

Описание "Бейсик-План" из файлов k35a_index-1_index-2.txt и k35b_index-3.txt
(В конце есть пояснение о выводе на печать под управлением Бейсик-План в имитаторе ДЗ-28.
(Линии отреза и номера листов удалены, чтобы они не мешали читать это "Описание".)

С О Д Е Р Ж А Н И Е

- - - - -

1. НАЗНАЧЕНИЕ ЯЗЫКА И ПРОГРАММЫ ИНТЕРПРЕТАТОРА
2. ОСНОВНЫЕ ЭЛЕМЕНТЫ ЯЗЫКА
 - 2.1 АЛФАВИТ
 - 2.2 ФОРМЫ ПРЕДСТАВЛЕНИЯ ДАННЫХ В ОЗУ
 - 2.3 КОНСТАНТЫ
 - 2.4 ПЕРЕМЕННЫЕ
 - 2.5 ФАЙЛЫ ИНФОРМАЦИОННЫХ СТРОК
 - 2.6 ЭЛЕМЕНТЫ ИНФОРМАЦИОННЫХ СТРОК
 - 2.7 РЕГИСТРЫ
 - 2.8 СЛОВА
 - 2.9 ИНДЕКСНЫЕ ВЫРАЖЕНИЯ И ИНДЕКСЫ
 - 2.10 ФУНКЦИИ
 - 2.11 АРИФМЕТИЧЕСКИЕ ВЫРАЖЕНИЯ
 - 2.12 ТЕКСТОВЫЕ ВЫРАЖЕНИЯ
 - 2.13 ПРОГРАММА
3. ПРИСВАИВАНИЕ ЗНАЧЕНИЙ
 - 3.1 ОПЕРАТОРЫ "DIM" И "COM"
 - 3.2 ОПЕРАТОР "LET"
 - 3.3 ОПЕРАТОР "INPUT"
 - 3.4 ОПЕРАТОР "LINPUT"
 - 3.5 ОПЕРАТОР "READ"
 - 3.6 ОПЕРАТОР "CLEAR"
4. ИСПОЛЬЗОВАНИЕ НМЛ
5. ВЫВОД ИНФОРМАЦИИ НА ПЕЧАТЬ
 - 5.1 ОБЩИЕ СВЕДЕНИЯ
 - 5.2 ОПЕРАТОР "SET"
 - 5.3 ОПЕРАТОР "PRINT"
 - 5.4 ОПЕРАТОР "LIST"
 - 5.5 МАКЕТЫ ВЫВОДА ИНФОРМАЦИИ
6. УПРАВЛЕНИЕ ХОДОМ РАБОТЫ ПРОГРАММЫ
 - 6.1 ОПЕРАТОР "IF"
 - 6.2 ОРГАНИЗАЦИЯ ЦИКЛОВ
 - 6.3 БЕЗУСЛОВНЫЕ ПЕРЕХОДЫ И ПОДПРОГРАММЫ
 - 6.4 ОПЕРАТОР "RUN"
 - 6.5 ОПЕРАТОРЫ "END", "STOP" И "START"
7. ПРОЧИЕ ОПЕРАТОРЫ
 - 7.1 ОПЕРАТОР "CHANGE"
 - 7.2 ОПЕРАТОР "REN"
 - 7.3 ОПЕРАТОР "RESTORE"
 - 7.4 ОПЕРАТОР "CALL"
8. РАБОТА С ТЕКСТОВЫМИ ФАЙЛАМИ
9. ИСПОЛЬЗОВАНИЕ РАСШИРЕННОЙ ПАМЯТИ
10. ПОРЯДОК РАБОТЫ С ИНТЕРПРЕТАТОРОМ
11. ДОПОЛНИТЕЛЬНЫЕ СВЕДЕНИЯ О КОНСТРУКЦИИ ИНТЕРПРЕТАТОРА
 - 11.1 ОПИСАНИЕ СТРУКТУРЫ ИНТЕРПРЕТАТОРА
 - 11.2 ПРОСТЕЙШИЕ ВОЗМОЖНОСТИ КОМПОНОВКИ ИНТЕРПРЕТАТОРА
 - 11.3 РАСПРЕДЕЛЕНИЕ ОЗУ
 - 11.4 РАЗРАБОТКА БЛОКОВ ВНЕШНИХ ПОДПРОГРАММ
 - 11.5 ОПИСАНИЕ НЕКОТОРЫХ ПОДПРОГРАММ ИНТЕРПРЕТАТОРА

