ РАЗМЕРА МАССИВА. В ВЫЗЫВАЮЩЕЙ ПРОГРАММЕ ОБЩАЯ ОБЛАСТЬ ФОРМИРУЕТСЯ АНАЛОГИЧНО, НО ВОЗМОЖНО С ДРУГИМИ ИМЕНАМИ ПЕРЕМЕННЫХ, МЕТОК, КРОМЕ ИМЕН ПРОГРАММ, КОТОРЫЕ ДОЛЖНЫ БЫТЬ ИДЕНТИЧНЫ ВО ВСЕХ ПРОГРАММАХ, ИСПОЛЬЗУЮЩИХ ОБЩУЮ ОБЛАСТЬ.

В ВЫЗЫВАЕМОЙ ПРОГРАММЕ (ПОДПРОГРАММЕ) ПРИ ОПИСАНИИ ОБЩЕЙ ОБЛАСТИ НЕ ОБЯЗАТЕЛЬНО УКАЗЫВАТЬ ВСЕ ОБЪЕКТЫ. ТЕ ИЗ НИХ, КОТОРЫЕ НЕ ИСПОЛЬЗУЮТСЯ В ДАННОЙ ПРОГРАММЕ МОГУТ БЫТЬ ПРОПУЩЕНЫ В ОПИСАНИИ С ПОМОЩЬЮ УКАЗАТЕЛЯ ПРОПУСКА - ЦЕЛОГО ЧИСЛА, ПОМЕЩАЕМОГО В СООТВЕТСТВУЮЩЕМ МЕСТЕ ОПИСАНИЯ. НАПРИМЕР:

```
COMMON I:M,2,QR[N,4]; 2; R:SS; P:ПРОГ,1; C:ФИО(30)
```

ЗДЕСЬ В ОПИСАНИИ ОПУЩЕНЫ: ПЕРЕМЕННАЯ "FLAG", МАССИВЫ:"ТАБЛ", "X", "WW" И ИМЯ ПРОГРАММЫ "ПОИС". ВЫЗЫВАЕМАЯ ПРОГРАММА (ПОДПРОГРАММА) В СВОЮ ОЧЕРЕДЬ МОЖЕТ РАСШИРИТЬ ОБЩУЮ ОБЛАСТЬ ДЛЯ ОБМЕНА ДАННЫМИ С ВЫЗЫВАЕМЫМИ ЕЮ ПРОГРАММАМИ (ПОДПРОГРАММАМИ). НАПРИМЕР, ПРЕДЫДУЩЕЕ ОПИСАНИЕ МОЖЕТ БЫТЬ РАСШИРЕНО ТАК:

```
COMMON I:N,5; R:SS; 1;P:ПОИС; C:ФИФ(30); R:ZY,V[5]; I:ММ
```

ЗДЕСЬ ОБЩАЯ ОБЛАСТЬ РАСШИРЕНА ВКЛЮЧЕНИЕМ ПЕРЕМЕННЫХ "ZY", "ММ" И МАССИВА "V[5]".

В ВЫЗЫВАЕМОЙ ПРОГРАММЕ ПРИ ОПИСАНИИ МАССИВА В ОБЩЕЙ ОБЛАСТИ ЕГО ГРАНИЦЫ (РАЗМЕРЫ) МОГУТ ОТЛИЧАТЬСЯ ОТ ЗАДАННЫХ В ВЫЗЫВАЮЩЕЙ ПРОГРАММЕ. ПРИ СОВМЕСТНОЙ РАБОТЕ ТАКИХ ПРОГРАММ ОПРЕДЕЛЯЮЩИМ ЯВЛЯЕТСЯ ОПИСАНИЕ В ВЫЗЫВАЮЩЕЙ ПРОГРАММЕ, ЧТО НЕОБХОДИМО УЧИТЫВАТЬ В ОПЕРАТОРАХ, ИСПОЛЬЗУЮЩИХ ЭЛЕМЕНТЫ ЭТОГО МАССИВА.

### 3.5. ВНЕШНЯЯ ОБЩАЯ ОБЛАСТЬ.

~~~~~

ДЛЯ УВЕЛИЧЕНИЯ ОБЪЕМА ДАННЫХ, ОБРАБАТЫВАЕМЫХ ПРОГРАММОЙ, МОЖЕТ ИСПОЛЬЗОВАТЬСЯ РАСШИРЕННАЯ ПАМЯТЬ "ДЗ-28", НАЧИНАЯ С СЕГМЕНТА 0. МАССИВЫ ДАННЫХ, РАЗМЕЩЕННЫЕ В РАБОЧЕЙ ОБЛАСТИ ПАМЯТИ (СЕГМЕНТЫ С, D, E, F), МОГУТ ПЕРЕСЫЛАТЬСЯ В РАСШИРЕННУЮ ПАМЯТЬ И ОБРАТНО С ПОМОЩЬЮ СПЕЦИАЛЬНЫХ ОПЕРАТОРОВ ОБМЕНА. ОПИСАНИЕ МАССИВОВ (ТОЛЬКО ВЕЩЕСТВЕННЫХ !) В РАСШИРЕННОЙ ПАМЯТИ ЗАДАЕТСЯ ОПЕРАТОРОМ:

```
COMMON#XX[КОНТ] , YU [Длин*Врем] , ZZ [Nota+ind]
```

МАССИВЫ РАСШИРЕННОЙ ПАМЯТИ ТОЛЬКО ОДНОМЕРНЫЕ. ПРИ ЗАДАНИИ ИХ РАЗМЕРОВ МОГУТ ИСПОЛЬЗОВАТЬСЯ ПРОИЗВОЛЬНЫЕ ВЫРАЖЕНИЯ ЦЕЛОГО ИЛИ ВЕЩЕСТВЕННОГО ТИПА. В ПОСЛЕДНЕМ СЛУЧАЕ БЕРЕТСЯ ЦЕЛАЯ ЧАСТЬ.

ПЕРЕСЫЛКА МАССИВОВ (ВЕЩЕСТВЕННЫХ И ЦЕЛЫХ) ИЗ РАБОЧЕЙ ОБЛАСТИ ПАМЯТИ В РАСШИРЕННУЮ ПРОИЗВОДИТСЯ ОПЕРАТОРАМИ:

```
#XX:=FIO
#XX[25] :=DAMP
#YU:=ТАБЛ[3]
#ZZ[N+4] :=X[N, 2, 4]
```

ИНДЕКСАЦИЯ МАССИВОВ РАСШИРЕННОЙ ПАМЯТИ ПРИ ЭТОМ УКАЗЫВАЕТ НОМЕР ЭЛЕМЕНТА, С КОТОРОГО БУДЕТ РАЗМЕЩАТЬСЯ ПЕРЕСЫЛАЕМЫЙ МАССИВ. ИНДЕКСАЦИЯ ПЕРЕСЫЛАЕМОГО МАССИВА - НОМЕР ЭЛЕМЕНТА НАЧИНАЯ С КОТОРОГО И ДО КОНЦА ПЕРЕСЫЛАЕТСЯ МАССИВ.

В Н И М А Н И Е !!!

ВСЕГДА СЛЕДУЕТ ПОМНИТЬ, ЧТО ОБЪЕМ ПЕРЕСЫЛАЕМОЙ ИНФОРМАЦИИ ОПРЕДЕЛЯЕТСЯ РАЗМЕРОМ МАССИВА, РАСПОЛОЖЕННОГО В РАБОЧЕЙ ПАМЯТИ И НЕ ЗАВИСИТ ОТ РАЗМЕРА МАССИВА РАСШИРЕННОЙ ПАМЯТИ.

ПРИ ПЕРЕСЫЛКЕ ЦЕЛЫХ ИЛИ СИМВОЛЬНЫХ МАССИВОВ ИМ ОТВОДИТСЯ В РАСШИРЕННОЙ ПАМЯТИ ЧИСЛО ЭЛЕМЕНТОВ ВЕЩЕСТВЕННОГО МАССИВА, ЭКВИВАЛЕНТНОЕ ИХ СУММАРНОМУ РАЗМЕРУ. НАПРИМЕР, ПРИ ПЕРЕСЫЛКЕ ЦЕЛОГО МАССИВА "M[10-]", РАЗМЕР КОТОРОГО РАВЕН 20 БАЙТ, ОН РАЗМЕСТИТСЯ В 3-Х ЭЛЕМЕНТАХ МАССИВА РАСШИРЕННОЙ ПАМЯТИ.

ЗНАЧЕНИЯ ИНДЕКСОВ У МАССИВОВ РАСШИРЕННОЙ ПАМЯТИ В ОПЕРАТОРАХ ПЕРЕСЫЛКИ МОГУТ ВЫХОДИТЬ ЗА ПРЕДЕЛЫ, УСТАНОВЛЕННЫЕ ПРИ ИХ ОПИСАНИИ, АДРЕСУЯ ПРИ ЭТОМ ДАННЫЕ В ЭЛЕМЕНТЫ СЛЕДУЮЩИХ МАССИВОВ. НАПРИМЕР, ПЕРЕСЫЛКА:

```
#XX[КОНТ+2] :=FIO
      ЭКВИВАЛЕНТНА ПЕРЕСЫЛКЕ:
#YU[2] :=FIO
```

ВЫЗОВ ИЗ МАССИВОВ РАСШИРЕННОЙ ПАМЯТИ ПРОИЗВОДИТСЯ АНАЛОГИЧНЫМИ ОПЕРАТОРАМИ:

```
#FIO:=XX
#DAMP:=XX[25]
#ТАБЛ[3] :=YU
#X[N, 2, 4] :=ZZ[N+4]
#DAMP[4] :=ZZ[2]
```


ПРИ ВЫЗОВЕ ИЗ РАСШИРЕННОЙ ПАМЯТИ ПРОИЗВОДИТСЯ ЗАПОЛНЕНИЕ МАССИВА С НАЧАЛА ИЛИ С УКАЗАННОГО ЭЛЕМЕНТА ДО КОНЦА. ПЕРЕДАЧА ДАННЫХ ИДЕТ С УКАЗАННОГО ЭЛЕМЕНТА МАССИВА РАСШИРЕННОЙ ПАМЯТИ НЕЗАВИСИМО ОТ ЕГО РАЗМЕРА.

PP

4. ФУНКЦИИ.

~~~~~

##### 4.1. СТАНДАРТНЫЕ ФУНКЦИИ.

~~~~~

В ВТ-МХТИ ИМЕЕТСЯ СЛЕДУЮЩИЙ НАБОР СТАНДАРТНЫХ ФУНКЦИЙ, АРГУМЕНТАМИ КОТОРЫХ МОГУТ БЫТЬ ПРОИЗВОЛЬНЫЕ ВЫРАЖЕНИЯ:

ПЕРЕВОД ГРАД. В РАД. И РАД. В ГРАД.

RAD (X) ; GRAD (X)

ТРИГОНОМЕТРИЧЕСКИЕ ФУНКЦИИ

SIN (X) ; ARCSIN (X)
COS (X) ; ARCCOS (X)
TG (X) ; ARCTG (X)

ГИПЕРБОЛИЧЕСКИЕ ФУНКЦИИ

SH (X) ; ARSH (X)
CH (X) ; ARCH (X+1)
TH (X) ; ARTH (X)

ДЕСЯТИЧНЫЙ ЛОГАРИФМ И СТЕПЕНЬ ДЕСЯТИ

LG (X) ; DEX (X)

НАТУРАЛЬНЫЙ ЛОГАРИФМ И ЭКСПОНЕНТА

LN (X) ; EXP (2*X)

ОБРАТНАЯ ВЕЛИЧИНА И КВАДРАТНЫЙ КОРЕНЬ

INV (X) ; SQRT (X)

АБСОЛЮТНОЕ ЗНАЧЕНИЕ И ФУНКЦИЯ ЗНАКА (СИГНАТУРА)

ABS (X) ; SIGN (X-2)

ЦЕЛАЯ ЧАСТЬ И КРАТНОЕ ПИ

ENT (X) ; PI (X)

4.2. ФУНКЦИИ ОТ ВЕЩЕСТВЕННЫХ МАССИВОВ.

~~~~~

ВСЕ ПЕРЕЧИСЛЕННЫЕ ВЫШЕ ФУНКЦИИ (КРОМЕ "PI") ОПРЕДЕЛЕНА ТАКЖЕ ДЛЯ ВЕЩЕСТВЕННЫХ МАССИВОВ ЛЮБОЙ РАЗМЕРНОСТИ. В КАЧЕСТВЕ АРГУМЕНТА ФУНКЦИИ ПРИ ЭТОМ УКАЗЫВАЕТСЯ ИМЯ МАССИВА. УКАЗАТЕЛЬ ФУНКЦИИ МОЖЕТ ИСПОЛЬЗОВАТЬСЯ ТОЛЬКО В ПРОСТОМ ОПЕРАТОРЕ ПРИСВАИВАНИЯ, ЛЕВАЯ ЧАСТЬ КОТОРОГО ТАКЖЕ ЯВЛЯЕТСЯ ИМЕНЕМ МАССИВА ТАКОЙ ЖЕ РАЗМЕРНОСТИ. ДОПУСКАЕТСЯ ИСПОЛЬЗОВАНИЕ В ЛЕВОЙ ЧАСТИ МАССИВА - АРГУМЕНТА. ВЫЧИСЛЯЕМОЕ ДЛЯ КАЖДОГО ЭЛЕМЕНТА МАССИВА ЗНАЧЕНИЕ ФУНКЦИИ ПРИ ЭТОМ ПРИСВАИВАЕТСЯ СООТВЕТСТВУЮЩЕМУ ЭЛЕМЕНТУ МАССИВА ЛЕВОЙ ЧАСТИ ОПЕРАТОРА.

НАПРИМЕР:

```
REAL: Z [2, 3], W [2, 3]
READ / Z
W = EXP (Z) ; Z = SIGN (Z)
WRITE / W ; Z
```

#### 4.3. СПЕЦИАЛЬНЫЕ ФУНКЦИИ ОТ МАССИВОВ.

~~~~~

В ВТ-МХТИ ОПРЕДЕЛЕНА СПЕЦИАЛЬНЫЕ ФУНКЦИИ ОТ МАССИВОВ, ИМЕЮЩИЕ СКАЛЯРНЫЕ ЗНАЧЕНИЯ. СПЕЦ. ФУНКЦИИ МОГУТ ИСПОЛЬЗОВАТЬСЯ В ЛЮБЫХ ВЫРАЖЕНИЯХ. В ИХ ЧИСЛО ВХОДЯТ:

1. СУММА ЭЛЕМЕНТОВ МАССИВА -

SUM (W)

2. СРЕДНЕЕ АРИФМЕТИЧЕСКОЕ ЭЛЕМЕНТОВ МАССИВА -

AVER (W)

3. ДИСПЕРСИЯ ЭЛЕМЕНТОВ МАССИВА -

DISP (W)

4. ЭВКЛИДОВА НОРМА МАССИВА -

NORM (W)

5. ЗНАЧЕНИЕ МАКСИМАЛЬНОГО ЭЛЕМЕНТА -

MAX (W)

6. ЗНАЧЕНИЕ МИНИМАЛЬНОГО ЭЛЕМЕНТА -

MIN (W)

ЕСЛИ ФУНКЦИИ "MAX" ИЛИ "MIN" ИСПОЛЬЗУЮТСЯ В ПРОСТЫХ ОПЕРАТОРАХ ПРИСВАИВАНИЯ, ЛЕВАЯ ЧАСТЬ КОТОРЫХ - ЦЕЛАЯ ПЕРЕМЕННАЯ, ТО ОНА ПОЛУЧАЕТ ЗНАЧЕНИЕ, РАВНОЕ НОМЕРУ МАКСИМАЛЬНОГО ИЛИ МИНИМАЛЬНОГО ЭЛЕМЕНТА ПРИ ОДНОМЕРНОЙ ИНДЕКСАЦИИ МАССИВА.

НАПРИМЕР:

```
INTEGER: IMAX, IMIN
IMAX := MAX (W) ; IMIN := MIN (W)
WRITE / IMIN ; IMAX
```

4.4. ГЕНЕРАЦИЯ СЛУЧАЙНЫХ ЧИСЕЛ.

~~~~~

ГЕНЕРАЦИЯ СЛУЧАЙНЫХ (ПСЕВДОСЛУЧАЙНЫХ) ЧИСЕЛ, РАВНОМЕРНО РАСПРЕДЕЛЕННЫХ В ИНТЕРВАЛЕ  $[0,1]$ , ПРОИЗВОДИТСЯ С ПОМОЩЬЮ ФУНКЦИИ "RAND(X)". АРГУМЕНТОМ ФУНКЦИИ ЯВЛЯЕТСЯ СЛУЧАЙНОЕ ЧИСЛО, ПЕРВОНАЧАЛЬНО ЗАДАВАЕМОЕ ПОЛЬЗОВАТЕЛЕМ И ИЗМЕНЯЮЩЕЕСЯ ПРИ ОБРАЩЕНИИ К ФУНКЦИИ. В ПРОГРАММЕ МОЖНО ОПРЕДЕЛИТЬ НЕСКОЛЬКО АРГУМЕНТОВ ДЛЯ ФУНКЦИИ "RAND", ЧТО ПОЗВОЛЯЕТ ДЛЯ КАЖДОГО ИЗ НИХ СТРОИТЬ СВОЙ РЯД СЛУЧАЙНЫХ ЧИСЕЛ.

ПРИМЕРЫ:

ПРИСВАИВАНИЕ НАЧАЛЬНОГО ЗНАЧЕНИЯ АРГУМЕНТУ ФУНКЦИИ

```
X:=SQRT(PI(1))
```

"Y" ПОЛУЧАЕТ ЗНАЧЕНИЕ СЛУЧАЙНОГО ЧИСЛА ИЗ ПОСЛЕДОВАТЕЛЬНОСТИ, РАВНОМЕРНО РАСПРЕДЕЛЕННОЙ В ИНТЕРВАЛЕ  $[-1,1]$ :

```
Y:=2*(RAND(X)-.5)
```

ВСЕ ЭЛЕМЕНТЫ МАССИВА "ST" ПОЛУЧАЮТ РАЗЛИЧНЫЕ ЗНАЧЕНИЯ СЛУЧАЙНЫХ ЧИСЕЛ:

```
INTEGER:ST[10]
```

```
ST=RAND(X)
```

ДЛЯ ГЕНЕРАЦИИ НОРМАЛЬНО РАСПРЕДЕЛЕННЫХ СЛУЧАЙНЫХ ЧИСЕЛ МОЖНО ИСПОЛЬЗОВАТЬ, НАПРИМЕР, ПОСЛЕДОВАТЕЛЬНОСТЬ ОПЕРАТОРОВ:

```
REAL:ZNORM; I:N
```

```
N:=10
```

```
REAL:SS[N]
```

```
SS:=RAND(X); ZNORM:=(SUM(SS)-N/2.)/SQRT(N/12.)
```

#### 4.5. РЕШЕНИЕ УРАВНЕНИЙ.

~~~~~

В ВТ-МХТИ ОПРЕДЕЛЕНА СПЕЦИАЛЬНАЯ ФУНКЦИЯ ДЛЯ РЕШЕНИЯ УРАВНЕНИЯ С ОДНИМ НЕИЗВЕСТНЫМ, ИМЕЮЩАЯ ТРИ АРГУМЕНТА:

```
ROOT(<ПЕРЕМЕННАЯ>,<ВЫРАЖЕНИЕ>,<МЕТКА>)
```

ИСКОМОЕ НЕИЗВЕСТНОЕ - ПЕРЕМЕННАЯ (ПРОСТАЯ ИЛИ С ИНДЕКСАМИ) ОБЯЗАТЕЛЬНО ВЕЩЕСТВЕННОГО ТИПА. ПЕРЕД ОБРАЩЕНИЕМ К "ROOT" ЕЙ ЗАДАЕТСЯ НАЧАЛЬНОЕ ПРИБЛИЖЕННОЕ ЗНАЧЕНИЕ. ЗНАЧЕНИЕ ПЕРЕМЕННОЙ УТОЧНЯЕТСЯ МЕТОДОМ СЕКУЩИХ ПО УРАВНЕНИЮ - $\langle \text{ВЫРАЖЕНИЕ} \rangle = 0$ ($\pm .1E-6$). ЕСЛИ ПОСЛЕ 50 ИТЕРАЦИЙ (ПОСЛЕДОВАТЕЛЬНЫХ ПРИБЛИЖЕНИЙ) РЕШЕНИЯ НЕ ПОЛУЧЕНО, ПРОИСХОДИТ ПЕРЕХОД НА МЕТКУ, КОТОРАЯ МОЖЕТ БЫТЬ ЗАДАНА: ЦЕЛЫМ ЧИСЛОМ, ЦЕЛОЙ ПЕРЕМЕННОЙ ИЛИ ИМЕНЕМ СИМВОЛЬНОЙ МЕТКИ.

ПРИМЕРЫ:

```
LABEL:ОШИБКА; P:F
```

```
READ/ X
```

```
WRITE/ ROOT(X,X**2-4*X+3,10)
```

```
READ/ X
```

```
WRITE/ ROOT(X,F(X),ОШИБКА)
```

```
PROGRAM F(R:XX)
```

```
RETURN(XX**2-4*XX+3)
```

```
END
```

ФУНКЦИЯ "ROOT" ЯВЛЯЕТСЯ РЕКУРСИВНОЙ, ЧТО ПОЗВОЛЯЕТ СТРОИТЬ НА ЕЕ ОСНОВЕ РЕШЕНИЕ СИСТЕМ УРАВНЕНИЙ. НАПРИМЕР:

```

REAL/ X; Y
WRITE/ ROOT (X, F1 (X, ROOT (Y, F2 (X, Y) , ОШИБКА) ) , ОШИБКА)
PROGRAM F1 (R:XX, YY)
RETURN (XX**2+YY**2-2)
END
PROGRAM F2 (R:XX, YY)
RETURN (XX**YY-1)
END
10 LABEL ОШИБКА

```

ЕЩЕ ОДНА СПЕЦИАЛЬНАЯ ФУНКЦИЯ - ПОИСК СТРОКИ СИМВОЛОВ РАССМОТРЕНА В РАЗДЕЛЕ 10.

5. ПАУЗА.

~~~~~

ДЛЯ СОЗДАНИЯ ПРОГРАММ, РАБОТАЮЩИХ В РЕАЛЬНОМ МАСШТАБЕ ВРЕМЕНИ, МОЖЕТ ОКАЗАТЬСЯ ПОЛЕЗНОЙ ВОЗМОЖНОСТЬ ПРОГРАММИРОВАТЬ ЗАДЕРЖКУ ВЫЧИСЛИТЕЛЬНОГО ПРОЦЕССА НА ЗАДАННЫЙ ПРОМЕЖУТОК ВРЕМЕНИ. ЭТА ФУНКЦИЯ РЕАЛИЗУЕТСЯ ОПЕРАТОРОМ:

```

T,s: 2.5
T,s: IMAХ
T,s: Z [1,2]

```

ВРЕМЯ ПАУЗЫ В СЕКУНДАХ И ЕЕ ДОЛЯХ ЗАДАЕТСЯ ВЕЩЕСТВЕННЫМ ИЛИ ЦЕЛЫМ ЗНАЧЕНИЕМ ЧИСЛА ИЛИ ПЕРЕМЕННЫМИ СООТВЕТСТВУЮЩЕГО ТИПА.

#### 6. АДРЕС ПЕРЕМЕННОЙ.

~~~~~

ПРИ ПРОГРАММИРОВАНИИ ФРАГМЕНТОВ ПРОГРАММЫ В КОДАХ ЭВМ МОЖЕТ ВОЗНИКНУТЬ НЕОБХОДИМОСТЬ ВЫЗОВА ЗНАЧЕНИЙ ПЕРЕМЕННЫХ В РЕГИСТРЫ ЭВМ ИЛИ ОПРЕДЕЛЕНИЯ ИХ АДРЕСОВ ДЛЯ ПОСТРОЕНИЯ ПРЯМОГО ОБРАЩЕНИЯ К НИМ В КОДАХ ПРОГРАММЫ. С ЭТОЙ ЦЕЛЬЮ МОЖНО ИСПОЛЬЗОВАТЬ ОПЕРАТОР:

```

ADR: X<
ADR: IMIN
ADR: W
ADR: Z [1,2]<

```

ГДЕ СИМВОЛ "<" УКАЗЫВАЕТ НА ВЫЗОВ ЗНАЧЕНИЯ ПЕРЕМЕННОЙ. ДЛЯ ПРОСТОЙ ВЕЩЕСТВЕННОЙ ПЕРЕМЕННОЙ МОЖНО ВЫЗВАТЬ ТОЛЬКО ЗНАЧЕНИЕ В РЕГИСТР "X" ЭВМ. ДЛЯ ВСЕХ ОСТАЛЬНЫХ ПЕРЕМЕННЫХ МОЖНО ОПРЕДЕЛИТЬ И АДРЕС И ЗНАЧЕНИЕ. ПРИ ЭТОМ:

```

ДЛЯ ПРОСТОЙ ЦЕЛОЙ ПЕРЕМЕННОЙ - АДРЕС В "R11", ЗНАЧЕНИЕ В "R8"
ДЛЯ ЭЛЕМЕНТА ЦЕЛОГО МАССИВА - АДРЕС В "R4" , ЗНАЧЕНИЕ В "R8"
ДЛЯ ЭЛЕМЕНТА ВЕЩЕСТ. МАССИВА - АДРЕС В "R4" , ЗНАЧЕНИЕ В "X"
ДЛЯ ПРОСТОЙ ВЕЩЕСТ. ПЕРЕМЕННОЙ - АДРЕС      НЕТ , ЗНАЧЕНИЕ В "X".

```

ДЛЯ ЦЕЛЫХ ПЕРЕМЕННЫХ И МАССИВОВ АДРЕСА ДАЮТСЯ ОТНОСИТЕЛЬНО НАЧАЛЬНОГО АДРЕСА "04576". ДЛЯ ВЕЩЕСТВЕННЫХ ПРОСТЫХ ПЕРЕМЕННЫХ - АБСОЛЮТНЫЕ АДРЕСА ДЕСЯТИЧНЫХ ЯЧЕЕК (ОТНОСИТЕЛЬНО АДРЕСА "04576") ДЛЯ ВЕЩЕСТВЕННЫХ И ЛИТЕРАЛЬНЫХ МАССИВОВ И ИХ ЭЛЕМЕНТОВ АДРЕСА ДАЮТСЯ ОТНОСИТЕЛЬНО НУЛЕВОГО АДРЕСА "00000". АДРЕС МАССИВА ОПРЕДЕЛЯЕТСЯ КАК АДРЕС ЕГО ПЕРВОГО ЭЛЕМЕНТА.

ОПЕРАТОР "ADR:" НЕОБХОДИМ ПРЕЖДЕ ВСЕГО, ДЛЯ ОПРЕДЕЛЕНИЯ ПРЯМЫХ АДРЕСОВ МАССИВОВ И ИХ ЭЛЕМЕНТОВ. ПРЯМЫЕ АДРЕСА ПРОСТЫХ ПЕРЕМЕННЫХ МОЖНО ОПРЕДЕЛИТЬ ПРИ ВВОДЕ ИСХОДНОГО МОДУЛЯ ДИРЕКТИВОЙ "T", ЗАДАВАЕМОЙ В ПОЗИЦИИ ВВОДА МЕТКИ ОЧЕРЕДНОГО ОПЕРАТОРА.

PP

7. ОПЕРАТОРЫ ПРИСВАИВАНИЯ.

~~~~~

В ОПЕРАТОРАХ ПРИСВАИВАНИЯ ЗНАЧЕНИЕ ВЫРАЖЕНИЯ, СТОЯЩЕГО СПРАВА ОТ ЗНАКА ПРИСВАИВАНИЯ (ПАРА СИМВОЛОВ "!="), ПОЛУЧАЕТ ПЕРЕМЕННАЯ (МАС-СИВ), ИМЯ КОТОРОЙ (КОТОРОГО) СТОИТ В ЛЕВОЙ ЧАСТИ. ОПЕРАТОРЫ ПРИСВАИВАНИЯ МОГУТ ЗАПИСЫВАТЬСЯ В ОДНУ СТРОКУ С РАЗДЕЛИТЕЛЕМ ";". В ВТ-МХТИ ЕСТЬ ДВА КЛАССА ОПЕРАТОРОВ ПРИСВАИВАНИЯ: СКАЛЯРНЫЕ И МАТРИЧНЫЕ.