ПРИЛОЖЕНИЯ :

1. ШИФРЫ ОШИБОК И ВЕРОЯТНЫЕ ПРИЧИНЫ ИХ ВОЗНИКНОВЕНИЯ
2. НАЗНАЧЕНИЕ УПРАВЛЯЮЩИХ СИМВОЛОВ НА ДИСПЛЕЕ
3. ВАРИАНТЫ ИНТЕРПРЕТАТОРА
4. ОПИСАНИЕ БЛОКА ВНЕШНИХ ПОДПРОГРАММ

1. НАЗНАЧЕНИЕ ЯЗЫКА И ПРОГРАММЫ-ИНТЕРПРЕТАТОРА.

РАЗРАБОТАННЫЙ АЛГОРИТМИЧЕСКИЙ ЯЗЫК ПРЕДСТАВЛЯЕТ СОБОЙ ОДНУ ИЗ ВЕРСИЙ ЯЗЫКА БЕЙСИК И ЯВЛЯЕТСЯ ПРОБЛЕМНО-ОРИЕНТИРОВАННЫМ ЯЗЫКОМ ВЫСОКОГО УРОВНЯ. ГЛАВНАЯ ОРИЕНТАЦИЯ ЯЗЫКА - РЕШЕНИЕ С ПОМОЩЬЮ МИКРО-ЭВМ ЗАДАЧ ПО ТЕХНИКО-ЭКОНОМИЧЕСКОМУ ПЛАНИРОВАНИЮ. ЗАДАЧИ ТАКОГО РОДА, КАК ПРАВИЛО, СОДЕРЖАТ В СЕБЕ НЕСЛОЖНЫЕ МАТЕМАТИЧЕСКИЕ ВЫЧИСЛЕНИЯ, А ТАКЖЕ, ОБРАБОТКУ НЕБОЛЬШОГО ОБЪЕМА КАК ЧИСЛОВОЙ, ТАК И ТЕКСТОВОЙ ИНФОРМАЦИИ.

РАЗРАБОТАННЫЙ ЯЗЫК ОТЛИЧАЕТСЯ СВОЕЙ ПРОСТОТОЙ, ЧТО ПРЕДОПРЕДЕЛЯЕТСЯ СОСТАВЛЯЮЩИМ ЕГО ОСНОВУ ЯЗЫКОМ БЕЙСИК, РАСЧИТАННОМ, КАК ИЗВЕСТНО, НА НАЧИНАЮЩИХ ПРОГРАММИСТОВ. ПРИМЕНЯЕМЫЕ АНГЛИЙСКИЕ ТЕРМИНЫ, ОБОЗНАЧАЮЩИЕ СЛУЖЕБНЫЕ СЛОВА И ИМЕНА ОПЕРАТОРОВ, ПРАКТИЧЕСКИ ОГРАНИЧЕНЫ ТЕРМИНАМИ, ПРИМЕНЯЕМЫМИ В ДРУГИХ ВЕРСИЯХ БЕЙСИКА И, ПО ВОЗМОЖНОСТИ, СООТВЕТСТВУЮТ ИМ ПО СВОЕМУ НАЗНАЧЕНИЮ.

В ДАННОЙ ВЕРСИИ ЯЗЫКА, ПО СРАВНЕНИЮ С ДРУГИМИ, ПРИМЕНЯЮТСЯ БОЛЕЕ РАЗНООБРАЗНЫЕ ФОРМЫ ПРЕДСТАВЛЕНИЯ ДАННЫХ, С ДРУГОЙ СТОРОНЫ, НЕСКОЛЬКО УПРОЩЕНЫ ВОЗМОЖНЫЕ МАТЕМАТИЧЕСКИЕ ОПЕРАЦИИ, НЕ ИСПОЛЬЗУЕТСЯ РЯД ОПЕРАТОРОВ И ФУНКЦИЙ, РЕДКО ВСТРЕЧАЮЩИХСЯ В ИНЖЕНЕРНЫХ ПРАКТИЧЕСКИХ РАСЧЕТАХ.

ВЫСОКАЯ ПЛОТНОСТЬ РАЗМЕЩЕНИЯ ДАННЫХ ЯВЛЯЕТСЯ ОСНОВНОЙ ОТЛИЧИТЕЛЬНОЙ ОСОБЕННОСТЬЮ ДАННОЙ ВЕРСИИ ЯЗЫКА И ТРАНСЛИРУЮЩЕЙ ПРОГРАММЫ. ЭТО ПОЗВОЛЯЕТ, ВО-1-Х, ЭКОНОМИТЬ СРАВНИТЕЛЬНО НЕБОЛЬШОЙ ОБЪЕМ ОЗУ МИКРО-ЭВМ, ВО-2-Х, БОЛЕЕ ПЛОТНО РАСПОЛАГАТЬ ДАННЫЕ НА ВНЕШНЕМ НОСИТЕЛЕ. ПОСЛЕДНЕЕ ОБСТОЯТЕЛЬСТВО ОСОБЕННО ВАЖНО ДЛЯ ЭВМ, РАБОТАЮЩЕЙ С КОМПАКТНЫМИ МАГНИТОФОННЫМИ КАССЕТАМИ, Т.К. ВОЗРАСТАЕТ СКОРОСТЬ ОБМЕНА ИНФОРМАЦИЕЙ И НАДЕЖНОСТЬ ЕЕ ХРАНЕНИЯ.

ТРАНСЛИРУЮЩАЯ ПРОГРАММА РАЗРАБОТАНА ДЛЯ ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА НА БАЗЕ МИКРО-ЭВМ "ЭЛЕКТРОНИКА ДЗ-28". КОМПЛЕКС МОЖЕТ ВКЛЮЧАТЬ В СЕБЯ:

- ЭВМ "ЭЛЕКТРОНИКА ДЗ-28" С ОЗУ РАЗМЕРОМ 32 ИЛИ 128 К-БАЙТ;
- АЛФАВИТНО-ЦИФРОВОЙ ДИСПЛЕЙ;
- АЛФАВИТНО-ЦИФРОВОЕ ПЕЧАТАЮЩЕЕ УСТРОЙСТВО.

ТРАНСЛИРУЮЩАЯ ПРОГРАММА ВЫПОЛНЕНА В ВИДЕ ИНТЕРПРЕТАТОРА, Т.Е. ОДНОВРЕМЕННО С ТРАНСЛЯЦИЕЙ ПРОГРАММЫ ПРОИСХОДИТ ЕЕ ВЫПОЛНЕНИЕ. ЭТО ОБСТОЯТЕЛЬСТВО ТРЕБУЕТ ПОСТОЯННОГО НАЛИЧИЯ В ОЗУ ИНТЕРПРЕТИРУЮЩЕЙ ПРОГРАММЫ И ЗАМЕТНО УМЕНЬШАЕТ СКОРОСТЬ РАБОТЫ ПРОГРАММЫ, ОДНАКО, С ДРУГОЙ СТОРОНЫ, ИНТЕРПРЕТАТОР ЗНАЧИТЕЛЬНО ОБЛЕГЧАЕТ ДИАЛОГ ПОЛЬЗОВАТЕЛЯ С МАШИНОЙ.