### 7.1. СКАЛЯРНЫЕ ОПЕРАТОРЫ ПРИСВАИВАНИЯ.

~~~~~

В СКАЛЯРНЫХ ОПЕРАТОРАХ ПРИСВАИВАНИЯ ЗНАЧЕНИЕ ПРАВОЙ ЧАСТИ ПРЕДСТАВЛЯЕТ СОБОЙ СКАЛЯРНУЮ ВЕЛИЧИНУ (ЦЕЛУЮ ИЛИ ВЕЩЕСТВЕННУЮ), ОПРЕДЕЛЯЕМУЮ ВЫРАЖЕНИЕМ. В ОБЩЕМ СЛУЧАЕ СКАЛЯРНОЕ ВЫРАЖЕНИЕ СОСТОИТ ИЗ ОПЕРАНДОВ:

- ЧИСЛОВЫХ КОНСТАНТ;
- ПЕРЕМЕННЫХ;
- УКАЗАТЕЛЕЙ ФУНКЦИЙ;
- УКАЗАТЕЛЕЙ ОПЕРАТОР-ФУНКЦИЙ;
- УКАЗАТЕЛЕЙ ПОДПРОГРАММ-ФУНКЦИЙ;
- СКАЛЯРНЫХ ВЫРАЖЕНИЙ В КРУГЛЫХ СКОБКАХ "()", СОЕДИНЕННЫХ ЗНАКАМИ ОПЕРАЦИЙ. ВЫЧИСЛЕНИЕ ИДЕТ С УЧЕТОМ СТАРШИНСТВА ЗНАКОВ ОПЕРАЦИЙ И КРУГЛЫХ СКОБОК.

В ВТ-МХТИ ПРИНЯТ СЛЕДУЮЩИЙ НАБОР ОПЕРАЦИЙ С УЧЕТОМ ИХ СТАРШИНСТВА:

1. ^ - ОТРИЦАНИЕ ЗНАЧЕНИЯ ЦЕЛОГО ОПЕРАНДА СПРАВА. РЕЗУЛЬТАТ ЦЕЛЫЙ И РАВЕН "1", ЕСЛИ ОПЕРАНД РАВЕН "0", И РАВЕН "0" В ПРОТИВНОМ СЛУЧАЕ.
2. ** - ВОЗВЕДЕНИЕ В СТЕПЕНЬ. РЕЗУЛЬТАТ ЦЕЛЫЙ, ЕСЛИ ОСНОВАНИЕ И ПОКАЗАТЕЛЬ СТЕПЕНИ ЦЕЛЫЕ, ИНАЧЕ ВЕЩЕСТВЕННЫЕ.
3. * , / - УМНОЖЕНИЕ И ДЕЛЕНИЕ. РЕЗУЛЬТАТ УМНОЖЕНИЯ ЦЕЛЫЙ, ЕСЛИ МНОЖИМОЕ И МНОЖИТЕЛЬ ЦЕЛЫЕ, ИНАЧЕ ВЕЩЕСТВЕННЫЙ. РЕЗУЛЬТАТ ДЕЛЕНИЯ ВСЕГДА ВЕЩЕСТВЕННЫЙ.
4. - , + - ВЫЧИТАНИЕ И СЛОЖЕНИЕ. РЕЗУЛЬТАТ ЦЕЛЫЙ, ЕСЛИ ОБА ОПЕРАНДА ЦЕЛЫЕ, ИНАЧЕ ВЕЩЕСТВЕННЫЙ.
5. = , # - ОПЕРАЦИИ ОТНОШЕНИЯ "РАВНО", "НЕ РАВНО",
>= , <= "БОЛЬШЕ ИЛИ РАВНО", "РАВНО ИЛИ МЕНЬШЕ"
ОПРЕДЕЛЕНА ДЛЯ ЛЮБЫХ ТИПОВ (ВЕЩЕСТВЕННЫХ И ЦЕЛЫХ) ОПЕРАНДОВ. РЕЗУЛЬТАТ ВСЕГДА ЦЕЛЫЙ И РАВЕН "1", ЕСЛИ ОТНОШЕНИЕ ИСТИННО, И РАВЕН "0", ЕСЛИ ОТНОШЕНИЕ ЛОЖНО.

6. & - ОПЕРАЦИЯ ЛОГИЧЕСКОЕ "И" ОПРЕДЕЛЕНА ДЛЯ ЦЕЛЫХ ОПЕРАНДОВ. РЕЗУЛЬТАТ РАВЕН "0", ЕСЛИ НЕ СОВПАДАЮТ ЕДИНИЧНЫЕ БИТЫ ДВУХБАЙТОВЫХ ЦЕЛЫХ ОПЕРАНДОВ, ИНАЧЕ РАВЕН ЦЕЛОМУ ЗНАЧЕНИЮ СОВПАДАЮЩИХ ЕДИНИЧНЫХ БИТ ОПЕРАНДОВ, НАПРИМЕР:

```

ПЕРВЫЙ ОПЕРАНД - 0110 0000 1010 1011
ВТОРОЙ ОПЕРАНД - 0100 1100 0100 1111
-----
РЕЗУЛЬТАТ -      0100 0000 0000 1011

```

7. ! - ОПЕРАЦИЯ ЛОГИЧЕСКОЕ "ИЛИ" ОПРЕДЕЛЕНА ДЛЯ ЦЕЛЫХ ОПЕРАНДОВ. РЕЗУЛЬТАТ РАВЕН "0", ЕСЛИ ОБА ОПЕРАНДА РАВНЫ "0", ИНАЧЕ РАВЕН ЦЕЛОМУ ЗНАЧЕНИЮ, СОСТАВЛЕН НОМУ ИЗ ВСЕХ ЕДИНИЧНЫХ (НЕ НУЛЕВЫХ) БИТОВ ОБОИХ ОПЕРАНДОВ, НАПРИМЕР:

```

ПЕРВЫЙ ОПЕРАНД - 0110 0000 1010 1011
ВТОРОЙ ОПЕРАНД - 0100 1100 0100 1111
-----
РЕЗУЛЬТАТ-      0110 1100 1110 1111

```

В ЛЕВОЙ ЧАСТИ ОПЕРАТОРА ПРИСВАИВАНИЯ МОЖЕТ БЫТЬ УКАЗАНО ИМЯ ПЕРЕМЕННОЙ (ЦЕЛОЙ ИЛИ ВЕЩЕСТВЕННОЙ, ПРОСТОЙ ИЛИ С ИНДЕКСАМИ) ИЛИ ИМЯ МАССИВА (ЦЕЛОГО ИЛИ ВЕЩЕСТВЕННОГО). В ПОСЛЕДНЕМ СЛУЧАЕ ЗНАЧЕНИЕ ВЫРАЖЕНИЯ (ПРАВОЙ ЧАСТИ) ПОЛУЧАЮТ ВСЕ ЭЛЕМЕНТЫ МАССИВА, УКАЗАННОГО В ЛЕВОЙ ЧАСТИ. НАПРИМЕР:

```
Y:=X+1; Z:=SUM(D)/Y; F[3]:=D[1]; G[1,1]:=2.
```

ОБНУЛЕНИЕ МАССИВОВ:

```
F:=0; D:=0; G:=0
```

ЗАДАНИЕ ЗНАЧЕНИЯ "1.5" ВСЕМ ЭЛЕМЕНТАМ МАССИВА "G":

```
G:=1.5
```

ПРИМЕРЫ СЛОЖНЫХ ВЫРАЖЕНИЙ В ОПЕРАТОРАХ ПРИСВАИВАНИЯ:
ДОБАВЛЕНИЕ "1.25" К ЗНАЧЕНИЮ "Z=EXP(-2.0*X)" ПО УСЛОВИЮ "IN=FL"

```
Z:=EXP(-2.*X)+(IN=FL)*1.25
```

ПЕРЕМЕННАЯ "Y" ПОЛУЧАЕТ ЗНАЧЕНИЕ ЛИБО "0", ЛИБО "X":

```
Y:=(T+G[1,2]>=SOLT**2!IN<FL)&(SIGN(KONT)=SIGN(DIM))*X
```

ПРИМЕР РЕАЛИЗАЦИИ УСЛОВНОГО ПРИСВАИВАНИЯ:

```
T:=(FL>0)*V[1]+(FL<=0)*V[2]
```

7.2. ЗАДАНИЕ ЗНАЧЕНИЙ ЭЛЕМЕНТАМ МАССИВА.

~~~~~  
DATA: S=K:5; (M-K-1):5+EXP(-N);100.

В ОПЕРАТОРЕ ЗАДАНИЯ ЗНАЧЕНИЙ МАССИВ РАССМАТРИВАЕТСЯ КАК ОДНОМЕРНЫЙ. ЗНАЧЕНИЯ ЭЛЕМЕНТОВ МАССИВА ОПРЕДЕЛЯЮТСЯ СПИСКОМ ЗНАЧЕНИЙ, ЭЛЕМЕНТЫ КОТОРОГО ЗАДАЮТСЯ СКАЛЯРНЫМИ ВЫРАЖЕНИЯМИ. ПЕРЕД ПРИСВАЕВЫМ ЗНАЧЕНИЕМ МОЖЕТ БЫТЬ УКАЗАН ПОВТОРИТЕЛЬ: ЧИСЛО, ПРОСТАЯ ИЛИ С ИНДЕКСОМ ПЕРЕМЕННАЯ ЛЮБОГО ТИПА, ВЫРАЖЕНИЕ (В КРУГЛЫХ СКОБКАХ). ДЛЯ ПОВТОРИТЕЛЯ ВЕЩЕСТВЕННОГО ТИПА БЕРЕТСЯ ЦЕЛАЯ ЧАСТЬ. ПОВТОРИТЕЛЬ ОТДЕЛЯЕТСЯ ОТ ПРИСВАЕВОМОГО ЗНАЧЕНИЯ СИМВОЛОМ ":". ОПЕРАТОР ЗАДАНИЯ ЗНАЧЕНИЙ МОЖЕТ ПРИМЕНЯТЬСЯ ДЛЯ ПРИСВАИВАНИЯ ЗНАЧЕНИЙ ГРУППЕ ЭЛЕМЕНТОВ МАССИВА, НАЧИНАЯ С ЗАДАННОГО:

DATA: S[K]=2:17

ПРИ ВЫПОЛНЕНИИ ЭТОГО ОПЕРАТОРА ЗНАЧЕНИЕ "17" БУДЕТ ПРИСВОЕНО ТОЛЬКО ЭЛЕМЕНТАМ "S[K]" И "S[K+1]".

## 7.3. ОПЕРАЦИИ С МАССИВАМИ И МАТРИЦАМИ.

~~~~~  
В ВТ-МХТИ ОПРЕДЕЛЕН НАБОР ОПЕРАТОРОВ, ПОЗВОЛЯЮЩИЙ РЕАЛИЗОВАТЬ НЕКОТОРЫЕ ОПЕРАЦИИ С МАССИВАМИ. ЧАСТЬ ЭТИХ ОПЕРАТОРОВ ПРИМЕНИМА ТОЛЬКО К МАТРИЦАМ - МАССИВАМ С ДВУМЯ ИЗМЕРЕНИЯМИ (ДВУМЕРНЫМ). РЕЗУЛЬТАТ ОПЕРАЦИИ, ЗАДАННОЙ ПРАВОЙ ЧАСТЬЮ ОПЕРАТОРА, ПРИСВАИВАЕТСЯ МАССИВУ, ИМЯ КОТОРОГО УКАЗАНО В ЛЕВОЙ ЧАСТИ. ПРИ ЗАПИСИ ТАКИХ ОПЕРАТОРОВ ИСПОЛЮЮТСЯ ТОЛЬКО ИМЕНА МАССИВОВ СООТВЕТСТВИЕ РАЗМЕРНОСТЕЙ МАССИВОВ В ЛЕВОЙ ПРАВОЙ ЧАСТЯХ ОПЕРАТОРА КОМПИЛЯТОРОМ НЕ КОНТРОЛИРУЕТСЯ И ДОЛЖНО СОБЛЮДАТЬСЯ ПОЛЬЗОВАТЕЛЕМ. ЗА ИСКЛЮЧЕНИЕМ ОПЕРАЦИЙ ПЕРЕСЫЛОК ЭТА ГРУППА ОПЕРАТОРОВ ОПРЕДЕЛЕНА ТОЛЬКО ДЛЯ ВЕЩЕСТВЕННЫХ МАССИВОВ.

1. ПЕРЕСЫЛКА МАССИВОВ:

ЦТАБ:=ВТАБ; WT:=ЦТАБ; ВТАБ:=WT; DT:=ЦТАБ

2. УМНОЖЕНИЕ МАССИВА НА ВЕЩЕСТВЕННУЮ ПЕРЕМЕННУЮ:

WT:=ВТАБ*MULT; ВТАБ:=ВТАБ*MULT

3. ТРАНСПОНИРОВАНИЕ МАТРИЦЫ:

YT:=YN'

ЗДЕСЬ ИМЕНА МАССИВОВ ПРАВОЙ И ЛЕВОЙ ЧАСТИ ДОЛЖНЫ БЫТЬ РАЗНЫМИ.

4. ОБРАЩЕНИЕ КВАДРАТНОЙ МАТРИЦЫ:

Q:=A\$; Q:=Q\$; Q:=A\$ДЕТ

В ПОСЛЕДНЕМ ОПЕРАТОРЕ ПЕРЕМЕННОЙ "ДЕТ" ПРИСВАИВАЕТСЯ ЗНАЧЕНИЕ ОПРЕДЕЛИТЕЛЯ МАТРИЦЫ "A".

5. СЛОЖЕНИЕ И ВЫЧИТАНИЕ МАССИВОВ:

WT:=WT+ВТАБ; ВUF:=ВТАБ-WT

6. ПРОИЗВЕДЕНИЕ МАТРИЦ:

P:=YN*YT

ПЕРМНОЖЕНИЕ МАТРИЦЫ С ДИАГОНАЛЬНОЙ МОЖЕТ БЫТЬ ЗАМЕНЕНО УМНОЖЕНИЕМ НА ВЕКТОР ДИАГОНАЛЬНЫХ ЭЛЕМЕНТОВ С ПОМОЩЬЮ ОПЕРАТОРОВ:

а) . УМНОЖЕНИЕ СЛЕВА:

YND:=YD/*YN

б) . УМНОЖЕНИЕ СПРАВА:

YTD:=YT*/DY

PP

7.4. СТРОКА ОПЕРАТОРОВ.

~~~~~

В ПРОГРАММАХ ВТ-МХТИ ОПЕРАТОРЫ ПРИСВАИВАНИЯ, ОБРАЩЕНИЙ К ПРОГРАММАМ И ПРЕХОДА НА СИМВОЛЬНЫЕ МЕТКИ МОГУТ ЗАПИСЫВАТЬСЯ С РАЗДЕЛИТЕЛЕМ ";" В ОДНУ СТРОКУ, ЗАВЕРШАЕМУЮ "VK". ИМЕЕТСЯ ВОЗМОЖНОСТЬ ОРГАНИЗОВАТЬ ПОВТОРНОЕ ВЫПОЛНЕНИЕ ЛЮБОЙ ГРУППЫ ОПЕРАТОРОВ СТРОКИ. ПЕРЕД ГРУПОЙ УКАЗЫВАЕТСЯ ПОВТОРИТЕЛЬ (СКАЛЯРНОЕ ВЫРАЖЕНИЕ), ОТДЕЛЯЕМЫЙ ОТ ПОВТОРЯЕМЫХ ОПЕРАТОРОВ СИМВОЛОМ ":". ПРИ ЭТОМ ГРУППА ПОВТОРЯЕМЫХ ОПЕРАТОРОВ ЗАКЛЮЧАЕТСЯ В КРУГЛЫЕ СКОБКИ.

НАПРИМЕР:

```
REAL:A[2]; I:I; P:ТАТА; L:ТТТ
I:=1; 2:(A[I]:=I+2; ТАТА; I:=I+1)
```

ЕСЛИ ПОВТОРИТЕЛЬ ИМЕЕТ НУЛЕВОЕ ЗНАЧЕНИЕ, ТО ВЫДЕЛЕННАЯ ГРУППА ОПЕРАТОРОВ НЕ ВЫПОЛНЯЕТСЯ.

НАПРИМЕР:

```
I:=3; (X>Y):(I:=2); (X=Y):(I:=5); (X<Y):(ТТТ)
```

ДОПУСКАЕТСЯ ВЛОЖЕНИЕ ВЫДЕЛЕННЫХ ГРУПП ОПЕРАТОРОВ С СООТВЕТСТВУЮЩИМ ПОВТОРИТЕЛЕМ В КАЖДОЙ.

НАПРИМЕР:

```
I:=1;4:(J:=1;4:(B[I,J]:=I+J;(I=J):(B[I,J]:=0);J:=J+1);I:=I+1)
```

## 8. СТРОКИ КОДОВ.

~~~~~

В ЯЗЫКЕ ВТ-МХТИ ДОПУСКАЕТСЯ ИСПОЛЬЗОВАНИЕ В ЛЮБОМ МЕСТЕ ПРОГРАММЫ СРЕДИ ЛЮБЫХ ДРУГИХ ОПЕРАТОРОВ СТРОКИ КОДОВ ЭВМ, КОТОРЫЕ ПЕРЕНОСЯТСЯ КОМПИЛЯТОРОМ В РАБОЧУЮ ПРОГРАММУ БЕЗ ИЗМЕНЕНИЙ.

НАПРИМЕР:

```
[ED F8 B0 78 B1 48 4D 48 5F]
```


СТРОКИ КОДОВ (СМ. РАЗДЕЛ 10.) ФОРМИРУЮТСЯ КАК ПОСЛЕДОВАТЕЛЬНОСТЬ РАЗДЕЛЕННЫХ ПРОБЕЛАМИ ПАР ШЕСТНАДЦАТЕРИЧНЫХ ЦИФР. ДЛИНА СТРОКИ КОДОВ НЕ ОГРАНИЧЕНА. СТРОКА КОДОВ ЯВЛЯЕТСЯ РАВНОПРАВНЫМ ОПЕРАТОРОМ ПРОГРАММЫ И МОЖЕТ ТАКЖЕ ПОМЕЧАТЬСЯ ЧИСЛОВОЙ МЕТКОЙ.

PP

9. СТРУКТУРЫ

~~~~~

В ВТ-МХТИ ИСПОЛЬЗУЮТСЯ СЛЕДУЮЩИЕ СТРУКТУРНЫЕ ОПЕРАТОРЫ:

1. УСЛОВНЫЙ - "IF"...("ELSE")..."END"
2. ЦИКЛ ПОКА - "WHILE"..."END"
3. ЦИКЛ - "DO"...("END")
4. ВЫБОР - "CASE"..."END"
5. БЛОК - "BEGIN"..."END"

ВСЕ СТРУКТУРНЫЕ ОПЕРАТОРЫ ЗАВЕРШАЮТСЯ ЛЮБЫМ ИЗ ЭКВИВАЛЕНТНЫХ ОПЕРАТОРОВ: "END" И Т.Д., ПОСЛЕ КОТОРЫХ, А ТАКЖЕ ПОСЛЕ ОПЕРАТОРА "ELSE" МОЖНО ПОМЕЩАТЬ ЛЮБЫЕ КОММЕНТАРИИ.

### 9.1. УСЛОВНЫЙ "IF"

~~~~~

1. НЕПОЛНАЯ ФОРМА

```
IF X>Y
.ПОСЛЕДОВАТЕЛЬНОСТЬ ЛЮБЫХ ОПЕРАТОРОВ И СТРОК:
.( ВЫПОЛНЯЕТСЯ ЕСЛИ УСЛОВИЕ ИСТИННО )
END
```

2. ПОЛНАЯ ФОРМА

```
IF X*X-Y*Y>7
.ПОСЛЕДОВАТЕЛЬНОСТЬ ЛЮБЫХ ОПЕРАТОРОВ И СТРОК:
.(ВЫПОЛНЯЕТСЯ ЕСЛИ УСЛОВИЕ ИСТИННО)
.ELSE
.ПОСЛЕДОВАТЕЛЬНОСТЬ ЛЮБЫХ ОПЕРАТОРОВ И СТРОК:
.(ВЫПОЛНЯЕТСЯ ЕСЛИ УСЛОВИЕ ЛОЖНО)
.END
```

9.2. ЦИКЛ "WHILE"

~~~~~

```
WHILE X>Y
.ПОСЛЕДОВАТЕЛЬНОСТЬ ОПЕРАТОРОВ И СТРОК:
.(ВЫПОЛНЯЕТСЯ ПОВТОРНО, ПОКА УСЛОВИЕ ИСТИННО)
END
```

В ОБЩЕМ СЛУЧАЕ В ОПЕРАТОРАХ "IF" И "WHILE" В КАЧЕСТВЕ УСЛОВИЯ МОЖЕТ ИСПОЛЬЗОВАТЬСЯ ЛЮБОЕ СКАЛЯРНОЕ ВЫРАЖЕНИЕ, ИМЕЮЩЕЕ ЦЕЛОЕ ЗНАЧЕНИЕ. УСЛОВИЕ ИСТИННО, ЕСЛИ ЗНАЧЕНИЕ ЭТОГО ВЫРАЖЕНИЯ НЕ РАВНО НУЛЮ.

### 9.3. ЦИКЛ "DO"

~~~~~

DO ПАРАМЕТР ЦИКЛА:=НАЧ ЗНАЧЕНИЕ, КОН ЗНАЧЕНИЕ, ШАГ
.ПОСЛЕДОВАТЕЛЬНОСТЬ ОПЕРАТОРОВ И СТРОК
(ВЫПОЛНЯЕТСЯ ДЛЯ ВСЕХ ЗНАЧЕНИЙ ПАРАМЕТРА ЦИКЛА)
END ПАРАМЕТР ЦИКЛА

ЦИКЛ ЗАВЕРШАЕТСЯ ОПЕРАТОРОМ "END" ,ПОСЛЕ КОТОРОГО УКАЗЫВАЕТСЯ ПАРАМЕТР ЦИКЛА.

ПАРАМЕТР ЦИКЛА, ИЗМЕНЯЮЩИЙ ЗНАЧЕНИЕ В ОПЕРАТОРЕ ЦИКЛА "DO", ИМЕЕТ ЦЕЛЫЙ ТИП И ИЗМЕНЯЕТСЯ ОТ НАЧАЛЬНОГО ЗНАЧЕНИЯ (ПЕРВЫЙ ЭЛЕМЕНТ СПИСКА) ДО КОНЕЧНОГО (ВТОРОЙ ЭЛЕМЕНТ СПИСКА) С ПОСТОЯННЫМ ШАГОМ (ТРЕТИЙ ЭЛЕМЕНТ СПИСКА). ПРИ ОТСУТСТВИИ ТРЕТЬЕГО ЭЛЕМЕНТА ШАГ ПО УМОЛЧАНИЮ РАВЕН "1".

ЛЮБОЙ ЭЛЕМЕНТ СПИСКА ЦИКЛА МОЖЕТ ЗАДАВАТЬСЯ СКАЛЯРНЫМ ВЫРАЖЕНИЕМ - ЦЕЛЫМ ИЛИ ВЕЩЕСТВЕННЫМ (ВО ВТОРОМ СЛУЧАЕ БЕРЕТСЯ ТОЛЬКО ЦЕЛАЯ ЧАСТЬ). ЭЛЕМЕНТЫ СПИСКА МОГУТ БЫТЬ И ОТРИЦАТЕЛЬНЫМИ.

ПРИМЕРЫ :

СУММИРОВАНИЕ ЭЛЕМЕНТОВ МАССИВА

```
INTEGER:N  
READ/N  
REAL:A[N]  
READ/ A  
X:=0  
DO I:=1,N  
. X:=X+A[I]  
.ENDI
```

СУММА НЕЧЕТНЫХ ЧИСЕЛ

```
Y:=0  
DO НЕЧЕТ:=1,99,2  
. Y:=Y+НЕЧЕТ  
.END НЕЧЕТ
```

РАЗРЕШАЕТСЯ ИСПОЛЬЗОВАНИЕ ЦИКЛА "DO" С МЕТКОЙ, КОТОРЫЙ СООТВЕТСТВУЕТ АНАЛОГИЧНОМУ ОПЕРАТОРУ "FORTRAN".

ПРИМЕРЫ:

```
X:=0  
DO 10 I=1,N  
10 . X:=X+A[I]  
  
Y:=0  
DO 15 НЕЧЕТ=1,99,2  
15 . Y:=Y+НЕЧЕТ
```

9.4. ВЫБОР "CASE"

~~~~~

```
READ/НОМЕР
CASE НОМЕР
1###/*ИЗ ВСЕХ ПОСЛЕДУЮЩИХ ВЫДЕЛЕННЫХ ОПЕРАТОРОВ ВЫПОЛНЯЕТСЯ
1###/*ТОЛЬКО ОДИН С ЗАДАННЫМ НОМЕРОМ
1### Y:=25*X
2### I:=1; N:(A[I]:=I+I; I:=I+1)
3###IF(X>Y)
#.THEN
#. X:=5.75
#.ELSE
#. X:=Y+3
#.END
4###WHILE X>Y
#. X:=X-1
#.END
5###DO I:=1,N
#. A[I]:=I+X
#.END I
6### X:=2; Y:=3; ПРИМ(X,Y)
7### X:=10; ЛИПА
8###BEGIN
#:/*ПОСЛЕДОВАТЕЛЬНОСТЬ ОПЕРАТОРОВ И СТРОК
#:END
9###END
```

ВЛОЖЕНИЕ ОПЕРАТОРОВ ВЫБОРА ЗАПРЕЩЕНО.

В КАЧЕСТВЕ АРГУМЕНТА ОПЕРАТОРА "CASE" МОЖЕТ БЫТЬ ЛЮБОЕ СКАЛЯРНОЕ ВЫРАЖЕНИЕ ЦЕЛОГО ТИПА. ЕСЛИ АРГУМЕНТ РАВЕН "0", НИ ОДИН ОПЕРАТОР НЕ ВЫПОЛНЯЕТСЯ. ЕСЛИ ЖЕ АРГУМЕНТ НЕ РАВЕН "0" И НЕ СООТВЕТСТВУЕТ НИ ОДНОМУ ИЗ ОПЕРАТОРОВ ВЫБОРА, ТО ПРИ ВЫПОЛНЕНИИ ПРОГРАММЫ ДАЕТСЯ СООБЩЕНИЕ ОБ ОШИБКЕ.