В СОСТАВ ПРОГРАММЫ-ИНТЕРПРЕТАТОРА ВКЛЮЧЕНА ПОДПРОГРАММА-РЕДАКТОР ТЕКСТА. ЕЕ ИСПОЛЬЗОВАНИЕ ПОЗВОЛЯЕТ ПРОИЗВОДИТЬ КОРРЕКТИРОВКУ ДАННЫХ И ПРОГРАММЫ ПОЛЬЗОВАТЕЛЯ БЕЗ ПЕРЕЗАГРУЗКИ ИНФОРМАЦИИ В ОЗУ. РАБОТА ПОДПРОГРАММЫ-РЕДАКТОРА МОЖЕТ ПРОИЗВОДИТЬСЯ ПАРАЛЛЕЛЬНО ДРУГОЙ - ФОНОВОЙ ЗАДАЧЕ, ЧТО ПОЗВОЛЯЕТ БОЛЕЕ ЭФФЕКТИВНО ИСПОЛЬЗОВАТЬ ВЫЧИСЛИТЕЛЬНЫЙ КОМПЛЕКС.

2. ОСНОВНЫЕ ЭЛЕМЕНТЫ ЯЗЫКА.

2.1. АЛФАВИТ.

АЛФАВИТ ЯЗЫКА СОСТОИТ ИЗ ВСЕХ СИМВОЛОВ, ИМЕЮЩИХСЯ НА ВНЕШНИХ УСТРОЙСТВАХ СО СЛЕДУЮЩИМИ ОГРАНИЧЕНИЯМИ И ИЗМЕНЕНИЯМИ:

- НЕ ПРИМЕНЯЕТСЯ НА ДИСПЛЕЕ ПОЛЕ РЕДАКТИРОВАНИЯ;
- СИМВОЛ <ПЕРЕВОД СТРОКИ> ОЗНАЧАЕТ ПЕРЕВОД ВЫВОДА ИНФОРМАЦИИ НА НАЧАЛО СЛЕДУЮЩЕЙ СТРОКИ;

- НЕ ДОПУСКАЕТСЯ ПРИМЕНЕНИЕ УПРАВЛЯЮЩИХ СИМВОЛОВ, КРОМЕ СИМВОЛОВ, ИМЕЮЩИХ СПЕЦИАЛЬНО ОГОВОРЕННОЕ ЗНАЧЕНИЕ;

- ВМЕСТО НЕДОПУСТИМЫХ СИМВОЛОВ ИНТЕРПРЕТАТОР ВЫВОДИТ НА ЭКРАН СИМВОЛ ПРОБЕЛА.

ДЛЯ ОБОЗНАЧЕНИЯ ТЕКСТА ПРОГРАММЫ ИСПОЛЬЗУЮТСЯ ЛАТИНСКИЕ БУКВЫ. РУССКИЕ БУКВЫ ДОПУСКАЕТСЯ ПРИМЕНЯТЬ ТОЛЬКО В ТЕКСТОВЫХ КОНСТАНТАХ И КОММЕНТАРИЯХ.

КОНКРЕТНЫЙ НАБОР ИСПОЛЬЗУЕМЫХ СИМВОЛОВ ОПРЕДЕЛЯЕТСЯ КОНСТРУКЦИЕЙ ПРИМЕНЯЕМЫХ ВНЕШНИХ УСТРОЙСТВ И ОПИСАН В ПРИЛОЖЕНИИ.

2.2. ФОРМЫ ПРЕДСТАВЛЕНИЯ ДАННЫХ В ОЗУ.

В ЯЗЫКЕ ИМЕЕТСЯ ВОЗМОЖНОСТЬ РАБОТЫ С ЧИСЛОВОЙ И ТЕКСТОВОЙ ИНФОРМАЦИЕЙ.

В ЧИСЛОВЫХ ПЕРЕМЕННЫХ ИСПОЛЬЗУЮТСЯ ЧИСЛА ВОСЬМИ- И ДВУХБАЙТОВОГО ФОРМАТА.