#### 9.5. БЛОК "BEGIN ... END"

~~~~~

В ОБЫЧНОЙ ФОРМЕ:

```
BEGIN
:ПОСЛЕДОВАТЕЛЬНОСТЬ ОПИСАНИЙ, ОПЕРАТОРОВ, СТРОК ОПЕРАТОРОВ
:END
```

БЛОК ИСПОЛЬЗУЕТСЯ ДЛЯ ОБЪЕДИНЕНИЯ ГРУППЫ ОПЕРАТОРОВ В ОДИН В ОПЕРАТОРАХ ВЫБОРА, А ТАКЖЕ ДЛЯ ЭКОНОМИИ ПАМЯТИ, ПОСКОЛЬКУ ПОСЛЕ ЗАВЕРШЕНИЯ БЛОКА НА ОПЕРАТОРЕ "END" ПРИ РАБОТЕ ПРОГРАММЫ ПАМЯТЬ, ЗАНИМАЕМАЯ ПЕРЕМЕННЫМИ И МАССИВАМИ, ОПИСАННЫМИ В БЛОКЕ, АВТОМАТИЧЕСКИ ОСВОБАЖДАЕТСЯ. ОСВОБОЖДЕНИЕ ПАМЯТИ ПРОИСХОДИТ ТАКЖЕ И ПРИ ВЫХОДЕ ИЗ БЛОКА С ПОМОЩЬЮ ОПЕРАТОРОВ ПЕРЕХОДА, УКАЗЫВАЮЩИХ В ЯВНОМ ВИДЕ (ЧИСЛО) МЕТКУ ОПЕРАТОРА ВНЕ БЛОКА, НА КОТОРЫЙ ПЕРЕДАЕТСЯ УПРАВЛЕНИЕ. ПРИ ВЫХОДЕ ИЗ БЛОКА С ПОМОЩЬЮ НЕЯВНЫХ ОПЕРАТОРОВ ПЕРЕХОДА (СИМВОЛЬНАЯ МЕТКА, МЕТКА, ОПРЕДЕЛЯЕМАЯ ЦЕЛОЙ ПЕРЕМЕННОЙ, ПРЕРЫВАНИЕ) ОСВОБАЖДАЕТСЯ ТОЛЬКО ПАМЯТЬ, ЗАНИМАЕМАЯ ПРОСТЫМИ ПЕРЕМЕННЫМИ, А ПАМЯТЬ ОТВЕДЕННАЯ ПОД МАССИВЫ ОСТАЕТСЯ ЗАНЯТОЙ И ПРИ ПОВТОРНОМ ВХОДЕ В БЛОК ПОД ЕГО МАССИВЫ ОТВОДИТСЯ НОВАЯ ОБЛАСТЬ ПАМЯТИ.

ВХОД В БЛОК ДОПУСКАЕТСЯ ТОЛЬКО ЧЕРЕЗ ЕГО НАЧАЛО, А ТАКЖЕ НА ПОМЕЧЕННЫЕ ОПЕРАТОРЫ БЛОКА С ПОМОЩЬЮ НЕЯВНЫХ ОПЕРАТОРОВ ПЕРЕХОДА. В ПОСЛЕДНЕМ СЛУЧАЕ ОТВЕТСТВЕННОСТЬ ЗА ПРАВИЛЬНОЕ ИСПОЛЬЗОВАНИЕ ДАННЫХ, ОПИСАННЫХ В БЛОКЕ МАССИВОВ, ЛЕЖИТ НА ПОЛЬЗОВАТЕЛЕ.

В ВТ-МХТИ РЕАЛИЗОВАНЫ ЕЩЕ ДВЕ ДОПОЛНИТЕЛЬНЫЕ ФУНКЦИИ БЛОКА:

1. АВТОМАТИЧЕСКАЯ ПЕРЕДАЧА ОПИСАННЫХ В НЕМ МАССИВОВ ДАННЫХ В МАССИВ, ЗАДАННЫЙ В РАСШИРЕННОЙ ПАМЯТИ "ДЗ-28";
2. АВТОМАТИЧЕСКИЙ ВЫЗОВ ДАННЫХ ИЗ ЗАДАННЫХ МАССИВОВ РАСШИРЕННОЙ ПАМЯТИ ПРИ ВХОДЕ В БЛОК.

ПЕРВАЯ ИЗ ЭТИХ ФУНКЦИЙ ВЫПОЛНЯЕТСЯ, НАПРИМЕР, БЛОКОМ:

```
BEGIN
:REAL:AT[4],B[3,4]
/*ПОСЛЕДОВАТЕЛЬНОСТЬ ОПЕРАТОРОВ
:END#ХОУТ
```

ПРИ ЗАВЕРШЕНИИ ЭТОГО БЛОКА ДАННЫЕ МАССИВОВ "АТ" И "В" БУДУТ ПЕРЕДАНЫ В МАССИВ "ХОУТ" РАСШИРЕННОЙ ПАМЯТИ И ЗАЙМУТ В НЕМ $4+3*4=16$ ЭЛЕМЕНТОВ. МАССИВ "ХОУТ" МОЖЕТ БЫТЬ ЗАДАН ЗАРАНИЕ СООТВЕТСТВУЮЩИМ ОПИСАНИЕМ МАССИВОВ РАСШИРЕННОЙ ПАМЯТИ (COMMON#...,ХОУТ[12],...), ЛИБО БУДЕТ ОПРЕДЕЛЕН ОПЕРАТОРОМ "END#ХОУТ", ЗАВЕРШАЮЩИМ БЛОК.

ВТОРАЯ ФУНКЦИЯ (ВЫЗОВ ДАННЫХ) РЕАЛИЗУЕТСЯ, НАПРИМЕР, БЛОКОМ:

```
BEGIN#ХОУТ
:REAL:ST[4],W[3,4]
/*ПОСЛЕДОВАТЕЛЬНОСТЬ ОПЕРАТОРОВ
:END
```

ПРИ ВХОДЕ В КОТОРЫЙ В ОБЛАСТЬ ПАМЯТИ, ГДЕ БУДУТ РАЗМЕЩАТЬСЯ МАССИВЫ "ST" И "W", БУДЕТ ВЫЗВАНО СОДЕРЖИМОЕ РАСШИРЕННОЙ ПАМЯТИ ИЗ МАССИВА "ХОУТ".

ТАКИМ ОБРАЗОМ, ПРИ ПОСЛЕДОВАТЕЛЬНОМ ВЫПОЛНЕНИИ ДВУХ ЭТИХ БЛОКОВ МАССИВ "ST" ПОЛУЧИТ ЗНАЧЕНИЯ МАССИВА "АТ", А МАССИВ "W" - "В".

ВОЗМОЖНА ПОДБОРКА ДАННЫХ ИЗ НЕСКОЛЬКИХ МАССИВОВ РАСШИРЕННОЙ ПАМЯТИ В ОБЛАСТЬ РАЗМЕЩЕНИЯ МАССИВОВ БЛОКА, КАК, НАПРИМЕР, В СЛЕДУЮЩЕМ ФРАГМЕНТЕ ПРОГРАММЫ :

```
COMMON#R1[N],R2[4],R3[10]
/*ПОСЛЕДОВАТЕЛЬНОСТЬ ОПЕРАТОРОВ
BEGIN#R3,R1
:REAL:D3[10],D1[N]
/*ПОСЛЕДОВАТЕЛЬНОСТЬ ОПЕРАТОРОВ
:END
```

В ЭТОМ ПРИМЕРЕ МАССИВЫ "D1" И "D3" ПОЛУЧАТ ЗНАЧЕНИЯ МАССИВОВ РАСШИРЕННОЙ ПАМЯТИ "R1" И "R3".

10. ОБРАБОТКА СИМВОЛЬНОЙ ИНФОРМАЦИИ.

10.1. СИМВОЛЬНЫЕ ПЕРЕМЕННЫЕ.

В ПРОГРАММАХ ВТ-МХТИ КРОМЕ СПЕЦИАЛЬНО ОПИСАННЫХ СИМВОЛЬНЫХ ПЕРЕМЕННЫХ В КАЧЕСТВЕ СИМВОЛЬНЫХ МОГУТ ИСПОЛЬЗОВАТЬСЯ И ЛЮБЫЕ ДРУГИЕ ПЕРЕМЕННЫЕ (ВЕЩЕСТВЕННЫЕ И ЦЕЛЫЕ), А ТАКЖЕ И МАССИВЫ.

```
CHAR:L(20); REAL:A[3],ТАБЛ[4,6],S; INT:D[15],P
```

В ДАННОМ ОПИСАНИЕ МАССИВ "А" МОЖЕТ ИСПОЛЬЗОВАТЬСЯ КАК СИМВОЛЬНАЯ ПЕРЕМЕННАЯ НА 24 СИМВОЛА, ВЕЩЕСТВЕННАЯ ПЕРЕМЕННАЯ "S" КАК СИМВОЛЬНАЯ НА 8 СИМВОЛОВ, ЦЕЛЫЙ МАССИВ "D" КАК СИМВОЛЬНАЯ ПЕРЕМЕННАЯ НА 30 СИМВОЛОВ И ЦЕЛАЯ ПЕРЕМЕННАЯ "P" НА 2 СИМВОЛА.

ЕСЛИ В ОПЕРАЦИЯХ НАД СИМВОЛЬНЫМИ ПЕРЕМЕННЫМИ УКАЗЫВАЕТСЯ ИМЯ ПЕРЕМЕННОЙ, ТО ЭТО ОЗНАЧАЕТ, ЧТО ОПЕРАЦИИ ДОСТУПНЫ ВСЕ СИМВОЛЫ ЭТОЙ ПЕРЕМЕННОЙ, ЕСЛИ УКАЗАН ЭЛЕМЕНТ МАССИВА (ПЕРЕМЕННАЯ С ИНДЕКСОМ) ТО В ДАННОМ СЛУЧАЕ ДОСТУПНЫ ВСЕ ЭЛЕМЕНТЫ С ДАННОГО И ДО КОНЦА МАССИВА.

НАПРИМЕР:

- A[2] - СООТВЕТСТВУЕТ УКАЗАНИЮ ФРАГМЕНТА СИМВОЛЬНОЙ ПЕРЕМЕННОЙ "А", ИМЕЮЩЕГО РАЗМЕР 16 СИМВОЛОВ (2-Й И 3-Й ЭЛЕМЕНТЫ МАССИВА "А");
- ТАБЛ[3,1] - ФРАГМЕНТ, ИМЕЮЩИЙ РАЗМЕР 96 СИМВОЛОВ (ЦЕЛИКОМ 2-Я И 3-Я СТРОКИ МАТРИЦЫ "ТАБЛ" - ВСЕГО 12 ЭЛЕМЕНТОВ);
- D[10] - ФРАГМЕНТ, ИМЕЮЩИЙ РАЗМЕР 12 СИМВОЛОВ (С 10-ГО ПО 15-ЫЙ ЭЛЕМЕНТЫ МАССИВА "D"), И Т.П.

ДЛЯ ВЫДЕЛЕНИЯ ЛЮБОЙ ЧАСТИ СИМВОЛЬНОЙ ПЕРЕМЕННОЙ ИСПОЛЬЗУЕТСЯ, СИМВОЛЬНАЯ ИНДЕКСАЦИЯ - В КРУГЛЫХ СКОБКАХ ЗА ИМЕНЕМ ПЕРЕМЕННОЙ ИЛИ ЕЕ ИНДЕКСОМ УКАЗЫВАЮТСЯ ДВА ЧИСЛА, ПЕРВОЕ - НОМЕР ПЕРВОГО СИМВОЛА ФРАГМЕНТА И ВТОРОЕ - КОЛИЧЕСТВО СИМВОЛОВ В ФРАГМЕНТЕ.

НАПРИМЕР:

- L(5,4) - УКАЗЫВАЕТ ФРАГМЕНТ В 4 СИМВОЛА, НАЧИНАЯ С 5-ГО ОТ НАЧАЛА;
- A2[4](4,3) - 3 СИМВОЛА, НАЧИНАЯ С 4-ГО, ИЗ 2-ГО ЭЛЕМЕНТА МАССИВА "А";
- ТАБЛ[3,1](1,48) - ВТОРАЯ СТРОКА МАССИВА "ТАБЛ";
- S(3,4) - 4 СИМВОЛА, НАЧИНАЯ С 3-ГО, В ВЕЩЕСТВЕННОЙ ПЕРЕМЕННОЙ "S".

ЕСЛИ ВТОРОЙ ИНДЕКС В СИМВОЛЬНОЙ ИНДЕКСАЦИИ НЕ УКАЗАН, ТО ФРАГМЕНТ СОСТОИТ ИЗ ОДНОГО СИМВОЛА.

НАПРИМЕР:

- L(15) - 15-ЫЙ СИМВОЛ СИМВОЛЬНОЙ ПЕРЕМЕННОЙ "L";
- A(9) - 9-ЫЙ СИМВОЛ ОТ НАЧАЛА ИЛИ ПЕРВЫЙ СИМВОЛ 2-ГО ЭЛЕМЕНТА МАССИВА "A";
- P(1) - ПЕРВЫЙ СИМВОЛ ДВУХСИМВОЛЬНОЙ ПЕРЕМЕННОЙ "P";
- ТАБЛ[2,1](17) - 17-ЫЙ СИМВОЛ ВТОРОЙ СТРОКИ ИЛИ ПЕРВЫЙ СИМВОЛ ЭЛЕМЕНТА "ТАБЛ[2,3]" МАССИВА "ТАБЛ" И Т.П.

ЕСЛИ НЕОБХОДИМО УКАЗАТЬ ФРАГМЕНТ ОТ ОПРЕДЕЛЕННОГО СИМВОЛА ДО КОНЦА, НА МЕСТЕ ВТОРОГО СИМВОЛА МОЖНО ПОСТАВИТЬ ЗНАК "*".

НАПРИМЕР:

- L(10,*) - С 10-ГО СИМВОЛА ДО КОНЦА;
- ТАБЛ[3,1](1,*) - ДВЕ ПОСЛЕДНИЕ СТРОКИ МАТРИЦЫ "ТАБЛ" И Т.П.

В КАЧЕСТВЕ ИНДЕКСОВ ПРИ СИМВОЛЬНОЙ ИНДЕКСАЦИИ ФРАГМЕНТОВ МОГУТ ИСПОЛЬЗОВАТЬСЯ СКАЛЯРНЫЕ ВЫРАЖЕНИЯ ЛЮБОГО ТИПА.

НАПРИМЕР:

L(3*I,K+1), ТАБЛ(S*48+16,8) И Т.П.

10.2. СИМВОЛЬНЫЕ КОНСТАНТЫ.

~~~~~

В ВТ-МХТИ ИСПОЛЬЗУЮТСЯ ДВА ТИПА СИМВОЛЬНЫХ КОНСТАНТ, ДЛИНА КОТОРЫХ НЕ ДОЛЖНА ПРЕВЫШАТЬ 255 БАЙТ.

ПЕРВЫЙ - СТРОКА СИМВОЛОВ ,ЗАКЛЮЧЕННЫХ В АППОСТРОФЫ '...'

НАПРИМЕР:

'123456789АБЦД', '0-----1-----2-----3' И Т.Д.

ВТОРОЙ - СТРОКА КОДОВ (ПОСЛЕДОВАТЕЛЬНОСТЬ РАЗДЕЛЕННЫХ ПРОБЕЛАМИ ПАР ШЕСТНАДЦАТИРИЧНЫХ КОДОВ), ЗАКЛЮЧЕННЫХ В КВАДРАТНЫЕ СКОБКИ. КАЖДАЯ ПАРА КОДОВ СООТВЕТСТВУЕТ СОДЕРЖИМОМУ ОДНОГО БАЙТА, ПОЭТОМУ ЧИСЛО ПАР И ОПРЕДЕЛЯЕТ ДЛИНУ СТРОКИ В БАЙТАХ.

НАПРИМЕР:

[0С 4D 3А]

ДОПУСКАЕТСЯ СИМВОЛЬНАЯ ИНДЕКСАЦИЯ КОНСТАНТ, ЧТО ПОЗВОЛЯЕТ ФОРМИРОВАТЬ ИЗ НИХ ЛЮБЫЕ ЗАДАННЫЕ ФРАГМЕНТЫ, КОТОРЫЕ МОГУТ ПРИМЕНЯТЬСЯ ДЛЯ АНАЛИЗА СИМВОЛЬНЫХ ПЕРЕМЕННЫХ И Т.П.

НАПРИМЕР:

'123456789'(I)

ДЛЯ ЗАДАННОГО ЗНАЧЕНИЯ "I" ОПРЕДЕЛЯЕТ СООТВЕТСТВУЮЩИЙ ЭТОМУ ЗНАЧЕНИЮ СИМВОЛ.

### 10.3. ОПЕРАЦИИ НАД СИМВОЛЬНЫМИ ПЕРЕМЕННЫМИ

В ВТ-МХТИ ОПРЕДЕЛЕНА ДВЕ ОПЕРАЦИИ ПЕРЕСЫЛОК ДЛЯ СИМВОЛЬНЫХ ПЕРЕМЕННЫХ : ЛЕВАЯ ПЕРЕСЫЛКА (УПАКОВКА) И ПРАВАЯ ПЕРЕСЫЛКА (РАСПАКОВКА). СООТВЕТСТВУЮЩИЕ ОПЕРАЦИИ ИМЕЮТ ВИД:

A[2] (1,8) <= S                    - УПАКОВКА  
A => ТАБЛ[2,1] (1,24)            - РАСПАКОВКА

ПЕРВАЯ ИЗ НИХ ВЫПОЛНЯЕТ ПЕРЕСЫЛКУ ВСЕХ СИМВОЛОВ ВЕЩЕСТВЕННОЙ ПЕРЕМЕННОЙ "S" В ЭЛЕМЕНТ МАССИВА "A[2]", ВТОРАЯ - ПЕРЕСЫЛКУ МАССИВА "A" В ТРИ ПЕРВЫХ ЭЛЕМЕНТА ВТОРОЙ СТРОКИ МАССИВА "ТАБЛ"

В ПРЕДСТАВЛЕННОМ СЛУЧАЕ ИСТОЧНИКИ И ПРИЕМНИКИ ИМЕЮТ СООТВЕТСТВУЮЩИЕ РАЗМЕРЫ. ПРИ НЕ СООТВЕТСТВИИ РАЗМЕРОВ, ОБЪЕМ ПЕРЕСЫЛКИ ОПРЕДЕЛЯЕТСЯ РАЗМЕРОМ ПРИЕМНИКА. ЕСЛИ РАЗМЕР ПРИЕМНИКА МЕНЬШЕ, ТО ПЕРЕСЫЛАЕТСЯ ТОЛЬКО НАЧАЛЬНАЯ ЧАСТЬ ИСТОЧНИКА ДО ЗАПОЛНЕНИЯ ПРИЕМНИКА. ЕСЛИ НАОБОРОТ ПРИЕМНИК БОЛЬШЕ, ТО ОСТАВШАЯСЯ ЕГО ЧАСТЬ ЗАПОЛНЯЕТСЯ ПРОБЕЛАМИ.

НАПРИМЕР:

S <= '1234567890'

В ПЕРЕМЕННУЮ "S" ЗАСЫЛАЮТСЯ СИМВОЛЫ '12345678', А ПОСЛЕ ОПЕРАЦИИ:

S (3,4) <= 'AB'

СОДЕРЖИМЫМ "S" БУДЕТ СТРОКА '12AB 78'.

ОПЕРАТОРЫ ПЕРЕСЫЛКИ ДОПУСКАЮТ ИСПОЛЬЗОВАНИЕ В ПРАВЫХ ЧАСТЯХ СПИСКА СИМВОЛЬНЫХ ПЕРЕМЕННЫХ, РАЗДЕЛЕННЫХ СИМВОЛАМИ ";".

НАПРИМЕР:

A <= '12345678'; S; P; 'ABCDE'

ДЕЛАЕТ УПАКОВКУ В МАССИВ "A" СОДЕРЖИМОГО КОНСТАНТ И ПЕРЕМЕННЫХ СПИСКА ПРАВОЙ ЧАСТИ, А ОПЕРАТОР:

ТАБЛ[3,1] (1,32) => A; S

ПРОИЗВОДИТ РАСПАКОВКУ 4-Х ПЕРВЫХ ЭЛЕМЕНТОВ 3-Й СТРОКИ МАССИВА "ТАБЛ" В МАССИВ "A" И ПРОСТУЮ ПЕРЕМЕННУЮ "S".

В ОПЕРАТОРЕ ЛЕВОЙ ПЕРЕСЫЛКИ ПЕРЕД ЭЛЕМЕНТАМИ СПИСКА ПРАВОЙ ЧАСТИ МОГУТ УКАЗЫВАТЬСЯ ПОВТОРИТЕЛИ, ОТДЕЛЕННЫЕ ОТ СЛЕДУЮЩЕГО ЭЛЕМЕНТА СИМВОЛОМ ":". В КАЧЕСТВЕ ПОВТОРИТЕЛЯ МОЖНО ИСПОЛЬЗОВАТЬ КОНСТАНТЫ, ПЕРЕМЕННЫЕ И ВЫРАЖЕНИЯ ЛЮБОГО ТИПА.

В СПИСОК ПРАВОЙ ЧАСТИ МОГУТ ВКЛЮЧАТЬСЯ ОПЕРАТОРЫ ПРИСВАИВАНИЯ, ЧТО ПОЗВОЛЯЕТ ВЫЧИСЛЯТЬ ЗНАЧЕНИЯ ПОВТОРИТЕЛЕЙ НЕПОСРЕДСТВЕННО ПРИ ВЫПОЛНЕНИИ ОПЕРАЦИИ ПЕРЕСЫЛКИ.

НАПРИМЕР:

A <= 8: '='; K: =16/2; K: '/'; K: '-'

В МАССИВЕ "A" БУДЕТ ЗАПИСАНА СТРОКА '=====/- - - - -'.

ДОПУСКАЕТСЯ ТАКЖЕ ИСПОЛЬЗОВАНИЕ ПОВТОРИТЕЛЕЙ ДЛЯ ГРУПП ЭЛЕМЕНТОВ СПИСКА, ЗАКЛЮЧЕННЫХ В КРУГЛЫЕ СКОБКИ, С ПРОИЗВОЛЬНЫМ УРОВНЕМ ВЛОЖЕНИЯ.

НАПРИМЕР:

A <= K: =1; 5: (K: '\*' ; '/'; K: =K+1) ; 'ABCD'

СФОРМИРУЕТ В МАССИВЕ "A" СТРОКУ '\*/\*\*/\*\*\*\*/\*\*\*\*\*/ABCD'

#### 10.4. РЕАЛИЗАЦИЯ СИМВОЛЬНОГО ПОИСКА

~~~~~

ПОИСК СИМВОЛЬНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ "СИМВ1", ЗАДАННОЙ СИМВОЛЬНОЙ ПЕРЕМЕННОЙ, ФРАГМЕНТОМ ИЛИ КОНСТАНТОЙ, В ДРУГОЙ СИМВОЛЬНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ "СИМВ2" ВЫПОЛНЯЕТСЯ СТАНДАРТНОЙ ФУНКЦИЕЙ:

@СИМВ1=СИМВ2

ПРИНИМАЮЩЕЙ ЦЕЛОЕ ЗНАЧЕНИЕ, РАВНОЕ НОМЕРУ БАЙТА В "СИМВ2", С КОТОРОГО В НЕМ НАЧИНАЕТСЯ ПОСЛЕДОВАТЕЛЬНОСТЬ "СИМВ1". ПРИ ОТСУТСТВИИ ВЛОЖЕНИЯ ДАННОЙ ПОСЛЕДОВАТЕЛЬНОСТИ ФУНКЦИЯ ПРИНИМАЕТ ЗНАЧЕНИЕ "0".

НАПРИМЕР, ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА:

К:=@'CD'='АВСВСДЕАВСДФЕ'

ПЕРЕМЕННАЯ "К" ПОЛУЧИТ ЗНАЧЕНИЕ 5, А ПОСЛЕ ОПЕРАТОРА:

К:=@'CDF'='АВСВСДЕАВСДФЕ'

ЗНАЧЕНИЕ 10

ФУНКЦИЯ "@" МОЖЕТ ИСПОЛЬЗОВАТЬСЯ В ДРУГОЙ ФОРМЕ ДЛЯ ПОИСКА В ПОСЛЕДОВАТЕЛЬНОСТИ СИМВОЛОВ ПЕРВОГО ОТЛИЧНОГО ОТ ЗАДАННОГО СИМВОЛА.

@'СИМВОЛ'#СИМВ2

ЭТА ФУНКЦИЯ РАВНА "0" ТОЛЬКО В ТОМ СЛУЧАЕ, ЕСЛИ ВСЯ "СИМВ2" СОСТОИТ ИЗ СИМВОЛОВ, ТОЖДЕСТВЕННЫХ ЗАДАННОМУ.

НАПРИМЕР, ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА:

К:=@'%'#'%%%%%%%%123456'

ЗНАЧЕНИЕ "К" БУДЕТ РАВНО "8".

PP

11. ВВОД/ВЫВОД ИНФОРМАЦИИ.

~~~~~

##### 11.1. ОПЕРАТОРЫ ВВОДА/ВЫВОДА

~~~~~

В ВТ-МХТИ ОБЕСПЕЧЕН ДОСТАТОЧНО ШИРОКИЙ НАБОР ФУНКЦИОНАЛЬНЫХ ВОЗМОЖНОСТЕЙ ВВОДА/ВЫВОДА ДАННЫХ В РАЗЛИЧНЫХ ФОРМАТАХ.

СРЕДСТВАМИ ОПЕРАЦИОННОЙ СИСТЕМЫ ОБЕСПЕЧИВАЕТСЯ ВВОД/ВЫВОД ПО ТРЕМ СИСТЕМНЫМ КАНАЛАМ:

- 1 - ДИСПЛЕЙ
- 2 - МАГНИТНАЯ ЛЕНТА
- 3 - ПЕЧАТЬ

ОСТАЛЬНЫЕ КАНАЛЫ С 5 ПО 255 МОГУТ ИСПОЛЬЗОВАТЬСЯ ДЛЯ ОБМЕНА ИНФОРМАЦИЕЙ С АДРЕСУЕМЫМИ (УПР РАВЕН НОМЕРУ КАНАЛА) ВНЕШНИМИ УСТРОЙСТВАМИ.

ДОПУСТИМЫ СЛЕДУЮЩИЕ ФОРМЫ ОПЕРАТОРОВ ВЫВОДА - "WRITE" ИЛИ ВВОДА - "READ":

1. WRITE!READ <СПИСОК ОБЪЕКТОВ>
2. WRITE!READ (<НОМЕР КАНАЛА>)<СПИСОК ОБЪЕКТОВ>
3. WRITE!READ (<НОМЕР КАНАЛА>,<МЕТКА ФОРМАТА>)<СПИСОК ОБЪЕКТОВ>
4. WRITE!READ (<ФОРМАТ>)<СПИСОК ОБЪЕКТОВ>
5. WRITE!READ (<НОМЕР КАНАЛА>,<ФОРМАТ>)<СПИСОК ОБЪЕКТОВ>

ОПЕРАТОРЫ ВВОДА/ВЫВОДА БЕЗ УКАЗАНИЯ НОМЕРА КАНАЛА ОБЕСПЕЧИВАЮТ ОБМЕН ИНФОРМАЦИЕЙ С ДИСПЛЕЕМ (КАНАЛ 1 ИЛИ 241). ДЛЯ УКАЗАНИЯ НОМЕРА КАНАЛА В ОСТАЛЬНЫХ СЛУЧАЯХ МОЖНО ИСПОЛЬЗОВАТЬ ЦЕЛОЕ ЧИСЛО ИЛИ ЦЕЛУЮ ПЕРЕМЕННУЮ, ПРОСТУЮ ИЛИ С ИНДЕКСАМИ.

11.2. СПИСОК ОБЪЕКТОВ ВВОДА/ВЫВОДА.

~~~~~

ОБЪЕКТАМИ ВВОДА/ВЫВОДА МОГУТ БЫТЬ ИМЕНА ПЕРЕМЕННЫХ, ПРОСТЫХ И С ИНДЕКСАМИ, ИМЕНА МАССИВОВ, А ТАКЖЕ ПЕРЕМЕННЫЕ С СИМВОЛЬНОЙ ИНДЕКСАЦИЕЙ. КРОМЕ ТОГО, В СПИСКЕ ВВОДА МОГУТ БЫТЬ СИМВОЛЬНЫЕ КОНСТАНТЫ, ВОЗМОЖНО ИНДЕКСИРОВАННЫЕ, КОТОРЫЕ ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА ВВОДА ОТРАБАТЫВАЮТСЯ НА ВЫВОД, ПОДОБНО СИМВОЛЬНЫМ КОНСТАНТАМ В ФОРМАТЕ. КРОМЕ ТОГО В ЛЮБОМ МЕСТЕ СПИСКА МОЖНО ИСПОЛЬЗОВАТЬ СИМВОЛ ПЕРЕВОДА СТРОКИ "/" И "ПРОБЕЛ", А ТАКЖЕ УКАЗАТЕЛЬ МАРКЕРА ДЛЯ УСТАНОВКИ ОПРЕДЕЛЕННЫХ ПОЗИЦИЙ ВВОДА ОБЪЕКТОВ НА ЭКРАНЕ.

ЕСЛИ ОБЪЕКТОМ ВВОДА ЯВЛЯЕТСЯ ИМЯ МАССИВА, ТО РЕАЛИЗУЕТСЯ ВВОД ВСЕХ ЕГО ЭЛЕМЕНТОВ, УПРАВЛЯЕМЫЙ СООТВЕТСТВУЮЩЕЙ ЧАСТЬЮ ФОРМАТА ПРИ ОТСУТСТВИИ ФОРМАТА ПРОИЗВОДИТСЯ ПОСЛЕДОВАТЕЛЬНЫЙ ВВОД МАССИВА В ОДНОЙ СТРОКЕ ДИСПЛЕЯ. ПРИ ЖЕЛАНИИ МОЖНО РЕАЛИЗОВАТЬ ВВОД ОЧЕРЕДНОГО ЭЛЕМЕНТА СО СЛЕДУЮЩЕЙ СТРОКИ, ДЛЯ ЧЕГО ВВОД ПРЕДШЕСТВУЮЩЕГО ЕМУ ЭЛЕМЕНТА СЛЕДУЕТ ЗАВЕРШИТЬ НЕ СИМВОЛОМ "VK", А СИМВОЛОМ "PC".

ОБЪЕКТАМИ СПИСКА ВЫВОДА, В ДОПОЛНЕНИЕ К ПЕРЕЧИСЛЕННЫМ ВЫШЕ, МОГУТ БЫТЬ ПРОИЗВОЛЬНЫЕ СКАЛЯРНЫЕ ВЫРАЖЕНИЯ, ВЫЧИСЛЕНИЕ КОТОРЫХ ПРОИЗВОДИТСЯ В ПРОЦЕССЕ ВЫВОДА.

ОБЪЕКТЫ ВВОДА/ВЫВОДА В СПИСКЕ РАЗДЕЛЯЮТСЯ СИМВОЛОМ ";". КРОМЕ ТОГО МЕЖДУ ОБЪЕКТАМИ ВВОДА/ВЫВОДА В ЛЮБОМ МЕСТЕ СПИСКА МОГУТ ВКЛЮЧАТЬСЯ КАК ОТДЕЛЬНЫЕ ОПЕРАТОРЫ ПРИСВАИВАНИЯ, ТАК И ИХ ГРУППЫ (СТРОКИ), ВКЛЮЧАЯ КОНСТРУКЦИИ С ПОВТОРИТЕЛЯМИ, ОБЪЕКТЫ ВВОДА/ВЫВОДА МОГУТ РАЗМЕЩАТЬСЯ ВНУТРИ ЭТИХ КОНСТРУКЦИЙ, ЧТО ПОЗВОЛЯЕТ ВЫПОЛНЯТЬ ПРИ ВВОДЕ/ВЫВОДЕ РАЗЛИЧНЫЕ ВЫЧИСЛЕНИЯ, А ТАКЖЕ ОРГАНИЗОВЫВАТЬ ЦИКЛИЧЕСКИЕ ОПЕРАЦИИ ВВОДА/ВЫВОДА В СОЧЕТАНИИ С ШИРОКИМИ ВОЗМОЖНОСТЯМИ ПОЗИЦИОНИРОВАНИЯ НА ЭКРАНЕ ДИСПЛЕЯ ИЛИ УСТРОЙСТВА ПЕЧАТИ.

В "VT-MXTI" РЕАЛИЗОВАНА ВОЗМОЖНОСТЬ СМЕНЫ РЕЖИМА ОПЕРАТОРА ВВОДА НА ВЫВОД И НА ОБОРОТ. С ЭТОЙ ЦЕЛЬЮ В СПИСКЕ ВВОДА/ВЫВОДА ИСПОЛЬЗУЮТСЯ СИМВОЛЫ:

- < - ПЕРЕХОД ОТ ВВОДА К ВЫВОДУ;
- > - ПЕРЕХОД ОТ ВЫВОДА К ВВОДУ.

ПРИ НАЛИЧИИ СИМВОЛА ВВОДА ">" В СПИСКЕ ОБЪЕКТОВ ОПЕРАТОРА ВЫВОДА ОБЪЕКТЫ СПИСКА, РАСПОЛОЖЕННЫЕ ДО СИМВОЛА ">", ПОДЛЕЖАТ ВЫВОДУ, ПОСЛЕ СИМВОЛА ">" - ВВОДУ. ВОЗМОЖНО ТАКЖЕ МНОГОКРАТНАЯ СМЕНА РЕЖИМОВ ВВОДА И ВЫВОДА В ПРЕДЕЛАХ ОДНОГО СПИСКА.

ПРИ ВВОДЕ ЧИСЛОВОЙ ИНФОРМАЦИИ ВОЗМОЖНО ИСПРАВЛЕНИЕ ВВЕДЕННОЙ ЧАСТИ ЧИСЛА С ПОМОЩЬЮ СЛУЖЕБНОЙ КЛАВИШИ ДИСПЛЕЯ " <- ". АНАЛОГИЧНЫМ СПОСОБОМ ВНОСЯТСЯ ИСПРАВЛЕНИЯ ПРИ СИМВОЛЬНОМ ВВОДЕ В БАЙТОВОМ РЕЖИМЕ. В ГРУППОВОМ РЕЖИМЕ СИМВОЛЬНОГО ВВОДА ИСПРАВЛЕНИЯ НЕ ПРЕДУСМОТРЕНЫ. МОЖНО ТАКЖЕ ПОЛНОСТЬЮ ПОВТОРИТЬ ВЕСЬ ОПЕРАТОР ВВОДА, ЕСЛИ ПРИ ВВОДЕ ЛЮБОГО ЧИСЛОВОГО ОБЪЕКТА СПИСКА ВВОДА ВВЕСТИ СИМВОЛ "/". НАЖАТИЕ КЛАВИШИ "СБР" ПРИ ВВОДЕ ПРИВОДИТ К ПРЕКРАЩЕНИЮ ВЫПОЛНЕНИЯ ПРОГРАММЫ И ВЫХОДУ НА СОСТОЯНИЕ "МРП".

ПРИ ПОСТРОЕНИИ ОПЕРАТОРОВ ВВОДА/ВЫВОДА ДОПУСКАЕТСЯ БОЛЬШОЙ ВЫБОР ВОЗМОЖНОСТЕЙ ЗА СЧЕТ ИСПОЛЬЗОВАНИЯ ВСТРОЕННЫХ ИЛИ ВНЕШНИХ ФОРМАТОВ С РАЗНООБРАЗНОЙ СТРУКТУРОЙ И УПРАВЛЯЮЩИХ ОБЪЕКТОВ И СТРУКТУР СПИСКОВ ВВОДА/ВЫВОДА. В ПРЕДСТАВЛЕННЫХ ВЫШЕ ПРИМЕРАХ РАССМОТРЕНЫ ЛИШЬ ОСНОВНЫЕ ПРИЕМЫ ПОСТРОЕНИЯ ОПЕРАТОРОВ ВВОДА/ВЫВОДА, НА БАЗЕ КОТОРЫХ СТРОЯТСЯ И БОЛЕЕ СЛОЖНЫЕ ОПЕРАТОРЫ ФОРМИРОВАНИЯ ТАБЛИЦ, ГРАФИКОВ И Т. П.

### 11.3. ФОРМАТ ВВОДА/ВЫВОДА

~~~~~

ФОРМАТ ВВОДА/ВЫВОДА ЗАДАЕТСЯ ЛИБО ОТДЕЛЬНЫМ ПОМЕЧЕННЫМ ОПЕРАТОРОМ "ФОРМАТ", МЕТКА КОТОРОГО УКАЗЫВАЕТСЯ В ОПЕРАТОРЕ ВВОДА/ВЫВОДА ПОСЛЕ ОБЯЗАТЕЛЬНОГО В ТАКИХ СЛУЧАЯХ НОМЕРА КАНАЛА ЛИБО НЕПОСРЕДСТВЕННО ВСТРАИВАЕТСЯ В ОПЕРАТОР ВВОДА/ВЫВОДА.

ПРИ ОТСУТСТВИИ ФОРМАТА В ОПЕРАТОРЕ ВВОДА/ВЫВОДА ПРОИЗВОДИТСЯ АВТОМАТИЧЕСКАЯ РАСПЕЧАТКА ИМЕН ОБЪЕКТОВ СПИСКА ВВОДА/ВЫВОДА ПРИ ВЫПОЛНЕНИИ ЭТИХ ОПЕРАТОРОВ. ВЫВОД ЧИСЛОВЫХ ДАННЫХ ПРИ ЭТОМ ПРОИЗВОДИТСЯ В СТАНДАРТНОМ ФОРМАТЕ "Е5". ВВОД/ВЫВОД СИМВОЛЬНЫХ ДАННЫХ ОБЕСПЕЧИВАЕТСЯ ПРИ НАЛИЧИИ СИМВОЛЬНОЙ ИНДЕКСАЦИИ У ОБЪЕКТА, ПРЕДСТАВЛЯЮЩЕГО СИМВОЛЬНУЮ ПЕРЕМЕННУЮ. СИМВОЛЬНЫЕ КОНСТАНТЫ КАК В ОПЕРАТОРАХ ВЫВОДА, ТАК И ВВОДА АВТОМАТИЧЕСКИ ОФОРМЛЯЮТСЯ НА ВЫВОД.

В СТРУКТУРЕ ФОРМАТА ОПРЕДЕЛЕННЫ СЛЕДУЮЩИЕ СПЕЦИФИКАЦИИ ВЫВОДА ДАННЫХ:

1. ЭКСПОНЕНЦИАЛЬНЫЙ ФОРМАТ "Е" ИЛИ "Е<ЦЕЛОЕ, МЕНЬШЕЕ 10>". ОПРЕДЕЛЯЕТ ВЫВОД МАНТИССЫ С ЗАДАННЫМ ЧИСЛОМ РАЗРЯДОВ (В ФОРМАТЕ Е - 12) И С ДВУЗНАЧНЫМ ПОРЯДКОМ 10.
2. ЦЕЛЫЙ ФОРМАТ "I<ЦЕЛОЕ, МЕНЬШЕЕ 10>". ОПРЕДЕЛЯЕТ ВЫВОД ЦЕЛОЙ ЧАСТИ ЧИСЛА СО ЗНАКОМ С УКАЗАННЫМ ЧИСЛОМ ПОЗИЦИЙ - НЕЗНАЧАЩИЕ НУЛИ ПОДАВЛЯЮТСЯ.
3. ЦЕЛЫЙ ФОРМАТ "Z< ЦЕЛОЕ МЕНЬШЕ 10>". ОПРЕДЕЛЯЕТ ВЫВОД ЦЕЛОЙ ЧАСТИ ЧИСЛА С УКАЗАННЫМ ЧИСЛОМ ПОЗИЦИЙ БЕЗ ЗНАКА - НЕЗНАЧАЩИЕ НУЛИ ПОДАВЛЯЮТСЯ.
4. ФОРМАТ С ФИКСИРОВАННОЙ ТОЧКОЙ "F<ЦЕЛОЕ>.<ЦЕЛОЕ>". ОПРЕДЕЛЯЕТ ВЫВОД ЦЕЛОЙ И ДРОБНОЙ ЧАСТИ ЧИСЛА С УКАЗАННЫМ ЧИСЛОМ РАЗРЯДОВ В КАЖДОЙ -НЕЗНАЧАЩИЕ НУЛИ ЦЕЛОЙ ЧАСТИ ПОДАВЛЯЮТСЯ.
5. ПЕРЕМЕННЫЙ ФОРМАТ С ФИКСИРОВАННОЙ ТОЧКОЙ "F(ВЕЩ.ПЕРЕМ)" ЧИСЛА РАЗРЯДОВ ЦЕЛОЙ И ДРОБНОЙ ЧАСТИ ЗАДАЮТСЯ ЦЕЛОЙ ЧАСТЬЮ И ЧИСЛОМ ДЕСЯТЫХ ДОЛЕЙ ВЕЩЕСТВЕННОЙ ПЕРЕМЕННОЙ.

ВСЕ ПЕРЕЧИСЛЕННЫЕ СПЕЦИФИКАЦИИ ИДЕНТИЧНЫ ПРИ ВВОДЕ ЧИСЛОВЫХ ДАННЫХ. ЕСЛИ ПРИ ВЫВОДЕ ЗНАЧЕНИЕ НЕ РАЗМЕЩАЕТСЯ В ФОРМАТАХ "I", "Z", "F", ТО АВТОМАТИЧЕСКИ ЗАДАЕТСЯ ВЫВОД В ФОРМАТЕ "E5".

6. ФОРМАТ СИМВОЛЬНОГО ВВОДА/ВЫВОДА "A" ИЛИ "A<ЦЕЛОЕ>". ИСПОЛЬЗУЕТСЯ ДЛЯ СПЕЦИФИКАЦИИ ВВОДА/ВЫВОДА ОБЪЕКТОВ В СИМВОЛЬНОЙ ФОРМЕ. ЗНАЧЕНИЕ ЦЕЛОГО ОПРЕДЕЛЯЕТ РАЗМЕР ФРАГМЕНТА ОБЪЕКТА, ЗАДАННЫЙ ДЛЯ ВВОДА/ВЫВОДА. БЕЗ УКАЗАНИЯ РАЗМЕРА ВВОД/ВЫВОД ОПРЕДЕЛЯЕТСЯ РАЗМЕРАМИ ОБЪЕКТА. ЕСЛИ ВВОД СИМВОЛОВ ПРЕКРАЩАЕТСЯ СИМВОЛОМ "BK" ДО ИСЧЕРПАНИЯ ЗАДАННОГО РАЗМЕРА, ОСТАВШАЯСЯ ЧАСТЬ АВТОМАТИЧЕСКИ ЗАПОЛНЯЕТСЯ СИМВОЛАМИ "ПРОБЕЛ".

ПРИ ВЫПОЛНЕНИЕ СИМВОЛЬНОГО ВВОДА/ВЫВОДА ВОЗМОЖНЫ ДВА РЕЖИМА: СИМВОЛЬНЫЙ И ГРУПОВОЙ.

СИМВОЛЬНЫЙ РЕЖИМ ЯВЛЯЕТСЯ ОСНОВНЫМ ПРИ РАБОТЕ С ДИСПЛЕЕМ И ПЕЧАТЬЮ. В ЭТОМ РЕЖИМЕ ВВОД/ВЫВОД ПРОИЗВОДИТСЯ ПО ОДНОМУ СИМВОЛУ. ПРИ ВВОДЕ С ДИСПЛЕЯ ВЫРАБАТЫВАЕТСЯ ОТВЕТ, ОТОБРАЖАЮЩИЙ ВВЕДЕННЫЙ СИМВОЛ НА ЭКРАНЕ (ДУПЛЕКСНЫЙ РЕЖИМ), ПОСЛЕ ЧЕГО ВЫПОЛНЯЕТСЯ ПРЕОБРАЗОВАНИЕ ВВЕДЕННОГО СИМВОЛА В КОД "КОИ-8", ПОЗВОЛЯЮЩИЙ РЕАЛИЗОВАТЬ ВСЬ ВОЗМОЖНЫЙ НАБОР СИМВОЛОВ ДИСПЛЕЯ:

- ПРОПИСНЫЕ И СТРОЧНЫЕ РУССКИЕ И ЛАТИНСКИЕ БУКВЫ, ЦИФРЫ И ВСЕ ОСТАЛЬНЫЕ ЗНАКИ ОСНОВНОЙ КЛАВИАТУРЫ ДИСПЛЕЯ.

ПРИ ВЫВОДЕ НА ДИСПЛЕЙ ПРОИЗВОДИТСЯ ПРЕОБРАЗОВАНИЕ ВЫВОДИМЫХ СИМВОЛОВ В КОД КОИ-7Н2 ДИСПЛЕЯ С ВЫРАБОТКОЙ НЕОБХОДИМЫХ ДОПОЛНИТЕЛЬНЫХ СИГНАЛОВ ПЕРЕКЛЮЧЕНИЯ РЕГИСТРОВ (РУС, ЛАТ).

ВЫВОД НА ПЕЧАТЬ ПРОИЗВОДИТСЯ НЕПОСРЕДСТВЕННО В КОДЕ КОИ-8 С ПРЕДВАРИТЕЛЬНОЙ ВЫДАЧЕЙ НА УСТРОЙСТВО ПЕЧАТИ УВВПЧ-30-004 КОМАНДНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ, УСТАНОВЛИВАЮЩЕЙ ЭТО УСТРОЙСТВО В РЕЖИМ ПЕЧАТИ КОДОВ КОИ-8. ПРИ ИСПОЛЬЗОВАНИЕ В КАЧЕСТВЕ УСТРОЙСТВ ПЕЧАТИ ТПУ, CONSUL-60.1, И ДР. РАБОТАЮЩИХ С КОДАМИ "КОИ-7Н2", СТРОЧНЫЕ ЛАТИНСКИЕ БУКВЫ ОТОБРАЖАЕТСЯ ПРОПИСНЫМИ РУССКИМИ, А СТРОЧНЫЕ РУССКИЕ - ПРОПИСНЫМИ ЛАТИНСКИМИ.

ГРУПОВОЙ РЕЖИМ РЕАЛИЗУЕТ МАКСИМАЛЬНУЮ СКОРОСТЬ ОБМЕНА СИМВОЛЬНОЙ ИНФОРМАЦИЕЙ, КОТОРАЯ ПЕРЕДАЕТСЯ ПОБАЙТНО БЕЗ КАКОГО-ЛИБО ПРЕОБРАЗОВАНИЯ. ПРИ ВВОДЕ С ДИСПЛЕЯ ИНФОРМАЦИЯ ПЕРЕДАЕТСЯ БЕЗ ОТОБРАЖЕНИЯ НА ЭКРАНЕ. ЭТОТ РЕЖИМ ЯВЛЯЕТСЯ ОСНОВНЫМ ПРИ ОБМЕНЕ ИНФОРМАЦИЕЙ С ВНЕШНИМИ КАНАЛАМИ.

ДЛЯ УСТАНОВКИ ГРУПОВОГО РЕЖИМА ОБМЕНА В ФОРМАТЕ ОПЕРАТОРА ВВОДА/ВЫВОДА К СПЕЦИФИКАЦИИ СИМВОЛЬНОГО ОБМЕНА "A<ЦЕЛОЕ>" ИЛИ "A", ДОБАВЛЯЕТСЯ СИМВОЛ "@". ЕСЛИ ОПЕРАТОР НЕ ИМЕЕТ ФОРМАТА, ТО РЕЖИМ СИМВОЛЬНОГО ОБМЕНА УСТАНОВЛИВАЕТСЯ ЗАПИСЬЮ СИМВОЛА "@" НЕПОСРЕДСТВЕННО ПОСЛЕ СИМВОЛЬНОЙ ПЕРЕМЕННОЙ В СПИСКЕ ВВОДА/ВЫВОДА, СИМВОЛЬНЫЕ КОНСТАНТЫ ТИПА СТРОКИ КОДОВ - [ПАРЫ ШЕСТНАДЦАТИРИЧНЫХ КОДОВ] - ВСЕГДА ВЫВОДЯТСЯ В ГРУППОВОМ РЕЖИМЕ.

7. ФОРМАТ ОБМЕНА С МАГНИТНОЙ ЛЕНТОЙ "T<ЦЕЛОЕ>" ИСПОЛЬЗУЕТСЯ ТОЛЬКО ДЛЯ КАНАЛА "2". ЗНАЧЕНИЕ ЦЕЛОГО ОПРЕДЕЛЯЕТ НОМЕР ЗАПИСЫВАЕМОГО ИЛИ СЧИТЫВАЕМОГО ФАЙЛА ДАННЫХ НА МЛ.

В ФОРМАТАХ СИМВОЛЬНОГО ВВОДА/ВЫВОДА И ОБМЕНА С МАГНИТНОЙ ЛЕНТОЙ ДЛЯ ЗАДАНИЯ ЦЕЛОГО МОГУТ ИСПОЛЬЗОВАТЬСЯ ПЕРЕМЕННЫЕ (ФОРМАЛЬНЫЕ ПАРАМЕТРЫ ПОДПРОГРАМ В КАЧЕСТВЕ ПОВТОРИТЕЛЕЙ ЗАПРЕЩЕНО).

КРОМЕ СПЕЦИФИКАЦИЙ В ФОРМАТАХ РАЗРЕШЕНО ИСПОЛЬЗОВАТЬ СИМВОЛЫ ПЕРЕВОДА СТРОКИ "/", ПРОБЕЛА "X", И СИМВОЛЬНЫЕ КОНСТАНТЫ, КОТОРЫЕ ТАКЖЕ МОГУТ ИМЕТЬ ПОВТОРИТЕЛИ. ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА (КАК ВВОДА ТАК И ВВОДА) ЭТИ ОБЪЕКТЫ ВСЕГДА ОБРАБАТЫВАЮТСЯ В РЕЖИМЕ ВЫВОДА, ЧТО ПОЗВОЛЯЕТ С ИХ ПОМОЩЬЮ СТРОИТЬ УДОБНЫЕ ПРЕДСТАВЛЕНИЯ ПРОЦЕССОВ ВВОДА/ВЫВОДА.

В СТРУКТУРЕ ФОРМАТА МОЖНО ПРИМЕНЯТЬ ОБЪЕДИНЕНИЕ ОТДЕЛЬНЫХ ЧАСТЕЙ КРУГЛЫМИ СКОБКАМИ "()" С СООТВЕТСТВУЮЩИМИ ПОВТОРИТЕЛЯМИ, ПРИЧЕМ ДОПУСКАЕТСЯ ПРОИЗВОЛЬНЫЙ УРОВЕНЬ ВЛОЖЕНИЯ ТАКИХ КОНСТРУКЦИЙ. ПРИ ИСЧЕРПАНИИ СПЕЦИФИКАЦИИ ФОРМАТА СПИСКОМ ОБЪЕКТОВ ВВОДА/ВЫВОДА, ВОЗОБНОВЛЕНИЕ ДЕЙСТВИЙ СПЕЦИФИКАЦИЙ ПРОИЗВОДИТСЯ ОТ ПОСЛЕДНЕЙ ОТКРЫВАЮЩЕЙ СКОБКИ.

ДЛЯ РАСШИРЕНИЯ ВОЗМОЖНОСТЕЙ ПОЗИЦИОНИРОВАНИЯ ВВОДА/ВЫВОДА НА ЭКРАНЕ ДИСПЛЕЯ (ТОЛЬКО КАНАЛ 1) ИСПОЛЬЗУЕТСЯ УКАЗАТЕЛЬ ПОЗИЦИИ МАРКЕРА, ИМЕЮЩИЙ ВИД:

%<ВЫРАЖЕНИЕ 1>,<ВЫРАЖЕНИЕ 2>

ПРИЧЕМ: "ВЫРАЖЕНИЕ 1" = (1 ... 24) И "ВЫРАЖЕНИЕ 2" = (1 ... 79)

ЗНАЧЕНИЕ "ВЫРАЖЕНИЯ 1" УКАЗЫВАЕТ НОМЕР СТРОКИ (С ВЕРХУ) ЭКРАНА ДИСПЛЕЯ, А ЗНАЧЕНИЕ "ВЫРАЖЕНИЯ 2" - НОМЕР ПОЗИЦИИ В СТРОКЕ. ВЫРАЖЕНИЯ В УКАЗАТЕЛЕ ПОЗИЦИИ МАРКЕРА МОГУТ БЫТЬ КАК ЦЕЛОГО, ТАК И ВЕЩЕСТВЕННОГО ТИПОВ. В ПОСЛЕДНЕМ СЛУЧАЕ ИСПОЛЬЗУЕТСЯ ЕГО ЦЕЛАЯ ЧАСТЬ.

УКАЗАТЕЛЬ МАРКЕРА ПОМЕЩАЕТСЯ НАПОСРЕДСТВЕННО ЗА ОТКРЫВАЮЩЕЙ СКОБКОЙ ФОРМАТНОЙ ЧАСТИ ОПЕРАТОРА. СЛЕДУЮЩИЕ ЗА НИМ ОБЪЕКТЫ БУДУТ РАЗМЕЩАТЬСЯ С УКАЗАННОЙ ПОЗИЦИИ ЭКРАНА. ЕСЛИ СКОБКА, ПРЕДШЕСТВУЮЩАЯ УКАЗАТЕЛЮ ИМЕЕТ ПОВТОРИТЕЛЬ, ТО ПРИ ПОВТОРНОМ ВЫПОЛНЕНИИ, УКАЗАТЕЛЬ БУДЕТ УСТАНАВЛИВАТЬ МАРКЕР В ЗАДАННОЙ ПОЗИЦИИ СЛЕДУЮЩЕЙ СТРОКИ ЭКРАНА.

ПРИМЕРЫ ОПЕРАТОРОВ ВВОДА/ВЫВОДА.

~~~~~

РАБОТУ ПРИВЕДЕННЫХ НИЖЕ ПРИМЕРОВ ОПЕРАТОРОВ ВВОДА/ВЫВОДА МОЖНО ПРОВЕРИТЬ ВЫПОЛНЕНИЕМ ПРИВЕДЕННОЙ ПРОГРАММЫ.

ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРОВ ВЫВОДА С УКАЗАНИЕМ НОМЕРА КАНАЛА ДЛЯ ПЕРЕМЕННОЙ "CAN" МОЖНО ВВЕСТИ "CAN=1" ДЛЯ ВЫВОДА НА ДИСПЛЕЙ ИЛИ "CAN=3" ДЛЯ ВЫВОДА НА ПЕЧАТЬ.

PP

INTEGER: CAN  
READ/'НОМЕР КАНАЛА ВЫВОДА: ДИСПЛЕЙ (1) ИЛИ ПЕЧАТЬ (3): 'CAN

ДИСПЛЕЙ И ПЕЧАТЬ.

~~~~~

11.4. ВВОД/ВЫВОД ЧИСЛОВОЙ ИНФОРМАЦИИ.

~~~~~

REAL: X, A[2, 3]; INT: M, N, MOD[2, 2]

1. БЕСФОРМАТНЫЙ И ФОРМАТНЫЙ ВВОД/ВЫВОД.

~~~~~

READ(X; M; N; /A; /MOD

ЭТОТ ЖЕ ВВОД МОЖНО ВЫПОЛНИТЬ ОПЕРАТОРОМ СО ВСТРОЕННЫМ ФОРМАТОМ:

READ (/ 'X='E, XX'M='E, XX'N='E / 'A='6E / 'MOD='4E) X; M; N; MOD

ИЛИ ОПЕРАТОРОМ С ВНЕШНИМ ФОРМАТОМ:

100 ФОРМАТ (/ 'X='F3.4, XX'M='I1, XX'N='I1 / 'A='6F2.2 / 'MOD='4I3)
READ(1, 100) X; M; N; A; MOD

РАЗДЕЛИТЕЛЕМ ОБЪЕКТОВ ФОРМАТА СЛУЖИТ СИМВОЛ ",", КОТОРЫЙ НЕ ЯВЛЯЕТСЯ ОБЯЗАТЕЛЬНЫМ.

МОЖНО ТАКЖЕ СДЕЛАТЬ СМЕШАННОЕ УПРАВЛЕНИЕ ВВОДОМ С ПОМОЩЬЮ ФОРМАТА И ОБЪЕКТОВ СПИСКА ВВОДА:

READ (E) / 'X='X; 'M='M; 'N='N; / 'A='A; / 'MOD='MOD

АНАЛОГИЧНЫМ ОБРАЗОМ СТРОЯТСЯ И ОПЕРАТОРЫ ВЫВОДА:

WRITE (CAN) / X; M; N; / A; / MOD

ИЛИ ЕГО АНАЛОГ СО ВСТРОЕННЫМ ФОРМАТОМ:

WRITE (CAN, / 'X='F3.4, XX'M='I1, XX'N='I1 / 'A='6F2.2 / 'MOD='(I3)) X; M; N; A; MOD

ЗДЕСЬ ДЛЯ ВЫВОДА "MOD" НЕ ЗАДАН ПОВТОРИТЕЛЬ. ТЕМ НЕ МЕНЕЕ, ВСЕ ЭЛЕМЕНТЫ "MOD" ВЫВОДЯТСЯ В ФОРМАТЕ "I3", ПОСКОЛЬКУ ПРИ ИСЧЕРПАНИИ СПИСКА ФОРМАТА ЕГО ПОВТОРЕНИЕ ПРОИЗВОДИТСЯ ОТ ПОСЛЕДНЕЙ ОТКРЫВАЮЩЕЙ СКОБКИ.

ВВОД ЛЮБОЙ ПЕРЕМЕННОЙ МОЖЕТ БЫТЬ ПРОИЗВЕДЕН В ЗАДАННОЙ ОБЛАСТИ ЭКРАНА С ПОМОЩЬЮ УКАЗАТЕЛЯ МАРКЕРА, КОТОРЫЙ ЗАПИСЫВАЕТСЯ НЕПОСРЕДСТВЕННО ПЕРЕД ЕЕ ИМЕНЕМ В СПИСКЕ ОБЪЕКТОВ ВВОДА:

READ%12, 40; X; %15, 30; M; N

2. ПЕРЕМЕННЫЙ ФОРМАТ ВЫВОДА.

```
~~~~~  
REAL: F1, F2  
READ%24, 1; F1; F2  
WRITE (CAN, F(F1), F(F2)) / 'X='X; 'A[2,3]='A[2,3]
```

ЗДЕСЬ ФОРМАТ ВЫВОДА "X" И ЭЛЕМЕНТА "A[2,3]" ЗАДАЕТСЯ ЗНАЧЕНИЯМИ "F1" И "F2" (НАПРИМЕР, F1=3.2 , F2=2.4), КОТОРЫЕ МОГУТ БЫТЬ ЛИБО ВВЕДЕНЫ, ЛИБО ВЫЧИСЛЕННЫ.

3. СМЕНА ВВОДА/ВЫВОДА.

```
~~~~~  
REAL: Y, Z  
READ/X; Y; < X+Y; > Z; <(X+Y)/Z
```

РЕЗУЛЬТАТ ВЫПОЛНЕНИЯ:

X=7 Y=5 X+Y= .12000 E 02 Z=3 (X+Y)/Z= .40000 E 01

4. ФОРМАТИЗАЦИЯ ВВОДА/ВЫВОДА МАССИВОВ.

```
~~~~~  
READ/M; N  
REAL: T[M, N]  
READ (' ТАБЛИЦА' /M(N: E/)) T
```

РЕЗУЛЬТАТ ДЛЯ M=3 N=4

```
ТАБЛИЦА  
1 2 3 4  
5 6 7 8  
9 10 11 12
```

АНАЛОГИЧНЫЙ ВВОД МОЖНО ПОСТРОИТЬ С ПОМОЩЬЮ СПИСКА ВВОДА:

```
INTEGER: I, J  
READ (E) / 'ТАБЛИЦА' / I:=1; M: (J:=1; N: (T[I, J]; J:=J+1); / I:=I+1)
```

В ПРИВЕДЕННЫХ ОПЕРАТОРАХ ВВОДА РАСПОЛОЖЕНИЕ ЭЛЕМЕНТОВ ПРИ ВВОДЕ ЗАВИСИТ ОТ ЧИСЛА ЗНАКОВ ВО ВВОДИМЫХ ЗНАЧЕНИЯХ. МОЖНО ПОСТРОИТЬ ОПЕРАТОР ВВОДА С ФИКСИРОВАННЫМ ПОЛОЖЕНИЕМ ВВОДИМЫХ ЭЛЕМЕНТОВ И С ВЫВОДОМ, ЕСЛИ НУЖНО, ИХ ОБОЗНАЧЕНИЙ:

```
READ (Z1) / 'ТАБЛИЦА' / I:=1; M: (J:=1; N: (%24, 10*(J-1); <'T['T;', 'J;']  
=>'>T[I, J]; J:=J+1); / I:=I+1)
```

РЕЗУЛЬТАТ:

```
ТАБЛИЦА  
T[1,1]=1 T[1,2]=2 T[1,3]=3 T[1,4]=4  
T[2,1]=5 T[2,2]=6 T[2,3]=7 T[2,4]=8  
T[3,1]=9 T[3,2]=10 T[3,3]=11 T[3,4]=12
```

В ПРИВЕДЕННЫХ ПРИМЕРАХ ВВОД МАССИВА ПРОИЗВОДИТСЯ В НИЖНЕЙ СТРОКЕ ЭКРАНА СО СМЕЩЕНИЕМ ЭКРАНА ВВЕРХ ДЛЯ КАЖДОЙ СЛЕДУЮЩЕЙ СТРОКИ. ЕСЛИ НЕОБХОДИМО СОХРАНИТЬ СОДЕРЖИМОЕ ЭКРАНА ПРИ ВВОДЕ МАССИВА, МОЖНО ОРГАНИЗОВАТЬ ЕГО ВВОД В ЖЕЛАЕМОЙ ЧАСТИ ЭКРАНА С ПОМОЩЬЮ УКАЗАТЕЛЯ МАРКЕРА В ФОРМАТНОЙ ЧАСТИ ИЛИ В СПИСКЕ ОБЪЕКТОВ ВВОДА:

```

INTEGER: IS, JS
READ/IS; JS
/*ВВОД МАССИВА С JS-Й ПОЗИЦИИ IS-Й СТРОКИ
READ (%IS, JS 'ТАБЛИЦА', М(%IS+1, JS; N: E/)) [0C] T

```

ИЛИ

```

READ (E) [0C] %IS, JS 'ТАБЛИЦА' IS:=IS+1; I:=1; M: (%IS, JS; J:=1; N: (T[I,
J]; J:=J+1); /I:=I+1; IS:=IS+1)

```

ОПЕРАТОРЫ ВЫВОДА МАССИВОВ СТРОЯТСЯ АНАЛОГИЧНО ОПЕРАТОРАМ ВВОДА С УКАЗАНИЕМ НЕОБХОДИМОГО ФОРМАТА ВЫВОДА ЗНАЧЕНИЙ.

11.5. ВВОД/ВЫВОД СИМВОЛЬНОЙ ИНФОРМАЦИИ.

~~~~~

```

CHAR: ТЕКСТ(10); REAL: ИМЯ, ИМЕНА[3, 2]; INT: Т, ТАБ[3, 8]

```

##### 1. БЕСФОРМАТНЫЙ ВВОД/ВЫВОД.

~~~~~

ПРИ БЕСФОРМАТНОМ ВВОДЕ (ВЫВОДЕ) СИМВОЛЬНОЙ ИНФОРМАЦИИ ПРИЗНАКОМ СИМВОЛЬНОГО ВВОДА ЯВЛЯЕТСЯ НАЛИЧИЕ СИМВОЛЬНОЙ ИНДЕКСАЦИИ У ПЕРЕМЕННОЙ:

```

READ%24, 1; ТЕКСТ(1, 10); /ИМЯ(1, 8) @; /I:=1; 3: (ИМЕНА[I, 1] (1, 16); '
:   ТАБ[I, 1] (1, 16); I:=I+1); T(1, 2) @

```

ВВОД ЗНАЧЕНИЙ ПЕРЕМЕННЫХ: "ИМЯ" И "Т" В ПРИВЕДЕННОМ ОПЕРАТОРЕ БУДЕТ РЕАЛИЗОВАН В ГРУППОВОМ РЕЖИМЕ, Т. Е. ВВОДИМАЯ В НИХ ИНФОРМАЦИЯ НЕ БУДЕТ ОТОБРАЖАТЬСЯ НА ЭКРАНЕ.

АНАЛОГИЧНЫМ ОБРАЗОМ СТРОИТСЯ БЕСФОРМАТНЫЙ ВЫВОД СИМВОЛЬНОЙ ИНФОРМАЦИИ. ДЛЯ ОБЪЕКТОВ С ГРУППОВЫМ ВЫВОДОМ ИХ ЗНАЧЕНИЯ БУДУТ ОТОБРАЖЕНЫ НА ЭКРАНЕ ИЛИ УСТРОЙСТВЕ ПЕЧАТИ.

2. ФОРМАТИЗОВАННЫЙ ВВОД/ВЫВОД.

~~~~~

```

CHAR: ТЕКСТ(10)

```

ПРИ СПЕЦИФИКАЦИИ "А" БЕЗ УКАЗАНИЯ РАЗМЕРА ФОРМАТИЗОВАННЫЙ ВВОД/ВЫВОД ОТЛИЧАЕТСЯ ОТ БЕСФОРМАТНОГО ТОЛЬКО ОТСУТСТВИЕМ ВЫВОДА ОБОЗНАЧЕНИЯ СИМВОЛЬНОЙ ПЕРЕМЕННОЙ, КОТОРОЕ В СЛУЧАЕ НЕОБХОДИМОСТИ МОЖЕТ БЫТЬ ЗАДАНО СООТВЕТСТВУЮЩЕЙ СИМВОЛЬНОЙ КОНСТАНТОЙ.

ПРИ УКАЗАНИЕ РАЗМЕРА В СПЕЦИФИКАЦИИ "А" ОН ЯВЛЯЕТСЯ ОПРЕДЕЛЯЮЩИМ ДЛЯ ВВОДА/ВЫВОДА, ЕСЛИ РАЗМЕР СИМВОЛЬНОЙ ПЕРЕМЕННОЙ (ИЛИ ЕЕ ФРАГМЕНТА) БОЛЬШЕ ЗАДАННОГО В СПЕЦИФИКАЦИИ. В ПРОТИВНОМ СЛУЧАЕ ДЛИНА ВВОДИМОЙ (ВЫВОДИМОЙ) СИМВОЛЬНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ ОПРЕДЕЛЯЕТСЯ РАЗМЕРОМ ПЕРЕМЕННОЙ (ФРАГМЕНТА). НАПРИМЕР:

```

ТЕКСТ<='ABCDEFGHIJ'
READ (/ 'ТЕКСТ: 'A4) ТЕКСТ(4, 3)
WRITE (CAN, A) /ТЕКСТ

```

РЕЗУЛЬТАТ:

ТЕХТ:123  
ABC123GHIJ

```
READ ('ТЕХТ: 'A3) ТЕХТ (4, 4)
WRITE (CAN, A) /ТЕХТ
```

РЕЗУЛЬТАТ:

ТЕХТ: 456  
ABC456GHIJ

ЕСЛИ ВВОД ПРЕКРАЩАЕТСЯ ДО ИСЧЕРПАНИЯ ОПРЕДЕЛЯЮЩЕГО РАЗМЕРА СИМВОЛОМ "BK", ТО ОСТАЮЩАЯСЯ ЧАСТЬ АВТОМАТИЧЕСКИ ЗАПОЛНЯЕТСЯ СИМВОЛАМИ "ПРОБЕЛ".

```
READ ('ТЕХТ: 'A5) ТЕХТ (2, *)
WRITE (CAN, A) /ТЕХТ
```

РЕЗУЛЬТАТ:

ТЕХТ: 789"BK"  
A789 GHIJ

### 3. ПОСТРОЕНИЕ СИМВОЛЬНЫХ ИЗОБРАЖЕНИЙ НА ЭКРАНЕ.

~~~~~

ДЛЯ ПОСТРОЕНИЯ СИМВОЛЬНЫХ ИЗОБРАЖЕНИЙ НА ЭКРАНЕ ДИСПЛЕЯ МОЖНО ИСПОЛЬЗОВАТЬ КАК ПРЕДВАРИТЕЛЬНО ПОДГОТОВЛЕННЫЕ СИМВОЛЬНЫЕ ПЕРЕМЕННЫЕ, ТАК И СИМВОЛЬНЫЕ КОНСТАНТЫ. НАПРИМЕР, ПОСТРОЕНИЕ ОБРАМЛЕНИЯ ЭКРАНА МОЖНО ВЫПОЛНИТЬ ОПЕРАТОРОМ:

```
WRITE [0C] %2, 1; 79' &' ; 20: ([1D 1A] '!') ; %3, 1; 20: ('!' [1D 1A]) ; 79' #'
```

4. АСИНХРОННЫЙ ВВОД С КЛАВИАТУРЫ ДИСПЛЕЯ.

~~~~~

ИСПОЛЬЗУЕТСЯ ДЛЯ ВВОДА ОДНОГО СИМВОЛА С КЛАВИАТУРЫ ДИСПЛЕЯ БЕЗ ОСТАНОВКИ ПРОГРАММЫ ДЛЯ ВЫПОЛНЕНИЯ ОПЕРАЦИЙ ВВОДА.

ВВЕДЕННЫЙ СИМВОЛ ЗАПИСЫВАЕТСЯ ВО ВТОРОЙ БАЙТ ЦЕЛОЙ ПЕРЕМЕННОЙ, УКАЗАННОЙ В ОПЕРАТОРЕ ВВОДА. ПОКА СИМВОЛ НЕ ВВЕДЕН, ЗНАЧЕНИЕ ПЕРЕМЕННОЙ ОСТАЕТСЯ РАВНЫМ НУЛЮ. ВВЕДЕННЫЙ СИМВОЛ НА ЭКРАНЕ НЕ ОТОБРАЖАЕТСЯ. ПРИМЕР ФРАГМЕНТА ПРОГРАММЫ:

```
INTEGER: INPUT
INPUT=0
WHILE ^INPUT
READ: INPUT
END
WRITE /'ЕСТЬ ' INPUT (2)
```



## МАГНИТНАЯ ЛЕНТА

## 11.6. ВВОД/ВЫВОД ДАННЫХ.

ОПЕРАЦИИ ВВОДА/ВЫВОДА С МЛ РЕАЛИЗУЮТСЯ УКАЗАНИЕМ КАНАЛА 2 В ОПЕРАТОРЕ ВВОДА/ВЫВОДА. ПРИ ЭТОМ ВОЗМОЖЕН КАК БЕСФОРМАТНЫЙ, ТАК И ФОРМАТИЗИРОВАННЫЙ ОБМЕН ДАННЫМИ С МЛ. ПРИ НАЛИЧИИ В ПРОГРАММЕ ОПЕРАТОРОВ ОБМЕНА С МЛ ПЕРЕД ЕЕ ВЫПОЛНЕНИЕМ ОБЯЗАТЕЛЬНО ЗАДАНИЕ ЧИСЛА УЖЕ ЗАПИСАННЫХ ФАЙЛОВ ДАННЫХ НА МЛ ДИРЕКТИВОЙ:

МРП:ЛЕНТА:ФАЙЛОВ:<ЧИСЛО>

ПРИ ОТСУТСТВИИ ЗАПИСЕЙ НА МЛ УКАЗЫВАЕТСЯ НУЛЕВОЕ ЗНАЧЕНИЕ.

## 1. БЕСФОРМАТНЫЙ ВВОД/ВЫВОД

ДЛЯ БЕСФОРМАТНОГО ВЫВОДА ПРОИЗВОДИТСЯ ЗАПИСЬ ВСЕХ ОБЪЕКТОВ СПИСКА ОПЕРАТОРА НА МЛ С ЕЕ ТЕКУЩЕГО СОСТОЯНИЯ. КАЖДЫЙ ОБЪЕКТ ОФОРМЛЯЕТСЯ ОТДЕЛЬНЫМ ФАЙЛОМ СООТВЕТСТВУЮЩЕГО РАЗМЕРА. ФАЙЛУ ПРИСВАИВАЕТСЯ ПОРЯДКОВЫЙ НОМЕР, РАВНЫЙ ПОРЯДКОВОМУ НОМЕРУ ОБЪЕКТА В СПИСКЕ.

НАПРИМЕР:

```
REAL:X,A[4]; CHAR:L(12); INT:K,M[5]
X=1; A=2; K=3; M=4
L<='ABCDEFGHIJKL'
WRITE(2)/X;A;K;M;L
```

ОПЕРАТОР БЕСФОРМАТНОГО ВЫВОДА МОЖЕТ БЫТЬ В ПРОГРАММЕ ТОЛЬКО ОДИН И ДОЛЖЕН ВЫПОЛНЯТЬСЯ РАНЕЕ ВСЕХ ОСТАЛЬНЫХ ОПЕРАТОРОВ ОБМЕНА С МЛ. ПРИ НЕОБХОДИМОСТИ ПОВТОРНОГО ВЫПОЛНЕНИЯ ЭТОГО ОПЕРАТОРА ЕМУ ДОЛЖНА ПРЕДШЕСТВОВАТЬ КОДОВАЯ КОМАНДА [C0] - ПЕРЕМОТКА МЛ НА НАЧАЛО.

ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА БЕСФОРМАТНОГО ВВОДА ПРОИЗВОДИТСЯ АВТОМАТИЧЕСКАЯ ПЕРЕМОТКА МЛ НА НАЧАЛО И СЧИТЫВАЮТСЯ ПОДРЯД ФАЙЛЫ, ОПРЕДЕЛЯЮЩИЕ ЗНАЧЕНИЯ ОБЪЕКТОВ СПИСКА В ПОРЯДКЕ ИХ СЛЕДОВАНИЯ.

```
REAL:Y,B[4]; C:LL(12); INT:N,MM[5]
READ(2)/Y; B; N; MM; LL
WRITE/Y;/B;/N;/MM;/LL(1,12)
```

НЕОБХОДИМЫМ УСЛОВИЕМ ВЫПОЛНЕНИЯ ОПЕРАТОРА ВВОДА ЯВЛЯЕТСЯ ТОЧНОЕ СООТВЕТСТВИЕ РАЗМЕРОВ СЧИТЫВАЕМЫХ ФАЙЛОВ И ОБЪЕКТОВ СПИСКА. ПРИ НАРУШЕНИИ ЭТОГО СООТВЕТСТВИЯ ВЫПОЛНЕНИЕ ОПЕРАТОРА ПРЕКРАЩАЕТСЯ И ДАЕТСЯ СООБЩЕНИЕ ОБ ОШИБКЕ. ЕСЛИ ЗАДАННОЕ ЧИСЛО ФАЙЛОВ МЕНЬШЕ ЧИСЛА ОБЪЕКТОВ ВВОДА, ОПЕРАТОР ТАКЖЕ НЕ БУДЕТ ВЫПОЛНЕН.

## 2. ФОРМАТИЗОВАННЫЙ ВВОД/ВЫВОД

ПРИ ФОРМАТИЗОВАННОМ ОБМЕНЕ С МЛ ОСНОВНОЙ В ФОРМАТНОЙ ЧАСТИ ОПЕРАТОРА ЯВЛЯЕТСЯ СПЕЦИФИКАЦИЯ T<ЧИСЛО>. ВСЕ ОСТАЛЬНЫЕ СПЕЦИФИКАЦИИ В ОПЕРАТОРЕ ОБМЕНА С МЛ ЗАПРЕЩЕНЫ. В ОСТАЛЬНОМ ПОСТРОЕНИЕ ФОРМАТНОЙ ЧАСТИ ОПЕРАТОРА НЕ ОТЛИЧАЕТСЯ ОТ ВВОДА/ВЫВОДА ЧИСЛОВОЙ И СИМВОЛЬНОЙ ИНФОРМАЦИИ.

В СПЕЦИФИКАЦИИ Т<ЧИСЛО> ЧИСЛО УКАЗЫВАЕТ НОМЕР ФАЙЛА НА МЛ, КОТОРОМУ СООТВЕТСТВУЕТ ОЧЕРЕДНОЦ ОБЪЕКТ СПИСКА. ВМЕСТО ЧИСЛА В СПЕЦИФИКАЦИИ "Т" МОЖЕТ БЫТЬ УКАЗАНА ЦЕЛАЯ ПЕРЕМЕННАЯ, ЗНАЧЕНИЕ КОТОРОЙ ОПРЕДЕЛЯЕТ НОМЕР ФАЙЛА. КРОМЕ ТОГО, ПЕРЕД СПЕЦИФИКАЦИЕЙ МОЖЕТ БЫТЬ УКАЗАН ПОВТОРИТЕЛЬ (ЧИСЛО ИЛИ ЦЕЛАЯ ПЕРЕМЕННАЯ), ОПРЕДЕЛЯЮЩИЙ КОЛИЧЕСТВО СЧИТЫВАЕМЫХ ИЛИ ЗАПИСЫВАЕМЫХ ФАЙЛОВ. ЕСЛИ НА МЛ ВЫВОДЯТСЯ НОВЫЕ ФАЙЛЫ В СООТВЕТСТВУЮЩИХ СПЕЦИФИКАЦИЯХ ДОЛЖНЫ БЫТЬ УКАЗАНЫ ИХ НОМЕРА, СЛЕДУЮЩИЕ ЗА НОМЕРОМ, ЗАДАННЫМ КАК ЧИСЛО ФАЙЛОВ НА МЛ НАЧАЛЬНОЙ ДИРЕКТИВОЙ.

ДОПУСКАЕТСЯ ПОВТОРНАЯ ЗАПИСЬ ФАЙЛА НА МЕСТО УЖЕ ЗАПИСАННОГО ПРИ УСЛОВИИ, ЧТО РАЗМЕР НОВОЙ ЗАПИСИ В БАЙТАХ РАВЕН РАЗМЕРУ ПРЕДШЕСТВУЮЩЕЙ.

ПОИСК ФАЙЛА, ЗАДАННОГО СПЕЦИФИКАЦИЕЙ "Т" ПРОИЗВОДИТСЯ В ПРЕДЕЛАХ ЧИСЛА ФАЙЛОВ, ЗАПИСАННЫХ К ТЕКУЩЕМУ МОМЕНТУ НА МЛ. В СЛУЧАЕ НЕОБХОДИМОСТИ ВОЗВРАТА К ФАЙЛАМ С МЕНЬШИМ НОМЕРОМ МЛ АВТОМАТИЧЕСКИ ПЕРЕМАТЫВАЕТСЯ НА НАЧАЛО.

ПРИМЕРЫ:

```
Y:=5; A:=6; B:=7; MM:=8
WRITE(2,/'Y,A->МЛ1-2'Т1,' MM->МЛ4'Т4,' B->МЛ6'Т6)Y;A;MM;B
READ(2,Т2,Т5,Т6)/'Т2->А'А;' Т5->LL'LL;' Т6->В'В
WRITE/A;/LL(1,12);B
```

#### 11.7. ВВОД СЕГМЕНТОВ РАБОЧИХ ПРОГРАММ.

В ВТ-МХТИ ПРЕДУСМОТРЕН СПЕЦИАЛЬНЫЙ ОПЕРАТОР ВВОДА С МЛ, ПОЗВОЛЯЮЩИЙ ПРОИЗВЕСТИ ЗАГРУЗКУ ГРУППЫ СЕГМЕНТОВ РАБОЧИХ ПРОГРАММ В РАСШИРЕННУЮ ПАМЯТЬ ДЗ-28. ЗАГРУЗКА ПРОИЗВОДИТСЯ НАЧИНАЯ С ЗАДАННОГО НОМЕРА СЕГМЕНТА ПАМЯТИ. ОПЕРАТОР ИМЕЕТ ВИД:

```
READ#<ЦЕЛОЕ>,<ЦЕЛОЕ>!<ЦЕЛЫЙ МАССИВ>,...
```

ЗНАЧЕНИЕ ПЕРВОГО ЦЕЛОГО (ЧИСЛО ИЛИ ЦЕЛАЯ ПЕРЕМЕННАЯ), ЗАДАЕТ НОМЕР СЕГМЕНТА ПАМЯТИ. ВТОРОЕ ЦЕЛОЕ (ЧИСЛО ИЛИ ЦЕЛАЯ ПЕРЕМЕННАЯ) ИЛИ ЭЛЕМЕНТЫ МАССИВОВ ЗАДАЮТ КОДЫ ЗАГРУЖАЕМЫХ СЕГМЕНТОВ РАБОЧИХ ПРОГРАММ. ПОИСК И СЧИТЫВАНИЕ СЕГМЕНТОВ С МЛ ВЕДЕТСЯ ДО ПЕРВОГО НУЛЕВОГО ЭЛЕМЕНТА МАССИВА ИЛИ ДО ИСЧЕРПАНИЯ ВСЕГО СПИСКА МАССИВОВ.

ЭТОТ ОПЕРАТОР ИСПОЛЬЗУЕТСЯ ДЛЯ СОЗДАНИЯ ПАКЕТОВ ПРОГРАММ, НЕ ТРЕБУЮЩИХ ДЛЯ ИСПОЛНЕНИЯ ВВОДА ОПЕРАЦИОННОЙ СИСТЕМЫ В ЭВМ.

ПРИМЕР:

```
INTEGER:СЕМ[10]
DATA:СЕМ=1;4;5;9;12;0
READ#2,СЕМ
```

ЗДЕСЬ ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА БУДУТ ЗАГРУЖЕНЫ СЕГМЕНТЫ С КОДАМИ 1,4,5,9,12, НАЧИНАЯ С СЕГМЕНТА 2 РАСШИРЕННОЙ ПАМЯТИ.

НАЧАЛЬНЫЙ ВЫЗОВ С МЛ И ЗАПУСК ПРОГРАММЫ, СОДЕРЖАЩЕЙ ОПЕРАТОР ЗАГРУЗКИ СЕГМЕНТОВ РАБОЧИХ ПРОГРАММ, МОЖЕТ БЫТЬ СДЕЛАН ДИРЕКТИВОЙ РАБОЧЕЙ ПРОГРАММЫ:

```
МРП:ЛЕНТА:ПРОГРАММА КОД:<ЧИСЛО> НБ:<ЧИСЛО>
```

ПОСЛЕ СЧИТЫВАНИЯ БЛОКА РАБОЧИХ ПОДПРОГРАММ (БРПП) С МЛ.

## 12. ОПЕРАТОРЫ ПЕРЕХОДОВ

~~~~~

В ВТ-МХТИ РЕАЛИЗОВАН ШИРОКИЙ НАБОР ОПЕРАТОРОВ ПЕРЕХОДОВ, ПОЗВОЛЯЮЩИХ СТРОИТЬ РАЗВЕТВЛЕНИЯ В ПРОГРАММАХ В ЗАВИСИМОСТИ ОТ РАЗЛИЧНЫХ ПРИЧИН, ВКЛЮЧАЯ ВНЕШНИЕ ПЕРЕРЫВАНИЯ И СИГНАЛЫ С КЛАВИАТУРЫ ДИСПЛЕЯ, А ТАКЖЕ СОЗДАВАТЬ ПРОГРАММНЫЕ КОМПЛЕКСЫ, ОБЪЕДИНЯЮЩИЕ РАБОТУ НЕСКОЛЬКИХ ПРОГРАММ НА ОБЩЕМ ПОЛЕ ПАМЯТИ.

ОСОБУЮ ГРУППУ СРЕДИ ОПЕРАТОРОВ ПЕРЕХОДА СОСТАВЛЯЮТ ОПЕРАТОРЫ ОБРАБОТКИ ВНЕШНИХ ПЕРЕРЫВАНИЙ И ОБРАЩЕНИЯ К ПОДПРОГРАММАМ, КОТОРЫЕ БУДУТ РАССМОТРЕНЫ В РАЗДЕЛЕ "ПОДПРОГРАММЫ".

В ПРОГРАММАХ ВТ-МХТИ ПРИНЯТЫ ДВА ТИПА ПЕРЕХОДОВ:

1. ПЕРЕХОД НА ОПЕРАТОР, ПОМЕЧЕННЫЙ ЧИСЛОВОЙ МЕТКОЙ - ЦЕЛЫМ ЧИСЛОМ ОТ 1 ДО 999. МЕТКОЙ МОЖЕТ БЫТЬ ПОМЕЧЕН ЛЮБОЙ ОПЕРАТОР ПРОГРАММЫ, ИСКЛЮЧАЯ ОПЕРАТОР ЗАГОЛОВКА ПРОГРАММЫ.
2. ПЕРЕХОД НА СИМВОЛЬНУЮ (НАЧИНАЮЩУЮСЯ С БУКВЫ) МЕТКУ, ИМЯ КОТОРОЙ ОПИСЫВАЕТСЯ ОПЕРАТОРОМ "LABEL:", А МЕСТО ПЕРЕХОДА ОПРЕДЕЛЯЕТСЯ ОПЕРАТОРОМ "LABEL <ИМЯ МЕТКИ>".

12.1. БЕЗУСЛОВНЫЕ ПЕРЕХОДЫ

~~~~~

## 1. ЯВНЫЙ ПЕРЕХОД НА ЧИСЛОВУЮ МЕТКУ:

~~~~~

ГОТО 123

ПЕРЕДАЕТ УПРАВЛЕНИЕ СТРОКЕ ОПЕРАТОРОВ:

123 X=5.756; Y=.001; MET=215; I=3

2. НЕЯВНЫЙ ПЕРЕХОД НА ЧИСЛОВУЮ МЕТКУ:

~~~~~

ГОТО MET

ПЕРЕДАЕТ УПРАВЛЕНИЕ СТРОКЕ С МЕТКОЙ, РАВНОЙ ЗНАЧЕНИЮ ЦЕЛОЙ ПЕРЕМЕННОЙ, УКАЗАННОЙ В ОПЕРАТОРЕ ПЕРЕХОДА. МОЖНО ИСПОЛЬЗОВАТЬ ТАКЖЕ ЦЕЛЫЕ ПЕРЕМЕННЫЕ С ИНДЕКСАМИ:

ГОТО LAB[I]

ЕСЛИ МЕТКА, ОПРЕДЕЛЯЕМАЯ НЕЯВНЫМ ПЕРЕХОДОМ, В ПРОГРАММЕ ОТСУТСТВУЕТ, ПРИ ВЫПОЛНЕНИИ ПРОГРАММЫ ДАЕТСЯ СООБЩЕНИЕ ОБ ОШИБКЕ. ЕСЛИ В ПРОГРАММЕ (В РАЗЛИЧНЫХ ПРОГРАММНЫХ ЕДИНИЦАХ) НЕСКОЛЬКО ОДИНАКОВЫХ МЕТОК, ТО НЕЯВНЫЙ ПЕРЕХОД ОСУЩЕСТВЛЯЕТСЯ К ПОСЛЕДНЕЙ ИЗ НИХ.

## 3. ПЕРЕХОД НА СИМВОЛЬНУЮ МЕТКУ:

~~~~~

ГОТО ПЕРВАЯ

ИЛИ

ПЕРВАЯ

В ОБОИХ СЛУЧАЯХ РЕАЛИЗУЕТСЯ ОДНА И ТА ЖЕ ОПЕРАЦИЯ ПЕРЕХОДА. ПОСКОЛЬКУ ДЛЯ АДРЕСОВ СИМВОЛЬНЫХ МЕТОК ОПРЕДЕЛЕНА СООТВЕТСТВУЮЩАЯ ЦЕЛЬ ПЕРЕМЕННЫЕ, ТО ДОПУСТИМО ПЕРЕНАЗНАЧЕНИЕ МЕТОК ПРИ ВЫПОЛНЕНИИ ПРОГРАММЫ. НАПРИМЕР, ПОСЛЕ ОПЕРАТОРОВ:

```
ПЕРВАЯ:=ВТОРАЯ
ГОТО ПЕРВАЯ
ПЕРЕХОД БУДЕТ ВЫПОЛНЕН НА МЕТКУ "ВТОРАЯ".
```

4. ПЕРЕХОД ПО СПИСКУ МЕТОК С УПРАВЛЯЮЩЕЙ ПЕРЕМЕННОЙ:
~~~~~  
В СПИСКЕ РАЗРЕШЕН ЛЮБОЙ СПОСОБ УКАЗАНИЯ МЕТОК.

```
ГОТО К: 70,ЛАВ[3],МЕТ,ПЕРВАЯ,ЛАВ[J],ВТОРАЯ
```

ЗДЕСЬ ПРОИСХОДИТ ПЕРЕХОД НА МЕТКУ, ПОРЯДКОВЫЙ НОМЕР КОТОРОЙ В СПИСКЕ РАВЕН ЗНАЧЕНИЮ ЦЕЛОЙ ПЕРЕМЕННОЙ "К", УПРАВЛЯЮЩЕЙ ЭТИМ ОПЕРАТОРОМ ПЕРЕХОДА. ЕСЛИ ЗНАЧЕНИЕ "К" НЕ СООТВЕТСТВУЕТ ЗАДАННОМУ СПИСКУ (НАПРИМЕР, В ДАННОМ СЛУЧАЕ "К"<1 ИЛИ "К">6), УПРАВЛЕНИЕ ПЕРЕДАЕТСЯ СЛЕДУЮЩЕМУ ОПЕРАТОРУ ПРОГРАММЫ.

5. ПЕРЕХОД ПО СПИСКУ МЕТОК С КЛАВИАТУРЫ ДИСПЛЕЯ:  
~~~~~  
ГОТО :50,ПЕРВАЯ,МЕТ,ЛАВ[1],ВТОРАЯ

В СПИСКЕ ЭТОГО ОПЕРАТОРА ПЕРЕХОДА МОЖНО ИСПОЛЬЗОВАТЬ НЕ БОЛЕЕ 15 МЕТОК. ПОРЯДКОВЫЙ НОМЕР МЕТКИ В СПИСКЕ, НА КОТОРУЮ ПРОИСХОДИТ ПЕРЕХОД ОПРЕДЕЛЯЕТСЯ МЛАДШЕЙ ТЕТРАДОЙ БАЙТА, СООТВЕТСТВУЮЩЕГО СИМВОЛУ НАЖАТОЙ КЛАВИШИ ДИСПЛЕЯ. КЛАВИШИ "1","2","3",... БУДУТ ЗАДАВАТЬ ПЕРЕХОДЫ СООТВЕТСТВЕННО НА 1-Ю, 2-Ю, 3-Ю,... МЕТКИ СПИСКА. АНАЛОГИЧНЫЕ ПЕРЕХОДЫ БУДУТ ВЫПОЛНЯТЬСЯ И ПРИ НАЖАТИИ КЛАВИШ А,В,С,... ИЛИ Q,R,S,..., А ТАКЖЕ В НЕКОТОРЫХ ДРУГИХ РЯДАХ СИМВОЛОВ, В КОТОРЫХ МЛАДШИЕ ТЕТРАДЫ БАЙТОВ ИДЕНТИЧНЫ ЦИФРОВОМУ РЯДУ. КАК И В ПРЕДЫДУЩЕМ СЛУЧАЕ, ЕСЛИ НЕТ СООТВЕТСТВУЮЩЕЙ МЕТКИ, ВЫПОЛНЯЕТСЯ СЛЕДУЮЩИЙ ОПЕРАТОР ПРОГРАММЫ.

12.2. УСЛОВНЫЕ ПЕРЕХОДЫ.

1. ПЕРЕХОД ПО ЗНАКУ ВЫРАЖЕНИЯ:
~~~~~

```
IF (X+Y-Z) 10,20,30
```

ЭТОТ ОПЕРАТОР АНАЛОГИЧЕН АРИФМЕТИЧЕСКОМУ "IF" В "FORTRAN" И ВЫПОЛНЯЕТ ПЕРЕХОДЫ НА МЕТКУ 10, ЕСЛИ  $X+Y-Z < 0$ , НА МЕТКУ 20, ЕСЛИ  $X+Y-Z = 0$ , И НА МЕТКУ 30, ЕСЛИ  $X+Y-Z > 0$ .

В ВТ-МХТИ ФУНКЦИИ ОПЕРАТОРА РАСШИРЕНА ЗА СЧЕТ РАЗЛИЧНЫХ СПОСОБОВ ЗАДАНИЯ МЕТОК, РАССМОТРЕННЫХ ВЫШЕ, И БОЛЕЕ ШИРОКОГО ОПРЕДЕЛЕНИЯ СКАЛЯРНОГО ВЫРАЖЕНИЯ (ДОПУСКАЕТСЯ, НАПРИМЕР, ИСПОЛЬЗОВАНИЕ ПРОСТЫХ  $(X > Y)$  И СЛОЖНЫХ  $(X+Y > Z ! X-Y < Z)$  УСЛОВИЙ). КРОМЕ ТОГО, СПИСОК МЕТОК МОЖЕТ БЫТЬ НЕПОЛНЫМ, И ОТСУТСТВИЕ КАКОЙ-ЛИБО ИЗ МЕТОК ОЗНАЧАЕТ, ЧТО ПРИ ВЫПОЛНЕНИИ СООТВЕТСТВУЮЩЕГО ЕЙ УСЛОВИЯ БУДЕТ ВЫПОЛНЯТЬСЯ СЛЕДУЮЩИЙ ОПЕРАТОР ПРОГРАММЫ. ЕСЛИ В ОПЕРАТОРЕ ИСПОЛЬЗУЕТСЯ ЛОГИЧЕСКОЕ ВЫРАЖЕНИЕ ИЛИ ОТНОШЕНИЕ, ТО ПЕРВАЯ МЕТКА В СПИСКЕ НЕ НУЖНА, ПОСКОЛЬКУ ЗНАЧЕНИЕМ ЛОГИЧЕСКОГО ВЫРАЖЕНИЯ МОЖЕТ БЫТЬ ТОЛЬКО 0 И 1.

ПРИМЕРЫ РЕАЛИЗАЦИИ УСЛОВНЫХ ПЕРЕХОДОВ:

IF (X>=Y) , , 30  
ПЕРЕХОД НА МЕТКУ 30 ПРИ ВЫПОЛНЕНИИ УСЛОВИЯ.

IF (X<Y) , ПЕРВАЯ , ВТОРАЯ  
ПЕРЕХОД НА МЕТКУ "ВТОРАЯ" ПРИ ВЫПОЛНЕНИИ УСЛОВИЯ И НА МЕТКУ "ПЕРВАЯ"  
ПРИ НЕВЫПОЛНЕНИИ.

IF (I&J) МЕТ , ЛАВ [2] , 30  
ПЕРЕХОД НА МЕТКУ , ЗАДАННУЮ ЗНАЧЕНИЕМ ПЕРЕМЕННОЙ МЕТ , ЕСЛИ I<0 И J<0 ;  
ЗАДАННУЮ ЗНАЧЕНИЕМ ЛАВ [2] , ЕСЛИ I И J НЕ ИМЕЮТ СОВПАДАЮЩИХ НЕ НУЛЕВЫХ  
БИТОВ , И НА МЕТКУ 30 В ОСТАЛЬНЫХ СЛУЧАЯХ.

IF (^K) , , ПЕРВАЯ  
ПЕРЕХОД НА МЕТКУ "ПЕРВАЯ" ПРИ K=0.

IF (^^K) , , ВТОРАЯ  
ПЕРЕХОД НА МЕТКУ "ВТОРАЯ" ПРИ K#0.

IF (SIGN (X) #SIGN (Y-Z) ) , , 30  
ПЕРЕХОД НА МЕТКУ 30 ПРИ ВЫПОЛНЕНИИ УСЛОВИЯ.

2. ПЕРЕХОД НА СИМВОЛЬНУЮ МЕТКУ ПО УСЛОВИЮ:

~~~~~  
(X>Y) : (ПЕРВАЯ)

ЭТА КОНСТРУКЦИЯ УСЛОВНОГО ПЕРЕХОДА МОЖЕТ ИСПОЛЬЗОВАТЬСЯ В СТРО-
КАХ ОПЕРАТОРОВ ИЛИ СПИСКАХ ОБЪЕКТОВ ВВОДА/ВЫВОДА ДЛЯ ПРОВЕРКИ РАЗЛИЧ-
НЫХ УСЛОВИЙ РАЗВЕТВЛЕНИЯ ИЛИ ОКОНЧАНИЯ ЦИКЛИЧЕСКИХ ПРОЦЕССОВ.

ПРИМЕРЫ НАПИСАНИЯ ОПЕРАТОРОВ , ПОМЕЧЕННЫХ МЕТКАМИ:

```
50 LABEL
70 K=10
   LABEL ПЕРВАЯ
   ВТОРАЯ
```

PP

13. ВСТРОЕННЫЕ ПОДПРОГРАММЫ.

~~~~~

В ВТ-МХТИ ВСТРОЕННАЯ ПОДПРОГРАММА ОРГАНИЗУЕТСЯ КАК ГРУППА  
ОПЕРАТОРОВ , ПЕРВЫЙ ИЗ КОТОРЫХ ИМЕЕТ ЧИСЛОВУЮ МЕТКУ , ОПРЕДЕЛЯЮЩУЮ ВХОД  
В ПОДПРОГРАММУ . ЗАВЕРШАЕТСЯ ПОДПРОГРАММА ОПЕРАТОРОМ ВЫХОДА (RETURN) ,  
ИМЕЮЩИМ ПРИЗНАК "F" . НАПРИМЕР:

```
/*ПОДПРОГРАММА ПРОИЗВЕДЕНИЕ ВЕКТОРОВ
10 S:=0.
   DO I:=1,6
     S:=S+MAS1[I]*MAS2[I]
   END I
RETURN F (ВЫХОД ИЗ ПОДПРОГРАММЫ)
```

ПОСКОЛЬКУ ВСТРОЕННАЯ ПОДПРОГРАММА ИМЕЕТ ОТКРЫТЫЙ ВХОД, НЕОБХОДИМЫ МЕРЫ, ИСКЛЮЧАЮЩИЕ ВОЗМОЖНОСТЬ ВХОДА В НЕЕ ИЗ ПРЕДШЕСТВУЮЩЕЙ ЧАСТИ ПРОГРАММЫ. С ЭТОЙ ЦЕЛЬЮ МОЖНО ПРИМЕНИТЬ ОПЕРАТОР БЕЗУСЛОВНОГО ПЕРЕХОДА, ОБОХОДЯЩИЙ ВЫДЕЛЕННУЮ ДЛЯ ВСТРОЕННЫХ ПОДПРОГРАММ ГРУППУ ОПЕРАТОРОВ, ИЛИ ВКЛЮЧИТЬ ЭТУ ГРУППУ ОПЕРАТОРОВ В НЕВЫПОЛНЯЕМЫЙ УСЛОВНЫЙ ОПЕРАТОР, НАПРИМЕР:

```
IF 0
/*СУММИРОВАНИЕ ПОЛОЖИТЕЛЬНЫХ ЭЛЕМЕНТОВ МАССИВА
20 S=0
DO I:=1,6
S=S+(MAS1[I]>0)*MAS1[I]
END I
RETURN F
END IF
```

РЕКОМЕНДУЕТСЯ ВСТРОЕННЫЕ ПОДПРОГРАММЫ РАЗМЕЩАТЬ В САМОМ НАЧАЛЕ ПРОГРАММЫ.

ОБРАЩЕНИЯ К ВСТРОЕННЫМ ПОДПРОГРАММАМ ВОЗМОЖНЫ ИЗ ЛЮБОГО МЕСТА ПРОГРАММЫ С ПОМОЩЬЮ ОПЕРАТОРОВ:

```
GOTO 10:F
GOTO 20:F
```

ГДЕ БУКВА "F" ЯВЛЯЕТСЯ ПРИЗНАКОМ ПЕРЕХОДА К ВСТРОЕННОЙ ПОДПРОГРАММЕ. В КАЧЕСТВЕ ПРИЗНАКА ОБРАЩЕНИЯ К ПОДПРОГРАММЕ ВМЕСТО "F" МОГУТ ИСПОЛЬЗОВАТЬСЯ И ЛЮБЫЕ ДРУГИЕ БУКВЫ (ИСКЛЮЧАЯ РУССКИЕ БУКВЫ "Б,В,И,Т" И ЛАТИНСКИЕ "B,W,I,T", ПРИМЕНЯЕМЫЕ В ДРУГИХ ОПЕРАТОРАХ ПЕРЕХОДА), А ТАКЖЕ ЛЮБЫЕ СИМВОЛЫ И ЦИФРЫ. НАПРИМЕР:

```
GOTO 10:*
GOTO 20:+
GOTO 10:Л
GOTO 20:G
```

PP

#### 14. ОБРАБОТКА ВНЕШНИХ ПРЕРЫВАНИЙ. ~~~~~

ДЛЯ ОБРАБОТКИ ВНЕШНИХ ПРЕРЫВАНИЙ ИСПОЛЬЗУЮТСЯ ВСТРОЕННЫЕ ПОДПРОГРАММЫ С ОПЕРАТОРОМ ВЫХОДА (RETURN), ИМЕЮЩИМ ПРИЗНАК "I".

НАПРИМЕР:

```
IF 0
30 WRITE/MAS1
RETURN I
END IF
```

ВНЕШНИЕ ПРЕРЫВАНИЯ ВЫЗЫВАЮТСЯ СИГНАЛАМИ:

1. УП2 (КОД 32) - СИГНАЛ ВВОДА СИМВОЛА С ПЕЧАТАЮЩЕГО УСТРОЙСТВА (ПМ);
2. УП1 (КОД 16) - СИГНАЛ ГОТОВНОСТИ ПЕЧАТАЮЩЕГО УСТРОЙСТВА;
3. ПР8 (КОД 8) - СИГНАЛ ВВОДА С КЛАВИАТУРЫ ДИСПЛЕЯ;
4. ПР4 (КОД 4) - СИГНАЛ ОТ ВНЕШНЕГО УСТРОЙСТВА;
5. ПР2 (КОД 2) - СИГНАЛ ОТ ВНЕШНЕГО УСТРОЙСТВА;
6. ПР1 (КОД 1) - СИГНАЛ ОТ ВНЕШНЕГО УСТРОЙСТВА.

ОПЕРАТОР ПЕРЕХОДА К ПОДПРОГРАММЕ ОБРАБОТКИ ПЕРЕРЫВАНИЙ ИМЕЕТ ПРИЗНАК "I" И В ПРОСТЕЙШЕМ СЛУЧАЕ МОЖЕТ БЫТЬ ЗАПИСАН В ВИДЕ:

```
ГОТО 30:I-4
```

КРОМЕ ЧИСЛА, ДЛЯ УКАЗАНИЯ МЕТКИ В ОПЕРАТОРЕ ПЕРЕХОДА МОЖЕТ ИСПОЛЬЗОВАТЬСЯ ТАКЖЕ ПЕРЕМЕННАЯ ЦЕЛОГО ТИПА, ПРОСТАЯ ИЛИ С ИНДЕКСАМИ.

ДЕЙСТВИЕ ЭТОГО ОПЕРАТОРА ПРИВОДИТ К ДЕМАСКИРОВАНИЮ СИГНАЛА ПЕРЕРЫВАНИЯ ПР4 (КОД 4). ЕСЛИ К МОМЕНТУ ВЫПОЛНЕНИЯ ОПЕРАТОРА СИГНАЛ ПР4 ПОСТУПИЛ, ТО СРАЗУ ПРОИСХОДИТ ПЕРЕХОД К ОПЕРАТОРУ С МЕТКОЙ 30. ПОСЛЕ ВОЗВРАТА ИЗ ПОД ПРОГРАММЫ ОБРАБОТКИ ПЕРЕРЫВАНИЯ ИЛИ ПРИ ОТСУТСТВИИ СИГНАЛА ПР4 ПРОДОЛЖАЕТСЯ ВЫПОЛНЕНИЕ ПРОГРАММЫ С ОПЕРАТОРОВ, СЛЕДУЮЩИХ ЗА ОПЕРАТОРОМ ПЕРЕХОДА. ПЕРЕРЫВАНИЕ ПР4 ОСТАЕТСЯ ПРИ ЭТОМ ДЕМАСКИРОВАННЫМ, И ПРИ ПОЯВЛЕНИИ СИГНАЛА ПР4 ОБРАЩЕНИЕ К ПОДПРОГРАММЕ ОБРАБОТКИ ПЕРЕРЫВАНИЯ С МЕТКОЙ 30 ПРОИЗОЙДЕТ ИЗ ТОГО МЕСТА ПРОГРАММЫ, ГДЕ ОНА ВЫПОЛНЯЛАСЬ В МОМЕНТ ПОСТУПЛЕНИЯ СИГНАЛА, ПОСЛЕ ВОЗВРАТА ИЗ ПОДПРОГРАММЫ ВЫПОЛНЕНИЕ ПРОГРАММЫ БУДЕТ ПРОДОЛЖЕНО С МЕСТА ЕЕ ПЕРЕРЫВАНИЯ.

ЕСЛИ ВЫХОД НА ПЕРЕРЫВАНИЕ ДОЛЖЕН БЫТЬ ТОЛЬКО ИЗ ТОЧКИ ПРОГРАММЫ ГДЕ РАСПОЛОЖЕН ОПЕРАТОР ПЕРЕХОДА, ТО СЛЕДУЕТ ИСПОЛЬЗОВАТЬ ДРУГУЮ ФОРМУ ОПЕРАТОРА:

```
ГОТО 30:I-4-0
```

ПОСЛЕ ВЫПОЛНЕНИЯ КОТОРОГО ВСЕ ПЕРЕРЫВАНИЯ МАСКИРУЮТСЯ.

ВОЗМОЖЕН ОПРОС НЕСКОЛЬКИХ ПЕРЕРЫВАНИЙ В ОДНОМ ОПЕРАТОРЕ ПЕРЕХОДА.

НАПРИМЕР:

```
ГОТО МЕТ:I-4-2-1-0
```

ИЛИ С СУММИРОВАНИЕМ КОДОВ:

```
ГОТО LAV[3]:I-7-0
```

МОЖНО СДЕЛАТЬ ПЕРЕХОДЫ НА РАЗНЫЕ МЕТКИ ДЛЯ РАЗНЫХ СИГНАЛОВ:

```
ГОТО 30:I-4-0,МЕТ:I-2-0,LAV[3]:I-1-0
```

А ТАКЖЕ ЗАВЕРШИТЬ ВЫПОЛНЕНИЕ ОПЕРАТОРА ОБРАБОТКИ ПЕРЕРЫВАНИЙ БЕЗУСЛОВНЫМ ПЕРЕХОДОМ НА ЗАДАННУЮ МЕТКУ:

```
ГОТО 30:I-6-0,LAV[4]:I-1-0,МЕТ
```

ЗДЕСЬ ПОСЛЕ КОНТРОЛЯ И (ПРИ НАЛИЧИИ СИГНАЛОВ) ОБРАБОТКИ ПЕРЕРЫВАНИЙ ПР4 И ПР2 (МЕТКА 30), ПР1 (МЕТКА - ЗНАЧЕНИЕ LAV[4]), А ТАКЖЕ ПРИ ОТСУТСТВИИ СИГНАЛОВ ПЕРЕРЫВАНИЯ ВЫПОЛНЯЕТСЯ ПЕРЕХОД НА МЕТКУ, ЗАДАННУЮ ЗНАЧЕНИЕМ ЦЕЛОЙ ПЕРЕМЕННОЙ "МЕТ".

ПРЕДУСМОТРЕНА ВОЗМОЖНОСТЬ ЗАДАНИЯ РЕЖИМА ОЖИДАНИЯ ПЕРЕРЫВАНИЯ. С ЭТОЙ ЦЕЛЬЮ В ЭЛЕМЕНТЕ СПИСКА ОПЕРАТОРА ПЕРЕХОДА НА ОБРАБОТКУ ПЕРЕРЫВАНИЯ ВМЕСТО ПРИЗНАКА "I" СЛЕДУЕТ УКАЗАТЬ ПРИЗНАК "W".

```
ГОТО 30:W-2-0
```

ЕСЛИ К МОМЕНТУ ВЫПОЛНЕНИЯ ЭТОГО ОПЕРАТОРА СИГНАЛ ПР2 ЕЩЕ НЕ ПОСТУПИЛ В ЭВМ, ВЫПОЛНЯЕТСЯ ДИНАМИЧЕСКИЙ ОСТАНОВ ДЛЯ ОЖИДАНИЯ ЕГО ПРИХОДА, ПОСЛЕ ЧЕГО ВЫПОЛНЕНИЕ ПРОГРАММЫ ПРОДОЛЖАЕТСЯ КАК ПРИ ОБЫЧНОМ ПРЕРЫВАНИИ.

ОСОБАЯ СИТУАЦИЯ ВОЗНИКАЕТ ПРИ ОБРАБОТКЕ СИГНАЛОВ ПРЕРЫВАНИЯ УП2 (СИГНАЛ ВВОДА С ПМ) И ПР8 (СИГНАЛ ВВОДА С КЛАВИАТУРЫ ДИСПЛЕЯ). ЗДЕСЬ ДЛЯ ПРОДОЛЖЕНИЯ НОРМАЛЬНОЙ РАБОТЫ ПРОГРАММЫ В ПОДПРОГРАММЕ ОБРАБОТКИ ПРЕРЫВАНИЯ НЕОБХОДИМО ВЫПОЛНИТЬ ВВОД С СООТВЕТСТВУЮЩЕГО КАНАЛА СИМВОЛА, ВЫЗВАВШЕГО ПРЕРЫВАНИЕ, ЛЮБЫМ ОПЕРАТОРОМ ВВОДА. ДЛЯ ПРЕРЫВАНИЯ С КЛАВИАТУРЫ ДИСПЛЕЯ МОЖЕТ, НАПРИМЕР, ОКАЗАТЬСЯ УДОБНЫМ ОПЕРАТОР АСИНХРОННОГО ВВОДА.

НАПРИМЕР:

```
IF 0
40 READ:I
WRITE/MAS2
RETURNI
END IF
```

ЕСЛИ ПРЕРЫВАНИЕ ИСПОЛЬЗУЕТСЯ ДЛЯ РАЗВЕТВЛЕНИЯ ПРОГРАММЫ И ВОЗВРАТА К ТОЧКЕ ПРЕРЫВАНИЯ НЕ ТРЕБУЕТСЯ ПРИМЕНЯЕТСЯ ОПЕРАТОР ПСЕВДОВОЗВРАТА ИЗ ПРЕРЫВАНИЯ "RETURNIP".

НАПРИМЕР, ПРИ ПЕРЕХОДЕ ПО ПРЕРЫВАНИЮ ПР8 НА ОПЕРАТОР С МЕТКОЙ 50 ВОЗМОЖНО СЛЕДУЮЩЕЕ ПРОДОЛЖЕНИЕ ПРОГРАММЫ:

```
50 READ:I
RETURN IP
/*<ПОСЛЕДОВАТЕЛЬНОСТЬ СТРОК ОПЕРАТОРОВ>
```

ПРИ ЭТОМ ПЕРЕХОД К ОПЕРАТОРУ С МЕТКОЙ 50 ДОПУСКАЕТСЯ ТОЛЬКО ПО ПРЕРЫВАНИЮ. КАКИЕ-ЛИБО ДРУГИЕ ОБРАЩЕНИЯ К МЕТКЕ 50 ЗАПРЕЩЕНЫ



## 15. ПРОГРАММЫ, ПОДПРОГРАММЫ, ОПЕРАТОР-ФУНКЦИИ.

В ВТ-МХТИ НЕТ РАЗДЕЛЕНИЯ ПРОГРАММНЫХ ЕДИНИЦ НА ПРОГРАММЫ И ПОДПРОГРАММЫ. ЛЮБОЙ ИСХОДНЫЙ МОДУЛЬ ЯВЛЯЕТСЯ ПРОГРАММОЙ И МОЖЕТ ОБРАЩАТЬСЯ К ДРУГИМ ПРОГРАММАМ (КАК К ПОДПРОГРАММАМ) ДЛЯ ВЫПОЛНЕНИЯ ОПРЕДЕЛЕННЫХ ИМИ ФУНКЦИЙ. ЭТО ДАЕТ ВОЗМОЖНОСТЬ БЕЗ КАКИХ-ЛИБО СПЕЦИАЛЬНЫХ ПРИЕМОВ ПРОВОДИТЬ МОДУЛЬНУЮ ОТЛАДКУ, НАЧИНАЯ С ПРОГРАММ САМОГО НИЗШЕГО УРОВНЯ, КОТОРЫЕ НЕ СОДЕРЖАТ ОБРАЩЕНИЙ К ДРУГИМ ПРОГРАММАМ, И ПЕРЕХОДЯ ЗАТЕМ К ВЫЗЫВАЮЩИМ ИХ ПРОГРАММАМ, ЧТО ПОЛНОСТЬЮ СООТВЕТСТВУЕТ ИДЕАЛОГИИ СТРУКТУРНОГО ПРОГРАММИРОВАНИЯ С НИСХОДЯЩИМ ПРОЦЕССОМ РАЗРАБОТКИ ПРОГРАММ И ВОСХОДЯЩЕЙ ПРОЦЕДУРОЙ ИХ ОТЛАДКИ.

ПРОГРАММА В ВТ-МХТИ МОЖЕТ ИМЕТЬ СПИСОК ПАРАМЕТРОВ, ВЫПОЛНЯЮЩИЙ РОЛЬ ОПИСАНИЙ ПРИ АВТОНОМНОМ ИСПОЛНЕНИИ ПРОГРАММЫ И ЯВЛЯЮЩИЙСЯ СПИСОКОМ ОПИСАНИЯ ФОРМАЛЬНЫХ ПАРАМЕТРОВ ПРИ ОБРАЩЕНИИ К ЭТОЙ ПРОГРАММЕ (КАК К ПОДПРОГРАММЕ) ИЗ ДРУГИХ ПРОГРАММ. КРОМЕ ТОГО, ОПРЕДЕЛЕННЫ ОПЕРАТОРЫ "ВВОДА/ВЫВОДА", ВЫПОЛНЯЕМЫЕ ПРИ АВТОНОМНОМ ИСПОЛНЕНИИ ПРОГРАММЫ И ЯВЛЯЮЩИЕСЯ КОМЕНТАРИЯМИ ПРИ ОБРАЩЕНИИ К НЕЙ КАК К ПОДПРОГРАММЕ.

ДЛЯ ОБЕСПЕЧЕНИЯ ВОЗВРАТА ИЗ ВЫЗЫВАЕМОЙ ПРОГРАММЫ В ВЫЗЫВАЮЩУЮ ИСПОЛЬЗУЕТСЯ ОПЕРАТОР "RETURN", КОТОРЫЙ В СЛУЧАЕ АВТОНОМНОГО ИСПОЛНЕНИЯ ПРОГРАММЫ ОПРЕДЕЛЯЕТ ЕЕ ОКОНЧАНИЕ.

ДОПУСКАЕТСЯ ОБРАЩЕНИЕ К ОДНОЙ И ТОЙ ЖЕ ВЫЗЫВАЕМОЙ ПРОГРАММЕ КАК К ПОДПРОГРАММЕ И КАК К ПОДПРОГРАММЕ-ФУНКЦИИ ИЗ ОДНОЙ ВЫЗЫВАЮЩЕЙ ПРОГРАММЫ. В ПОСЛЕДНЕМ СЛУЧАЕ СКАЛЯРНОЕ ВЕЩЕСТВЕННОЕ ЗНАЧЕНИЕ ЗАДАЕТСЯ УКАЗАТЕЛЮ ПОДПРОГРАММЫ-ФУНКЦИИ, КОТОРЫЙ МОЖЕТ ИСПОЛЬЗОВАТЬСЯ КАК АРГУМЕНТ ЛЮБОГО ВЫРАЖЕНИЯ.

КРОМЕ ВЫЗЫВАЕМЫХ ПРОГРАММ, КОТОРЫЕ ПО ОТНОШЕНИЮ ВЫЗЫВАЮЩЕЙ ПРОГРАММЕ РАССМАТРИВАЮТСЯ КАК ВНЕШНИЕ, В ЛЮБОЙ ПРОГРАММЕ МОГУТ ЕЩЕ ИСПОЛЬЗОВАТЬСЯ ВЛОЖЕННЫЕ (ВНУТРЕННИЕ) ПРОГРАММЫ, А ТАКЖЕ ОПЕРАТОР-ФУНКЦИИ. ПРИМЕНЕНИЕ ВЛОЖЕННЫХ ПРОГРАММ И ОПЕРАТОР-ФУНКЦИЙ ВМЕСТО ВНЕШНИХ В РЯДЕ СЛУЧАЕВ БОЛЕЕ ЦЕЛЕСООБРАЗНО, ТАК КА НАРЯДУ С ФОРМАЛЬНЫМИ ПАРАМЕТРАМИ В КАЧЕСТВЕ АРГУМЕНТОВ СВОИХ ВЫРАЖЕНИЙ ОНИ МОГУТ ИСПОЛЬЗОВАТЬ ГЛОБАЛЬНЫЕ ПЕРЕМЕННЫЕ ВЫЗЫВАЮЩЕЙ ПРОГРАММЫ.

ОБМЕН ИНФОРМАЦИЕЙ МЕЖДУ ВНЕШНИМИ ПО ОТНОШЕНИЮ ДРУГ К ДРУГУ ПРОГРАММАМИ МОЖЕТ ТАКЖЕ ПРОИЗВОДИТЬСЯ ЧЕРЕЗ ОБЪЕКТЫ, ВКЛЮЧЕННЫЕ В ОПЕРАТОР ОПИСАНИЯ ОБЩЕЙ ОБЛАСТИ ПРОГРАММ И ЧЕРЕЗ ОБЪЕКТЫ ОБЩЕЙ ОБЛАСТИ РАСШИРЕННОЙ ПАМЯТИ.

ВСЕ ИСХОДНЫЕ МОДУЛИ ПРОГРАММ ХРАНЯТСЯ ПОСЛЕ ИХ ВВОДА В БИБЛЕОТЕКЕ МОНИТОРА РАЗДЕЛЬНО И ПРИСОЕДИНЯЮТСЯ К КОМПИЛИРУЕМОЙ ПРОГРАММЕ АВТОМАТИЧЕСКИ ПО МЕРЕ ВЫЯВЛЕНИЯ ОБРАЩЕНИЙ К НИМ ИЗ ВЫЗЫВАЮЩИХ ПРОГРАММ ИЛИ В ПОСЛЕДОВАТЕЛЬНОСТИ, УСТАНОВЛИВАЕМОЙ В ДИРЕКТИВЕ КОМПИЛЯЦИИ. ПОСЛЕ ЗАВЕРШЕНИЯ КОМПИЛЯЦИИ В РАБОЧЕ ОБЛАСТИ ИСХОДНЫХ ПРОГРАММ НАХОДИТСЯ ПОЛНОСТЬЮ СОБРАННЫЙ ИСХОДНЫЙ МОДУЛЬ КОМПИЛИРУЕМОЙ ПРОГРАММЫ, ВКЛЮЧАЯ ИСХОДНЫЕ МОДУЛИ ВСЕХ ВЫЗЫВАЕМЫХ ПРОГРАММ.

## 15.1. ОФОРМЛЕНИЕ ПРОГРАММ.

~~~~~

ИСХОДНЫЙ МОДУЛЬ ЛЮБОЙ ПРОГРАММЫ НАЧИНАЕТСЯ С ЗАГОЛОВКА, ЗА КОТОРЫМ СЛЕДУЮТ СТРОКИ, ВКЛЮЧАЮЩИЕ ОПИСАНИЯ ОБЪЕКТОВ ПРОГРАММЫ (ЕСЛИ НЕОБХОДИМО) И ОПЕРАТОРЫ, ОПРЕДЕЛЯЮЩИЕ ЕЕ ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ. ИСХОДНЫЙ МОДУЛЬ ЗАВЕРШАЕТСЯ ОПЕРАТОРОМ "END".

1. ЗАГОЛОВОК ПРОГРАММЫ.

~~~~~

ЗАГОЛОВОК ПРОГРАММЫ ВКЛЮЧАЕТ:

- а) . СЛУЖЕБНОЕ СЛОВО "PROGRAM"
  - ВЫВОДИТСЯ АВТОМАТИЧЕСКИ В НАЧАЛЕ ВВОДА ИСХОДНОГО МОДУЛЯ (ДИРЕКТИВА "МОС:Ввод");
  - ПРИ ВВОДЕ ЗАГОЛОВКА ВЛОЖЕННОЙ ПРОГРАММЫ ОН ИДЕНТИФИЦИРУЕТСЯ ПО ПЕРВОМУ СИМВОЛУ "P" (ДЛЯ РУССКОЙ НОТАЦИИ - ПО ДВУМ "ПР") С АВТОМАТИЧЕСКИМ ДОПОЛНЕНИЕМ;
- б) . СИМВОЛ "ПРОБЕЛ" ЗА СЛОВОМ "PROGRAM"
  - ВЫВОДИТСЯ АВТОМАТИЧЕСКИ В ЗАГОЛОВКЕ ИСХОДНОГО МОДУЛЯ;
  - ВВОДИТСЯ ПОЛЬЗОВАТЕЛЕМ В ЗАГОЛОВКЕ ВЛОЖЕННОЙ ПРОГРАММЫ;
- в) . ИМЯ ПРОГРАММЫ
  - НАЧИНАЮЩАЯСЯ С БУКВЫ ПОСЛЕДОВАТЕЛЬНОСТЬ БУКВ И ЦИФР, ВКЛЮЧАЯ СИМВОЛ "\_", ИСПОЛЬЗУЕМЫЙ КАК РАЗДЕЛИТЕЛЬ;
- г) . КОД ПРОГРАММЫ
  - ЦЕЛОЕ ДЕСЯТИЧНОЕ ЧИСЛО < 256, ОТДЕЛЯЕТСЯ ОТ ИМЕНИ СИМВОЛОМ "/", ИСПОЛЬЗУЕТСЯ ТОЛЬКО В ИСХОДНЫХ МОДУЛЯХ ДЛЯ СОЗДАНИЯ ПАКЕТОВ И СЕГМЕНТИРОВАННЫХ ПРОГРАММ;
- д) . СПИСОК ПАРАМЕТРОВ (МОЖЕТ ОТСУТСТВОВАТЬ)
  - ЗАКЛЮЧЕННАЯ В КРУГЛЫЕ СКОБКИ "()" СТРОКА ОПИСАНИЙ ОБЪЕКТОВ, ЯВЛЯЮЩИХСЯ ФОРМАЛЬНЫМИ ПАРАМЕТРАМИ ПРИ ИСПОЛЬЗОВАНИИ ПРОГРАММЫ В КАЧЕСТВЕ ПОДПРОГРАММЫ.

В СПИСКЕ ПАРАМЕТРОВ ПРОГРАММЫ МОГУТ БЫТЬ ЛЮБЫЕ ОБЪЕКТЫ, ОПИСЫВАЕМЫЕ В ПРОГРАММАХ ВТ-МХТИ: ВЕЩЕСТВЕННЫЕ И ЦЕЛЫЕ ПРОСТЫЕ ПЕРЕМЕННЫЕ И МАССИВЫ, ИМЕНА ПРОГРАММ И МЕТОК.

## 2. ВВОД ЗНАЧЕНИЙ ПАРАМЕТРОВ ПРОГРАММЫ.

~~~~~

PROGRAM:ТЕСТ,ПРОБА,ПРИМА; INT:N

ЗНАЧЕНИЯ ПАРАМЕТРОВ ПРОГРАММЫ, ЗА ИСКЛЮЧЕНИЕМ ИМЕН ПРОГРАММ И МЕТОК МОГУТ БЫТЬ ЗАДАНЫ ПРИ АВТОНОМНОМ ИСПОЛНЕНИИ ПРОГРАММЫ. ДЛЯ ЧЕГО В ВТ-МХТИ ПРЕДУСМОТРЕНЫ СЛЕДУЮЩИЕ СРЕДСТВА:

1. СКАЛЯРНЫЕ ПАРАМЕТРЫ ПРОГРАММЫ, ЗНАЧЕНИЯ КОТОРЫХ ДОЛЖНЫ БЫТЬ ОПРЕДЕЛЕНЫ ПРИ ВХОДЕ В ПРОГРАММУ, ВЫДЕЛЯЮТ В ОТДЕЛЬНОЕ ОПИСАНИЕ И ПОМЕЧАЮТ СИМВОЛОМ ". НАПРИМЕР:

```
PROGRAM ТЕСТ (INT:I,J; INT:K"; R:X; R:Z")
END ПРОГРАММЫ ТЕСТ
```

ПРИ АВТОНОМНОМ ИСПОЛНЕНИИ ПРОГРАММЫ ЗДЕСЬ БУДЕТ ВЫПОЛНЕН БЕСФОРМАТНЫЙ ВВОД ЗНАЧЕНИЙ ПЕРЕМЕННЫХ "K" И "Z".

2. ЭЛЕМЕНТАМИ СПИСКА ПАРАМЕТРОВ МОГУТ БЫТЬ ПРОИЗВОЛЬНЫЕ СТРОКИ ОБЪЕКТОВ ОПЕРАТОРОВ "ВВОДА/ВЫВОДА", ВКЛЮЧАЯ УКАЗАТЕЛЬ КАНАЛА И ФОРМАТНУЮ ЧАСТЬ, ЗАКЛЮЧЕННЫЕ В ПАРУ СИМВОЛОВ " ", С ПОМОЩЬЮ КОТОРЫХ МОЖНО ОРГАНИЗОВАТЬ ВВОД ЗНАЧЕНИЙ ВСЕХ ПАРАМЕТРОВ ПРОГРАММЫ В ЛЮБОМ ФОРМАТЕ.

НАПРИМЕР:

```
PROGRAM ПРОБА ('/' ВВОД ПАРАМЕТРОВ ПРОГРАММЫ ПРОБА '"; I:K,J; '/'  
ЧИСЛО ЭЛЕМЕНТОВ МАССИВА 'K;"; R:X[K]; "(Z1) '/' ВВОД МАССИВА X[K  
] 'J:=1; /K: ('X['<J; ']='>X[J]; J:=J+1) "  
END ПРОБА
```

ПРОТОКОЛ ВЫПОЛНЕНИЯ ЭТОГО ЗАГОЛОВКА ИМЕЕТ ВИД:

```
ВВОД ПАРАМЕТРОВ ПРОГРАММЫ ПРОБА  
ЧИСЛО ЭЛЕМЕНТОВ МАССИВА K=5  
ВВОД МАССИВА X[K]  
X[1]=2.3 X[2]=5.1 X[3]=7.5 X[4]=8.9 X[5]=4.7
```

3. ДЛЯ ВВОДА/ВЫВОДА ПАРАМЕТРОВ ПРОГРАММЫ МОГУТ ИСПОЛЬЗОВАТЬСЯ ОПЕРАТОРЫ С ПРИЗНАКОМ " ", КОТРЫЕ ВЫПОЛНЯЮТСЯ ТОЛЬКО ПРИ АВТОНОМНОМ ИСПОЛНЕНИИ ПРОГРАММЫ.

НАПРИМЕР:

```
PROGRAM ПРИМА (I:K"; R:X[K], Y[K])  
INTEGER: J  
READ "(Z1) '/' ВВОД МАССИВА X[K] '/' X  
Y=EXP(X)  
WRITE "/Y  
END ПРИМЕРА
```

3. ОБЩИЕ ПРАВИЛА ПОСТРОЕНИЯ ОПИСАНИЯ ОБЪЕКТОВ ПРОГРАММЫ.

В ПРОГРАММАХ ВТ-МХТИ ОПИСАНИЯ ОБЪЕКТОВ МОГУТ РАЗМЕЩАТЬСЯ В ЛЮБОМ МЕСТЕ. ДЛЯ ВЕЩЕСТВЕННЫХ И ЦЕЛЫХ ПРОСТЫХ ПЕРЕМЕННЫХ ВОЗМОЖЕН ТАКЖЕ РЕЖИМ КОМПИЛЯТОРА С ОПИСАНИЕМ ИХ ПО УМОЛЧАНИЮ. В ЛЮБОМ СЛУЧАЕ ОПИСАНИЯ ПРОСТЫХ ПЕРЕМЕННЫХ ПРОГРАММЫ ДОЛЖНО ПРЕДШЕСТВОВАТЬ ОПИСАНИЯМ ПРОСТЫХ ПЕРЕМЕННЫХ, КОТОРЫЕ МОГУТ БЫТЬ ВО ВСТРОЕННЫХ ПОДПРОГРАММАХ И ВЛОЖЕННЫХ ПРОГРАММАХ. В ОТНОШЕНИИ ОСТАЛЬНЫХ ОПИСАНИЙ, ВКЛЮЧАЯ ОПИСАНИЯ МАССИВОВ, ИМЕН ПРОГРАММ И МЕТОК, НИКАКИХ ОГРАНИЧЕНИЙ НЕ НАКЛАДЫВАЕТСЯ.

ИМЕНА ВСЕХ ПРОГРАММ И ОПЕРАТОР-ФУНКЦИЙ, К КОТОРЫМ ОБРАЩАЕТСЯ ДАННАЯ ПРОГРАММА, А ТАКЖЕ ИМЕНА СИМВОЛЬНЫХ МЕТОК, ИСПОЛЬЗУЕМЫХ В ПРОГРАММЕ, (КРОМЕ УКАЗАННЫХ В СПИСКЕ ПАРАМЕТРОВ И В ОПИСАНИИ ОБЩЕЙ ОБЛАСТИ) ДОЛЖНЫ БЫТЬ ВКЛЮЧЕНЫ В СООТВЕТСТВУЮЩИЕ ОПИСАНИЯ.

НАПРИМЕР:

```
LABEL: ТОЧКА, КОНЕЦ; PROG: ВЛОЖ, КРАХ, FUN1, FUN2  
INTEGER: K, M[N, 10]; R: X, Y, A[N, N]; CHAR: ТЕКСТ(60)
```

ИМЕНА ПРОГРАММ И МЕТОК В ОПИСАНИИ ОБЩЕЙ ОБЛАСТИ ВЫЗЫВАЕМОЙ ПРОГРАММЫ ДОЛЖНЫ БЫТЬ ИДЕНТИЧНЫ АНАЛОГИЧНЫМ ОБЪЕКТАМ, ЗАДАННЫМ В ОПИСАНИИ ОБЩЕЙ ОБЛАСТИ ВЫЗЫВАЮЩЕЙ ПРОГРАММЫ. ИМЕНА ОСТАЛЬНЫХ ОБЪЕКТОВ МОГУТ БЫТЬ ВЫБРАНЫ ПРОИЗВОЛЬНЫМИ, НО ОТЛИЧНЫМИ ОТ ИМЕН ПАРАМЕТРОВ ПРОГРАММЫ.

ПРИ КОМПИЛЯЦИИ ВЫЗЫВАЮЩЕЙ ПРОГРАММЫ СОВМЕСТНО С ВЫЗЫВАЕМЫМИ ПРОИЗВОДИТСЯ КОНТРОЛЬ СООТВЕТСТВИЯ ТИПОВ ОБЪЕКТОВ ОБЩЕЙ ОБЛАСТИ ВЫЗЫВАЕМОЙ ПРОГРАММЫ С ОБЩЕЙ ОБЛАСТЬЮ ВЫЗЫВАЮЩЕЙ. ПРИ НАРУШЕНИИ СООТВЕТСТВИЯ ДАЕТСЯ СООБЩЕНИЕ ОБ ОШИБКЕ.

ОПЕРАТОР ОПИСАНИЯ ОБЩЕЙ ОБЛАСТИ В ВЫЗЫВАЕМЫХ ПРОГРАММАХ ИМЕЕТ ТУ ОСОБЕННОСТЬ, ЧТО В НЕМ МОГУТ ПЕРЕДАВАТЬСЯ ОДНОВРЕМЕННО И ГРАНИЦЫ ИНДЕКСОВ МАССИВОВ И ИХ ОПИСАНИЯ (КАК И В СЛУЧАЕ ПЕРЕДАЧИ ПАРАМЕТРОВ ПРОГРАММЫ). ПОСКОЛЬКУ ПРИ АВТОНОМНОМ ИСПОЛНЕНИИ ПРОГРАММЫ НЕОБХОДИМО ЗАДАНИЕ ГРАНИЦ ИНДЕКСОВ ПЕРЕД ОПИСАНИЕМ МАССИВОВ, В ОПЕРАТОРЕ ОБЩЕЙ ОБЛАСТИ ПРЕДУСМОТРЕНА ВОЗМОЖНОСТЬ ИСПОЛЬЗОВАНИЯ ТЕХ ЖЕ СРЕДСТВ ВВОДА, ЧТО И В СПИСКЕ ПАРАМЕТРОВ ПРОГРАММЫ.

НАПРИМЕР:

```
INTEGER: I, J
COMMON I: N"; R: S, W[N, N]; "/'ВВОД МАССИВА W[N, N]' I:=1; N: (/J:=1;
N: ( W[I, J]; J:=J+1); I:=I+1)"; C: СТРОКА(60); " (A) /'ВВОД ТЕКСТ
АСТРОКИ' / ': 'СТРОКА;"; P: ПУМА; L: ОШИБКА
```

ЗДЕСЬ ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА БУДЕТ ВВОД ЗНАЧЕНИЯ "N", ЗАТЕМ ФОРМАТИЗОВАННЫЙ ВВОД МАССИВА W[N, N] И СТРОКИ СИМВОЛОВ.

РАССМОТРЕННЫЕ СРЕДСТВА ВВОДА ЗНАЧЕНИЙ ПЕРЕМЕННЫХ ДЛЯ ОБЪЕКТОВ СПИСКА ПАРАМЕТРОВ ПРОГРАММЫ И ОПИСАНИЯ ОБЩЕЙ ОБЛАСТИ ПОЛНОСТЬЮ РЕШАЮТ ПРОБЛЕМУ АВТОНОМНОГО ИСПОЛНЕНИЯ ПРОГРАММ С ПАРАМЕТРАМИ И С ИСПОЛЬЗОВАНИЕМ ДАННЫХ ОБЩЕЙ ОБЛАСТИ, ЧТО ДЕЛАЕТ ИЗЛИШНИМ ДЕЛЕНИЕ ИСХОДНЫХ МОДУЛЕЙ НА ОСНОВНЫЕ ПРОГРАММЫ И ПОДПРОГРАММЫ, КАК ЭТО ПРИНЯТО В БОЛЬШИНСТВЕ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ.

PP

4. ОБРАЩЕНИЯ К ПРОГРАММАМ.

~~~~~

ОБРАЩЕНИЕ К ПРОГРАММЕ СОСТОИТ ИЗ ИМЕНИ, ЗА КОТОРЫМ УКАЗАН СПИСОК ФАКТИЧЕСКИХ ПАРАМЕТРОВ, ЗАКЛЮЧЕННЫЙ В КРУГЛЫЕ СКОБКИ "()", СООТВЕТСТВУЮЩИЙ СПИСКУ ПАРАМЕТРОВ ВЫЗЫВАЕМОЙ ПРОГРАММЫ. ПРИ ОТСУТСТВИИ У ПРОГРАММЫ ПАРАМЕТРОВ, ОБРАЩЕНИЕ СОСТОИТ ТОЛЬКО ИЗ ИМЕНИ.

ОБРАЩЕНИЕ МОЖЕТ ПОМЕЩАТЬСЯ В ЛЮБОМ МЕСТЕ СТРОКИ ОПЕРАТОРОВ, А ПРИ НАЛИЧИИ СПИСКА ПАРАМЕТРОВ МОЖЕТ ТАКЖЕ ИСПОЛЬЗОВАТЬСЯ, КАК УКАЗАТЕЛЬ ФУНКЦИИ, В КАЧЕСТВЕ СКАЛЯРНОГО ОПЕРАНДА ВЫРАЖЕНИЯ. В ВЫЗЫВАЮЩЕЙ ПРОГРАММЕ ДОПУСТИМО ИСПОЛЬЗОВАТЬ ОБА СПОСОБА ОБРАЩЕНИЯ К ОДНОЙ И ТОЙ ЖЕ ПРОГРАММЕ ПРИ СООТВЕТСТВУЮЩЕМ ЕЕ ОФОРМЛЕНИИ.

ОБРАЩЕНИЕ К ОПЕРАТОР-ФУНКЦИИ АНАЛОГИЧНО ОБРАЩЕНИЮ К ПРОГРАММЕ С ПАРАМЕТРАМИ И, КАК ПРАВИЛО, ИСПОЛЬЗУЕТСЯ В КАЧЕСТВЕ ОПЕРАНДА СКАЛЯРНОГО ВЫРАЖЕНИЯ.

ОБРАЩЕНИЕ БЕЗ ПАРАМЕТРОВ МОЖЕТ ПОМЕЩАТЬСЯ ТОЛЬКО В СТРОКЕ ОПЕРАТОРОВ И ПРИМЕНЯЕТСЯ ДЛЯ ВЫЗОВА ВСТРОЕННЫХ ИЛИ ВНЕШНИХ ПРОГРАММ БЕЗ ПАРАМЕТРОВ, КОТОРЫЕ ИСПОЛЬЗУЮТ, СООТВЕТСТВЕННО, ГЛОБАЛЬНЫЕ ОБЪЕКТЫ ВЫЗЫВАЮЩЕЙ ПРОГРАММЫ ИЛИ ОБЪЕКТЫ, ВКЛЮЧЕННЫЕ В ОПИСАНИЕ ОБЩЕЙ ОБЛАСТИ.

ОБЪЕКТЫ СПИСКА ФАКТИЧЕСКИХ ПАРАМЕТРОВ ДОЛЖНЫ БЫТЬ ОПИСАНЫ В ВЫЗЫВАЮЩЕЙ ПРОГРАММЕ ИЛИ ЯВЛЯТЬСЯ ЕЕ ПАРАМЕТРАМИ. ФАКТИЧЕСКИМИ ПАРАМЕТРАМИ СПИСКА ОБРАЩЕНИЯ МОГУТ БЫТЬ:

- ИМЕНА ВЕЩЕСТВЕННЫХ И ЦЕЛЫХ ПРОСТЫХ ПЕРЕМЕННЫХ;
- ИМЕНА ВЕЩЕСТВЕННЫХ И ЦЕЛЫХ МАССИВОВ;
- ИМЕНА СИМВОЛЬНЫХ ПЕРЕМЕННЫХ;
- ИМЕНА ПРОГРАММ;
- ИМЕНА МЕТОК;
- СКАЛЯРНЫЕ ВЫРАЖЕНИЯ.

В СКАЛЯРНЫХ ВЫРАЖЕНИЯХ ДОПУСКАЮТСЯ ОБРАЩЕНИЯ К ВЫЗЫВАЕМОЙ ПРОГРАММЕ, В КОТОРЫХ В СВОЮ ОЧЕРЕДЬ ТАКЖЕ МОГУТ БЫТЬ ОБРАЩЕНИЯ К НЕЙ И Т.Д. УРОВЕНЬ ВЛОЖЕНИЯ ПРИ ЭТОМ НЕ ОГРАНИЧЕН.

В ВЫЗЫВАЕМОЙ ПРОГРАММЕ ЗНАЧЕНИЯ ФАКТИЧЕСКИХ ПАРАМЕТРОВ МОГУТ БЫТЬ ИЗМЕНЕНЫ, КРОМЕ ЗНАЧЕНИЙ СКАЛЯРНЫХ ВЫРАЖЕНИЙ И ПЕРЕМЕННЫХ С ИНДЕСАМИ, КОТОРЫЕ РАССМАТРИВАЮТСЯ КАК ЧАСТНЫЙ СЛУЧАЙ СКАЛЯРНОГО ВЫРАЖЕНИЯ.

ПОРЯДОК СЛЕДОВАНИЯ И ТИП ЗНАЧЕНИЙ ОБЪЕКТОВ СПИСКА ПАРАМЕТРОВ ОБРАЩЕНИЯ ДОЛЖЕН СООТВЕТСТВОВАТЬ СПИСКУ ПАРАМЕТРОВ ВЫЗЫВАЕМОЙ ПРОГРАММЫ. СКАЛЯРНЫЕ ВЫРАЖЕНИЯ В ОБРАЩЕНИИ ПРИ ЭТО ДОЛЖНЫ БЫТЬ ТОГО ЖЕ ТИПА, ЧТО И СООТВЕТСТВУЮЩИЕ ИМ ПАРАМЕТРЫ ВЫЗЫВАЕМОЙ ПРОГРАММЫ. НАРУШЕНИЕ ЭТИХ ТРЕБОВАНИЙ ПРИВОДИТ К НЕПРЕДСКАЗУЕМЫМ РЕЗУЛЬТАТАМ ПРИ ИСПОЛНЕНИИ ПРОГРАММЫ. ПРИ КОМПИЛЯЦИИ КОНТРОЛИРУЕТСЯ ТОЛЬКО СООТВЕТСТВИЕ КОЛИЧЕСТВА ПАРАМЕТРОВ В ОБРАЩЕНИИ И ВЫЗЫВАЕМОЙ ПРОГРАММЕ. ПОРЯДОК ИХ СЛЕДОВАНИЯ И СООТВЕТСТВИЕ ТИПОВ ДОЛЖНО КОНТРОЛИРОВАТЬСЯ САМИМ ПОЛЬЗОВАТЕЛЕМ, ПОСКОЛЬКУ ТАКОЙ КОНТРОЛЬ ПРИ КОМПИЛЯЦИИ И ИСПОЛНЕНИИ РАБОЧЕЙ ПРОГРАММЫ НЕ ПРЕДУСМОТРЕН.

ПРИМЕРЫ:

```
REAL:V[4,4]
A:=5.;B:=12.;ВЛОЖ(N,A);Y:=FUN1(2,3,A)*FUN2(3,2,A[1,2],B,FUN1)
КРАХ(X,A,Y*FUN1(2,2,B),ВЛОЖ,МИНУС)
READ(N,A)
```

В ПРИВЕДЕННЫХ ПРИМЕРАХ ПРЕДСТАВЛЕНЫ РАЗЛИЧНЫЕ СПОСОБЫ ЗАДАНИЯ ФАКТИЧЕСКИХ ПАРАМЕТРОВ, ВКЛЮЧАЯ ПЕРЕДАЧУ В ВЫЗЫВАЕМУЮ ПРОГРАММУ ИМЕН ПРОГРАММ (FUN1, ВЛОЖ) И МЕТКИ (МИНУС).

#### 5. ВЫХОД ИЗ ПРОГРАММЫ.

~~~~~

ПРИ НОРМАЛЬНОМ ИСПОЛНЕНИИ АВТОНОМНОЙ ПРОГРАММЫ ВЫХОД ИЗ НЕЕ ПРОИСХОДИТ ЧЕРЕЗ ЗАВЕРШАЮЩИЙ ПРОГРАММУ ОПЕРАТОР "END". ЕСЛИ К ПРОГРАММЕ ПРОИЗОШЛО ОБРАЩЕНИЕ ИЗ ДРУГОЙ ПРОГРАММЫ, ТО ПОСЛЕ ЕЕ ВЫПОЛНЕНИЯ ЗАВЕРШАЮЩИЙ ОПЕРАТОР "END" ОБЕСПЕЧИВАЕТ ВОЗВРАТ К ВЫЗЫВАЮЩЕЙ ПРОГРАММЕ И ОДНОВРЕМЕННО ОСВОБОЖДАЕТ ПАМЯТЬ, КОТОРУЮ ЗАНИМАЛИ МАССИВЫ ВЫЗВАННОЙ ПРОГРАММЫ.

ЛЮБОЙ ДРУГОЙ ВЫХОД ИЗ ПРОГРАММЫ, НАПРИМЕР, ПРИ ВЫПОЛНЕНИИ НЕКОТОРОГО УСЛОВИЯ ИЛИ ОКОНЧАНИЯ ОДНОГО ИЗ ВОЗМОЖНЫХ ПУТЕЙ РАСЧЕТОВ МОЖЕТ БЫТЬ ОФОРМЛЕН ЛЮБЫМ ОПЕРАТОРОМ ПЕРЕХОДА НА ЗАВЕРШАЮЩИЙ ОПЕРАТОР "END", ДЛЯ ЧЕГО ЭТОТ ОПЕРАТОР МОЖЕТ БЫТЬ СНАБЖЕН ЧИСЛОВОЙ МЕТКОЙ. НА ОПЕРАТОР "END" МОЖНО ВЫЙТИ И ПО СИМВОЛЬНОЙ МЕТКЕ РАСПОЛОЖЕННОЙ ПЕРЕД НИМ.

ЭТОЙ ЖЕ ЦЕЛИ СЛУЖИТ ОПЕРАТОР "RETURN", ЗАНИМАЮЩИЙ В ПРОГРАММЕ ОТДЕЛЬНУЮ СТРОКУ И ВЫПОЛНЯЮЩИЙ ПЕРЕХОД НА КОНЕЦ ПРОГРАММЫ. КРОМЕ ТОГО, ОПЕРАТОР "RETURN" МОЖЕТ ИСПОЛЬЗОВАТЬСЯ ДЛЯ ВЫДАЧИ РЕЗУЛЬТАТА ПРИ ИСПОЛЬЗОВАНИИ ПРОГРАММЫ В КАЧЕСТВЕ ПОДПРОГРАММЫ-ФУНКЦИИ. В ЭТОМ СЛУЧАЕ ЗА ОПЕРАТОРОМ "RETURN" В КРУГЛЫХ СКОБКАХ "()" УКАЗЫВАЕТСЯ СКАЛЯРНОЕ ВЫРАЖЕНИЕ ВЕЩЕСТВЕННОГО (!!ОБЯЗАТЕЛЬНО!!) ТИПА, ЗНАЧЕНИЕ КОТОРОГО И ОПРЕДЕЛЯЕТ РЕЗУЛЬТАТ.

НАПРИМЕР:
RETURN (X)
RETURN (X**2+3.*X+14.)

В ПЕРВОМ СЛУЧАЕ ЗНАЧЕНИЕМ ПОДПРОГРАММЫ-ФУНКЦИИ ЯВЛЯЕТСЯ ЗНАЧЕНИЕ ПЕРЕМЕННОЙ "X", КОТОРОЕ ОНА ПОЛУЧИЛА К МОМЕНТУ ВЫХОДА ИЗ ПРОГРАММЫ. ВО ВТОРОМ - ЗНАЧЕНИЕ ПОЛИНОМА ОТ "X".

ЕСЛИ ПО КАКИМ-ЛИБО ПРИЧИНАМ ВОЗВРАТ К ВЫЗЫВАЮЩЕЙ ПРОГРАММЕ ИЗ ВЫЗВАННОЙ НЕ ТРЕБУЕТСЯ И ЗАДАЧА ЗАВЕРШАЕТСЯ В ВЫЗВАННОЙ ПРОГРАММЕ, ТО В МЕСТЕ ПРИНЯТИЯ ЭТОГО РЕШЕНИЯ СТАВИТСЯ ОПЕРАТОР ПСЕВДОВОЗВРАТА "RETURNR" ПРИ ЭТОМ ВЫЗВАННАЯ ПРОГРАММА СТАНОВИТСЯ ОСНОВНОЙ И ИСПОЛНЕНИЕ ПРОГРАММЫ В ЦЕЛОМ ЗАВЕРШАЕТСЯ НА ОПЕРАТОРЕ "END" ВЫЗВАННОЙ ПРОГРАММЫ. ЕСЛИ ПРИ ЭТОМ ОБРАЩЕНИЕ ИЗ ОСНОВНОЙ ПРОГРАММЫ К ДАННОЙ ПРОХОДИТ ЧЕРЕЗ РЯД ДРУГИХ ВЫЗЫВАЕМЫХ ПРОГРАММ (НЕСКОЛЬКО УРОВНЕЙ ВЛОЖЕННОСТИ), ТО ОПЕРАТОР "RETURNR" СЛЕДУЕТ ПОВТОРИТЬ СООТВЕТСТВУЮЩЕЕ ЧИСЛО РАЗ, ЧТОБЫ ПРИВЕСТИ УРОВЕНЬ ВЛОЖЕННОСТИ ПРОГРАММ К НУЛЕВОМУ, ЧТО СООТВЕТСТВУЕТ ОСНОВНОЙ ПРОГРАММЕ.

КРОМЕ ЕСТЕСТВЕННОГО ВЫХОДА ИЗ ВЫЗВАННОЙ ПРОГРАММЫ ЧЕРЕЗ ЗАВЕРШАЮЩИЙ ОПЕРАТОР "END", В ВТ-МХТИ РАЗРЕШЕН ПЕРЕХОД В ЛЮБУЮ ПРОГРАММУ (ВЫЗЫВАЮЩУЮ ИЛИ ВЫЗЫВАЕМУЮ) ПУТЕМ ПЕРЕДАЧИ УПРАВЛЕНИЯ ПО МЕТКЕ ЧИСЛОВОЙ ИЛИ СИМВОЛЬНОЙ. ПРИ ЭТОМ ЗНАЧЕНИЕ ЧИСЛОВОЙ МЕТКИ ЗАДАЕТСЯ ЦЕЛОЙ ПЕРЕМЕННОЙ, А СИМВОЛЬНОЙ - ИМЕНЕМ В СПИСКЕ ФАКТИЧЕСКИХ ПАРАМЕТРОВ ОПЕРАТОРА ОБРАЩЕНИЯ ИЛИ ЧЕРЕЗ СООТВЕТСТВУЮЩИЕ ОБЪЕКТЫ ОБЩЕЙ ОБЛАСТИ. ПРИ ПЕРЕХОДЕ В ДРУГУЮ ПРОГРАММУ ПО МЕТКЕ ВОЗНИКАЮТ ДОВОЛЬНО СЛОЖНЫЕ ВОПРОСЫ СОГЛАСОВАНИЯ ВЫЗЫВАЕМЫХ ПРОГРАММ. ПОЭТОМУ РЕКОМЕНДУЕТСЯ ТАКИЕ ПЕРЕХОДЫ ВЫПОЛНЯТЬ ТОЛЬКО В ОСНОВНУЮ ПРОГРАММУ И В СЛУЧАЯХ, КОГДА ВОЗНИКАЮТ СИТУАЦИИ, ТРЕБУЮЩИЕ ОПРЕДЕЛЕННОЙ ОБРАБОТКИ С ПОСЛЕДУЮЩИМ ВЫХОДОМ НА ПРЕКРАЩЕНИЕ ВЫПОЛНЕНИЯ ПРОГРАММЫ (ОШИБКИ В ДАННЫХ, ПРЕВЫШЕНИЕ ЗАДАННОГО ЧИСЛА ИТЕРАЦИЙ И Т.П.).

ПРИМЕРОМ ОРГАНИЗАЦИИ ТАКИХ ПЕРЕХОДОВ ЯВЛЯЮТСЯ СЛЕДУЮЩИЕ ПРОГРАММЫ:

```
PROGRAM ВВОД (I:NN"; R:D[NN,NN])
:READ"/D
: КОНТРОЛЬ (NN,D,100,МИНУС)
:RETURN
100 :RETURN/'НУЛЕВОЙ ЭЛЕМЕНТ'
:RETURN
:LABEL МИНУС
:RETURN/'ОТРИЦАТЕЛЬНЫЙ ЭЛЕМЕНТ'
:END
```

```

PROGRAM КОНТРОЛЬ (I:NN; R:D[NN,NN]; I:НУЛЬ; L:ОТРИЦ)
:/*ПРОГРАММА "КОНТРОЛЬ" ЯВЛЯЕТСЯ В ДАННОМ СЛУЧАЕ ВНЕШНЕЙ ПОДПРОГРАММОЙ ПО ОТНОШЕНИЮ К ОСНОВНОЙ ПРОГРАММЕ "ВВОД"
:DO I:=1,NN
:.DO J:=1,NN
:..IF D[I,J]<0
:...THEN
:...GOTO ОТРИЦ
:../*ПЕРЕХОД В ОСНОВНУЮ ПРОГРАММУ К СИМВОЛЬНОЙ МЕТКЕ "МИНУС"
   = "ОТРИЦ"
:...ELSE
:...IF D[I,J]=0
:....THEN
:....GOTO НУЛЬ
:../*ПЕРЕХОД В ОСНОВНУЮ ПРОГРАММУ К ЧИСЛОВОЙ МЕТКЕ, ЗАДАННОЙ ЦЕЛОЙ ПЕРЕМЕННОЙ "НУЛЬ"=100
:....END
:...END
:..END J
:.END I
:END КОНТРОЛЬ

```

ПРИМЕРЫ ЗАПИСИ ИМЕН ПРОГРАММ СО СПИСКОМ ПАРАМЕТРОВ:

```

PROGRAM ВЛОЖ (I:NM; R:AA[NM,NM])
:END ВЛОЖ

```

```

PROGRAM КРАХ (R:Z,AA[4,4],S; P:ТТТ; L:ММММ)
:END КРАХ

```

ПРИМЕРЫ ЗАПИСИ ОПЕРАТОР-ФУНКЦИЙ:

```

FUN1 (I:II,JJ; R:G[1]) :=G[II]+G[JJ]
FUN2 (I:II,JJ; R:G[1],SS; P:FUN) :=SS*FUN (II,JJ,G)

```

PP

6. ВЛОЖЕННЫЕ ПРОГРАММЫ.

~~~~~

ВЛОЖЕННЫЕ ПРОГРАММЫ (ПОДПРОГРАММЫ) МОГУТ ПРИМЕНЯТЬСЯ В ЛЮБЫХ ПРОГРАММАХ (ОСНОВНЫХ И ВНЕШНИХ) И ЯВЛЯЮТСЯ БОЛЕЕ ГИБКИМ СРЕДСТВОМ, ЧЕМ ВСТРОЕННЫЕ ПОДПРОГРАММЫ, ПОСКОЛЬКУ МОГУТ ИМЕТЬ СПИСОК ФОРМАЛЬНЫХ ПАРАМЕТРОВ И В ТО ЖЕ ВРЕМЯ ИМЕЮТ ДОСТУП КО ВСЕМ ГЛОБАЛЬНЫМ ОБЪЕКТАМ ОСНОВНОЙ ПРОГРАММЫ. ИХ ОТЛАДКА МОЖЕТ ПРОИЗВОДИТЬСЯ НЕЗАВИСИМО, С ПОСЛЕДУЮЩИМ РАЗМЕЩЕНИЕМ В ЗАДАННОМ МЕСТЕ ОСНОВНОЙ ПРОГРАММЫ СРЕДСТВАМИ РЕДАКТИРОВАНИЯ. ПРИ ЭТОМ ВЛОЖЕННЫЕ ПРОГРАММЫ МОГУТ РАЗМЕЩАТЬСЯ В ЛЮБОМ МЕСТЕ ОСНОВНОЙ ПРОГРАММЫ, ИСКЛЮЧАЯ ОБЛАСТИ ДЕЙСТВИЯ СТРУКТУРНЫХ ОПЕРАТОРОВ (УСЛОВНЫХ "IF", ЦИКЛОВ "DO", "WHILE", ВЫБОРА "CASE", БЛОКОВ "BEGIN") И ВЛОЖЕННЫХ ПРОГРАММ (РАЗРЕШЕН ТОЛЬКО ОДИН УРОВЕНЬ ВЛОЖЕНИЯ).

ЕСТЬ ОГРАНИЧЕНИЯ НА РАЗМЕЩЕНИЕ ВЛОЖЕННОЙ ПРОГРАММЫ, ЕСЛИ К НЕЙ ИМЕЮТСЯ ОБРАЩЕНИЯ КАК К ПОДПРОГРАММЕ-ФУНКЦИИ. ТАКИЕ ВЛОЖЕННЫЕ ПРОГРАММЫ ДОЛЖНЫ РАЗМЕЩАТЬСЯ В КОНЦЕ Т.Е. ПОСЛЕ ВСЕХ ОПЕРАТОРОВ ОСНОВНОЙ ПРОГРАММЫ. ПОЭТОМУ РЕКОМЕНДУЕТСЯ ВСЕ ВЛОЖЕННЫЕ ПРОГРАММЫ (НЕЗАВИСИМО ОТ СПОСОБА ОБРАЩЕНИЯ К НИМ) РАЗМЕЩАТЬ ПЕРЕД ОПЕРАТОРОМ "END" ОСНОВНОЙ ПРОГРАММЫ.

В ОТЛИЧИЕ ОТ ВСТРОЕННЫХ ПОДПРОГРАММ, ВЛОЖЕННЫЕ ПРОГРАММЫ НЕ ОКАЗЫВАЮТ ВЛИЯНИЯ НА ПОРЯДОК ВЫПОЛНЕНИЯ ОПЕРАТОРОВ, СРЕДИ КОТОРЫХ ОНИ НАХОДЯТСЯ, И ПОЭТОМУ НЕ ТРЕБУЕТСЯ СПЕЦИАЛЬНЫХ ПРИЕМОМ ИХ ОБХОДА.

СТРУКТУРА ВЛОЖЕННОЙ ПРОГРАММЫ ПОЛНОСТЬЮ АНАЛОГИЧНА СТРУКТУРЕ АВТОНОМНОЙ ПРОГРАММЫ, ЗА ИСКЛЮЧЕНИЕМ ЗАПРЕТА НА ИСПОЛЬЗОВАНИЕ В НЕЙ ДРУГИХ ВЛОЖЕННЫХ ПРОГРАММ И ОПИСАНИЯ ОБЩЕЙ ОБЛАСТИ.

ИМЕНА ОБЪЕКТОВ СПИСКА ФОРМАЛЬНЫХ ПАРАМЕТРОВ ВЛОЖЕННОЙ ПРОГРАММЫ ДОЛЖНЫ ОТЛИЧАТЬСЯ ОТ ИМЕН ГЛОБАЛЬНЫХ ОБЪЕКТОВ ОСНОВНОЙ ПРОГРАММЫ. ОДНАКО, В СПИСКАХ ПАРАМЕТРОВ РАЗНЫХ ВЛОЖЕННЫХ ПРОГРАММ ОДНИ И ТЕ ЖЕ ИМЕНА МОГУТ ПОВТОРЯТЬСЯ ДАЖЕ ПРИ ОПИСАНИИ РАЗЛИЧНЫХ ОБЪЕКТОВ. СПИСОК ФОРМАЛЬНЫХ ПАРАМЕТРОВ ВЛОЖЕННОЙ ПРОГРАММЫ ЛОКАЛИЗОВАН В ЕЕ ПРЕДЕЛАХ, КАК И ВСЕ ОСТАЛЬНЫЕ ЕЕ ОПИСАНИЯ.

ПРИМЕР:

```
PROGRAM ВЛОЖ (I:NN; R:Z [N,N])
:DO I:=1,NN
:.DO J:=1,NN
:..IF J<I
:... Z [I,J] :=I+J
:...ELSE
:... Z [I,J] :=X*I+Y*J
:...END
:..END J
:.END I
:END
```

В ПРИВЕДЕННОЙ ПРОГРАММЕ ИСПОЛЬЗУЮТСЯ ГЛОБАЛЬНЫЕ ПЕРЕМЕННЫЕ "X" И "Y".

ПОСКОЛЬКУ ВЛОЖЕННОЙ ПРОГРАММЕ ДОСТУПНЫ ГЛОБАЛЬНЫЕ ОБЪЕКТЫ ОСНОВНОЙ ПРОГРАММЫ, НА ПЕРИОД АВТОНОМНОЙ ОТЛАДКИ ОНИ МОУТ БЫТЬ ВКЛЮЧЕНЫ В ОПИСАНИЕ ОБЩЕЙ ОБЛАСТИ, КОТОРАЯ ИСКЛЮЧАЕТСЯ ПЕРЕД ВСТРАИВАНИЕМ ВЛОЖЕННОЙ ПРОГРАММЫ В ОСНОВНУЮ.

7. ПОДПРОГРАММЫ-ФУНКЦИИ.

~~~~~

РОЛЬ ПОДПРОГРАММЫ-ФУНКЦИИ В ВТ-МХТИ МОЖЕТ ВЫПОЛНЯТЬ ЛЮБАЯ ПРОГРАММА, ИМЕЮЩАЯ СПИСОК ПАРАМЕТРОВ. СВОЙСТВА ОПЕРАТОР-ФУНКЦИИ РЕАЛИЗУЕТСЯ ПРИ ОБРАЩЕНИИ К ЛЮБОЙ ПРОГРАММЕ УКАЗАТЕЛЕМ С ИМЕНЕМ ЭТОЙ ПРОГРАММЫ И ОБЯЗАТЕЛЬНЫМ СПИСОКОМ ФАКТИЧЕСКИХ ПАРАМЕТРОВ. ЗНАЧЕНИЕ УКАЗАТЕЛЯ ОПРЕДЕЛЕНО ПОСЛЕДНИМ ОПЕРАТОРОМ ПРИСВАИВАНИЯ ВЕЩЕСТВЕННОГО ЗНАЧЕНИЯ, КОТОРЫЙ ВЫПОЛНЯЕТСЯ ПЕРЕД ВЫХОДОМ, ИЛИ СПЕЦИАЛЬНЫМ ОПЕРАТОРОМ ВЫХОДА С УКАЗЫВАЕМЫМ ЗНАЧЕНИЕМ (СМ. 5. ВЫХОД ИЗ ПРОГРАММЫ).

8. ОПЕРАТОР-ФУНКЦИЯ.

~~~~~

ОПЕРАТОР-ФУНКЦИЯ ПРЕДСТАВЛЯЕТ СОБОЙ ОПЕРАТОР ПРИСВАИВАНИЯ, ПРАВА ЧАСТЬ КОТОРОГО ЯВЛЯЕТСЯ СКАЛЯРНЫМ ВЫРАЖЕНИЕМ ЦЕЛОГО ИЛИ ВЕЩЕСТВЕННОГО ТИПА, А В ЛЕВОЙ ЧАСТИ УКАЗАНО ЕЕ ИМЯ СО СПИСКОМ ФОРМАЛЬНЫХ ПАРАМЕТРОВ В КРУГЛЫХ СКОБКАХ "( )". СПИСОК ФОРМАЛЬНЫХ ПАРАМЕТРОВ В ОПЕРАТОР-ФУНКЦИИ ЯВЛЯЕТСЯ ОБЯЗАТЕЛЬНЫМ И СТРОИТСЯ АНАЛОГИЧНО СПИСКУ ПАРАМЕТРОВ ЛЮБОЙ ПРОГРАММЫ. СКАЛЯРНОЕ ВЫРАЖЕНИЕ В ПРАВОЙ ЧАСТИ МОЖЕТ ВКЛЮЧАТЬ В КАЧЕСТВЕ ОПЕРАНДОВ ОБЪЕКТЫ ИЗ СПИСКА ФОРМАЛЬНЫХ ПАРАМЕТРОВ, А ТАКЖЕ ЛЮБЫЕ ГЛОБАЛЬНЫЕ ОБЪЕКТЫ ОСНОВНОЙ ПРОГРАММЫ, ВКЛЮЧАЯ ПРОСТЫЕ И ИНДЕКСИРОВАННЫЕ ЦЕЛЫЕ И ВЕЩЕСТВЕННЫЕ ПЕРЕМЕННЫЕ, УКАЗАТЕЛИ ФУНКЦИЙ, ОПЕРАТОР-ФУНКЦИЙ, ПОДПРОГРАММ-ФУНКЦИЙ. НЕЗАВИСИМО ОТ ТИПА ЗНАЧЕНИЯ ВЫРАЖЕНИЯ ПРАВОЙ ЧАСТИ ЗНАЧЕНИЕ ОПЕРАТОР-ФУНКЦИИ ВСЕГДА ВЕЩЕСТВЕННОЕ. КАК И ВЛОЖЕННЫЕ ПРОГРАММЫ, ОПЕРАТОР-ФУНКЦИИ ДОЛЖНЫ РАЗМЕЩАТЬСЯ В КОНЦЕ ПРОГРАММЫ.

ПРИМЕР:

```
FUN1 (I: II, JJ; R: Z [N, N]) := Z [JJ, II] * (X+Y)
```

```
FUN2 (I: L1, L2; R: P, W [N, N]; P: FUN) := P * FUN (L1, L2, W) / X
```

ЗДЕСЬ В ОПЕРАТОРА-ФУНКЦИЯХ ИСПОЛЬЗОВАНЫ ГЛОБАЛЬНЫЕ ПЕРЕМЕННЫЕ "X" И "Y", ФОРМАЛЬНЫЕ ЦЕЛЫЕ "II, JJ, L1, L2", ВЕЩЕСТВЕННАЯ "P", ВЕЩЕСТВЕННЫЕ МАССИВЫ "Z" И "W", ИМЯ ДРУГОЙ ОПЕРАТОР-ФУНКЦИИ "FUN".

PP

## 15.2. СЕГМЕНТИРОВАНИЕ ПРОГРАММ.

~~~~~

В ВТ-МХТИ ПРЕДУСМОТРЕНА ВОЗМОЖНОСТЬ ПЕРЕХОДА ОТ ВЫПОЛНЕНИЯ ОДНОЙ РАБОЧЕЙ ПРОГРАММЫ К ВЫПОЛНЕНИЮ ДРУГИХ, РАБОЧИЕ СЕГМЕНТЫ КОТОРЫХ, ПОМЕЧЕННЫЕ РАЗЛИЧНЫМИ ИДЕНТИФИЦИРУЮЩИМИ КОДАМИ, НАХОДЯТСЯ ЛИБО В БИБЛИОТЕКЕ, РАЗМЕЩЕННОЙ В РАСШИРЕННОЙ ПАМЯТИ ДЗ-28 ЛИБО НА МАГНИТНОЙ ЛЕНТЕ. В ПОСЛЕДНЕМ СЛУЧАЕ ВЫЗОВ ОЧЕРЕДНОЙ ПРОГРАММЫ ДЛЯ ИСПОЛНЕНИЯ ТРЕБУЕТ ДОСТАТОЧНО МНОГО ВРЕМЕНИ, НЕОБХОДИМОГО ДЛЯ ПОИСКА СЕГМЕНТА НА МЛ И ЕГО СЧИТЫВАНИЯ. ПОЭТОМУ БОЛЕЕ ЦЕЛЕСООБРАЗНО ПЕРЕД СОВМЕЩЕННЫМ ИСПОЛНЕНИЕМ ГРУППЫ ПРОГРАММ ОРГАНИЗОВАТЬ ИХ ЗАГРУЗКУ В БИБЛИОТЕКУ, ОТКУДА ВЫЗОВ ОЧЕРЕДНОЙ ПРОГРАММЫ ТРЕБУЕТ НЕСРАВНЕННО МЕНЬШЕГО ВРЕМЕНИ.

ЗАГРУЗКА БИБЛИОТЕКИ СЕГМЕНТАМИ РАБОЧИХ ПРОГРАММ ВЫПОЛНЯЕТСЯ ЛИБО ОПЕРАЦИОННОЙ СИСТЕМОЙ ПРИ КОМПИЛЯЦИИ ИСХОДНЫХ ПРОГРАММ ИЛИ СЧИТЫВАНИЕМ РАНЕЕ СКОМПИЛИРОВАННЫХ СЕГМЕНТОВ С МЛ, ЛИБО ПРИ ВЫПОЛНЕНИИ РАБОЧЕЙ ПРОГРАММЫ, В КОТОРОЙ ПРЕДУСМОТРЕН ОПЕРАТОР ЗАГРУЗКИ СЕГМЕНТОВ С МЛ В РАСШИРЕННУЮ ПАМЯТЬ.

ИМЕЮТСЯ ДВА РЕЖИМА ПЕРЕДАЧИ УПРАВЛЕНИЯ ОТ ОДНОЙ ПРОГРАММЫ К ДРУГОЙ:

1. НЕЗАВИСИМАЯ ПЕРЕДАЧА УПРАВЛЕНИЯ;
2. ПЕРЕДАЧА УПРАВЛЕНИЯ С ОБЩИМ ПОЛЕМ ДАННЫХ.

НЕЗАВИСИМАЯ ПЕРЕДАЧА УПРАВЛЕНИЯ ИСПОЛЬЗУЕТСЯ ДЛЯ ПЕРЕХОДА ОТ ВЫПОЛНЕНИЯ ОДНОЙ НЕЗАВИСИМОЙ ПРОГРАММЫ К ДРУГОЙ. ЭТОТ РЕЖИМ НАИБОЛЕЕ ПОДХОДИТ ДЛЯ ОРГАНИЗАЦИИ ПАКЕТОВ ПРОГРАММ, РЕАЛИЗУЮЩИХ РАЗЛИЧНЫЕ НЕСВЯЗАННЫЕ МЕЖДУ СОБОЙ ВЫЧИСЛИТЕЛЬНЫЕ ПРОЦЕССЫ. ПОЛУЧАЮЩАЯ УПРАВЛЕНИЕ ПРОГРАММА В ЭТОМ СЛУЧАЕ СОЗДАЕТ СОБСТВЕННОЕ РАСПРЕДЕЛЕНИЕ ПАМЯТИ ДЛЯ ДАННЫХ, ПРОИЗВОДИТ ВВОД ИСХОДНОЙ ИНФОРМАЦИИ, ЕЕ ПЕРЕРАБОТКУ И ВЫВОД РЕЗУЛЬТАТОВ.

РЕЖИМ ПЕРЕДАЧИ УПРАВЛЕНИЯ С ОБЩИМ ПОЛЕМ ДАННЫХ ИСПОЛЬЗУЕТСЯ ДЛЯ СОЗДАНИЯ СЕГМЕНТИРОВАННЫХ ПРОГРАММ БОЛЬШОГО ОБЪЕМА, КОТОРЫЕ НЕ МОГУТ БЫТЬ РАЗМЕЩЕНЫ ВМЕСТЕ С ДАННЫМИ В РАБОЧЕЙ ОБЛАСТИ ПАМЯТИ ДЗ-28 (СЕГМЕНТЫ ПАМЯТИ С, D, E, F). ДЛЯ СОЗДАНИЯ ОБЩЕГО ПОЛЯ ПАМЯТИ ВСЕ СЕГМЕНТЫ РАБОЧЕЙ ПРОГРАММЫ ДОЛЖНЫ ИМЕТЬ ИДЕНТИЧНЫЕ ОПИСАНИЯ ОБЩИХ ДЛЯ ВСЕХ СЕГМЕНТОВ ДАННЫХ (ИМЕНА МОГУТ ОТЛИЧАТЬСЯ) И ОДИНАКОВЫЙ НАЧАЛЬНЫЙ АДРЕС РАЗМЕЩЕНИЯ МАССИВОВ "АМ", КОТОРЫЙ СООБЩАЕТСЯ ПОСЛЕ ЗАВЕРШЕНИЯ КОМПИЛЯЦИИ КАЖДОЙ ПРОГРАММЫ РЕЖИМ РАБОТЫ ГРУППЫ СЕГМЕНТОВ С ОБЩИМ ПОЛЕМ ДАННЫХ МОЖЕТ БЫТЬ РЕАЛИЗОВАН ДВУМЯ СПОСОБАМИ:

- а). ПРИ КОМПИЛЯЦИИ ИСХОДНЫХ ПРОГРАММ СЕГМЕНТОВ ДЛЯ КАЖДОГО ИЗ НИХ ПОСЛЕ КОМПИЛЯЦИИ ЗАДАЕТСЯ ЗНАЧЕНИЕ "АМ", РАВНОЕ НАИБОЛЬШЕМУ ИЗ ВСЕХ СЕГМЕНТОВ. ПОДГОТОВЛЕННЫЕ ТАКИМ ОБРАЗОМ СЕГМЕНТЫ ПРИ УСЛОВИИ ИДЕНТИЧНОСТИ ОПИСАНИЙ В НИХ ОБЩЕГО ПОЛЯ ДАННЫХ МОГУТ ПЕРЕДАВАТЬ УПРАВЛЕНИЕ ДРУГ ДРУГУ И ОПЕРАТОРАМИ НЕЗАВИСИМОЙ ПЕРЕДАЧИ УПРАВЛЕНИЯ.
- б). ТОЛЬКО ОДНОМУ ЗАПУСКАЮЩЕМУ ПРОГРАММУ НАЧАЛЬНОМУ СЕГМЕНТУ ЗАДАЕТСЯ НАИБОЛЬШЕЕ ЗНАЧЕНИЕ "АМ". В ЭТОМ СЛУЧАЕ ДЛЯ ПЕРЕДАЧИ УПРАВЛЕНИЯ ОТ СЕГМЕНТА К СЕГМЕНТУ НЕОБХОДИМО ИСПОЛЬЗОВАТЬ ОПЕРАТОР ПЕРЕДАЧИ УПРАВЛЕНИЯ С ОБЩИМ ПОЛЕМ ДАННЫХ, КОТОРЫЙ СОХРАНЯЕТ ЗАДАННОЕ В НАЧАЛЬНОМ СЕГМЕНТЕ ЗНАЧЕНИЕ "АМ" ВО ВСЕХ ПОСЛЕДУЮЩИХ.

НИЖЕ ПРЕДСТАВЛЕНЫ ПРИМЕРЫ ОПЕРАТОРОВ ПЕРЕХОДА ОТ ПРОГРАММЫ К ПРОГРАММЕ.

1. НЕЗАВИСИМАЯ ПЕРЕДАЧА УПРАВЛЕНИЯ:

~~~~~

```
GOTO П/25:Б
GOTO Р/25:В
GOTO П/S1:Б-4
GOTO Р/SEGМ[I]:В-J
```

В ПРЕДСТАВЛЕННЫХ ПРИМЕРАХ ОПЕРАТОРОВ СИМВОЛ ПЕРЕД "/" ЕСТЬ РУССКАЯ БУКВА "П" ИЛИ ЛАТИНСКАЯ "Р", ЧТО ЯВЛЯЕТСЯ ПРИЗНАКОМ НЕЗАВИСИМОЙ ПЕРЕДАЧИ УПРАВЛЕНИЯ. ЗА "/" УКАЗЫВАЕТСЯ КОД СЕГМЕНТА, КОТОРОМУ ПЕРЕДАЕТСЯ УПРАВЛЕНИЕ. КОД ЗАДАЕТСЯ ЦЕЛЫМ ЧИСЛОМ ИЛИ ЗНАЧЕНИЕМ ЦЕЛОЙ ПЕРЕМЕННОЙ (ПРОСТОЙ ИЛИ ИНДЕКСИРОВАННОЙ). ЗА ":" СЛЕДУЕТ ПРИЗНАК, УКАЗЫВАЮЩИЙ ЧТО СЕГМЕНТ НАХОДИТСЯ В БИБЛИОТЕКЕ (РУССКАЯ "Б" ИЛИ ЛАТИНСКАЯ "В"), ЗА КОТОРЫМ ЧЕРЕЗ "-" ЗАДАЕТСЯ НОМЕР СЕГМЕНТА РАСШИРЕННОЙ ПАМЯТИ НАЧАЛА БИБЛИОТЕКИ, ГДЕ РАЗМЕЩЕН СЕГМЕНТ ПРОГРАММЫ. ЕСЛИ НОМЕР НЕ УКАЗАН (СТРОКИ 30,31), ТО ОН АВТОМАТИЧЕСКИ ПРИНИМАЕТСЯ РАВНЫМ 2.

ЕСЛИ СЕГМЕНТЫ ВЫЗЫВАЮТСЯ С МАГНИТНОЙ ЛЕНТЫ ТО ИСПОЛЬЗУЮТСЯ ОПЕРАТОРЫ С ПРИЗНАКОМ МЕСТОНАХОЖДЕНИЯ СЕГМЕНТА "Т" (РУССКАЯ ИЛИ ЛАТИНСКАЯ БУКВА):

```
GOTO П/25:Т
GOTO Р/25:Т
GOTO П/S1:Т
GOTO Р/SEGМ[I]:Т
```

2. ПЕРЕДАЧА УПРАВЛЕНИЯ С ОБЩИМ ПОЛЕМ ДАННЫХ:

~~~~~

ДЛЯ СЕГМЕНТОВ ИЗ БИБЛИОТЕКИ В РАСШИРЕННОЙ ПАМЯТИ:

```
GOTO 30:Б
GOTO S2:В
GOTO C/SEGM[I]:Б-3
GOTO S/SEGM[J]:В-J
```

ЗДЕСЬ ПЕРЕД "/" УКАЗЫВАЕТСЯ ПРИЗНАК ПЕРЕДАЧИ УПРАВЛЕНИЯ С ОБЩИМ ПОЛЕМ ПАМЯТИ (РУССКАЯ БУКВА "С" ИЛИ ЛАТИНСКАЯ "S"), КОТОРЫЙ МОЖЕТ БЫТЬ ОПУЩЕН (СТРОКИ 48,49). НАЗНАЧЕНИЕ ОСТАЛЬНЫХ ЭЛЕМЕНТОВ ОПЕРАТОРОВ РАССМОТРЕНО ВЫШЕ.

ДЛЯ СЕГМЕНТОВ, НАХОДЯЩИХСЯ НА МАГНИТНОЙ ЛЕНТЕ:

```
GOTO 30:Т
GOTO S2:Т
GOTO C/SEGM[I]:Т
GOTO S/SEGM[J]:Т
```