ЧИСЛА ВОСЬМИБАЙТОВОГО ФОРМАТА ПРЕДСТАВЛЯЮТСЯ В ОЗУ В ДЕСЯТИЧНО-КОДИРОВАННОЙ СИСТЕМЕ. ИХ ДИАПАЗОН: МАНТИССА МОЖЕТ СОСТОЯТЬ ИЗ 12-ТИ, ПОРЯДОК ИЗ ДВУХ ДЕСЯТИЧНЫХ ЦИФР. И ПОРЯДОК И МАНТИССА МОГУТ БЫТЬ ОТРИЦАТЕЛЬНЫМИ. ДАЛЕЕ ЧИСЛА ВОСЬМИБАЙТОВОГО ФОРМАТА ИМЕНУЮТСЯ "ВЕЩЕСТВЕННЫМИ" (ИЛИ РЕАЛЬНЫМИ).

ЧИСЛА ДВУХБАЙТОВОГО ФОРМАТА ПРЕДСТАВЛЯЮТСЯ В ОЗУ В ШЕСТНАДЦАТИРИЧНОЙ СИСТЕМЕ. ИХ ДИАПАЗОН: ЦЕЛЫЕ ЗНАЧЕНИЯ ОТ -32767 ДО +32767. ДАЛЕЕ В ОПИСАНИИ ЭТИ ЧИСЛА ИМЕНУЮТСЯ "ЦЕЛЫМИ".

В ИНФОРМАЦИОННЫХ СТРОКАХ ПРИМЕНЯЮТСЯ ОДНОБАЙТНЫЕ ЧИСЛА, ПРЕДСТАВЛЯЕМЫЕ В ДЕСЯТИЧНО-КОДИРОВАННОЙ СИСТЕМЕ. ИХ ДИАПАЗОН: ЦЕЛЫЕ ЗНАЧЕНИЯ ОТ 0 ДО 99. ДАЛЕЕ ЭТИ ЧИСЛА ИМЕНУЮТСЯ "ЦЕЛЫМИ ОДНОБАЙТНЫМИ".

ТЕКСТОВАЯ ИНФОРМАЦИЯ ПРЕДСТАВЛЯЕТСЯ В ОЗУ В ВИДЕ ГРУПП СИМВОЛОВ, ЗАПИСАННЫХ В КОДАХ КОИ-7. СТРОЧНЫЕ БУКВЫ ДОПОЛНЯЮТСЯ УСТАНОВЛЕННЫМ ВОСЬМЬМ БИТОМ. ТЕКСТОВАЯ ПЕРЕМЕННАЯ ИЛИ РЕГИСТР ЗАНИМАЮТ В ОЗУ ОТВЕДЕННОЕ ДЛЯ НИХ КОЛИЧЕСТВО БАЙТ - ДО 128, ТЕКСТОВЫЕ ЭЛЕМЕНТЫ ИНФОРМАЦИОННЫХ СТРОК - КОЛИЧЕСТВО БАЙТ, РАВНОЕ КОЛИЧЕСТВУ СОДЕРЖАЩИХСЯ В НИХ СИМВОЛОВ.

КРОМЕ ИНФОРМАЦИИ ПОЛЬЗОВАТЕЛЯ, ИНТЕРПРЕТАТОР РАЗМЕЩАЕТ В ОЗУ СЛУЖЕБНУЮ ИНФОРМАЦИЮ (ИДЕНТИФИКАТОРЫ, РАЗДЕЛИТЕЛЬНЫЕ ЗНАКИ И Т.Д.). ЭТА ИНФОРМАЦИЯ НЕ ДОСТУПНА ПОЛЬЗОВАТЕЛЮ, НО О ЕЕ НАЛИЧИИ НЕОБХОДИМО ЗНАТЬ ПРИ РАСЧЕТЕ ПОТРЕБНОГО ОБЪЕМА ОЗУ.

2.3. КОНСТАНТЫ.

КОНСТАНТАМИ НАЗЫВАЮТСЯ ЧИСЛА И ГРУППЫ СИМВОЛОВ, ЗАПИСАННЫЕ В ТЕКСТЕ ПРОГРАММЫ В КАЧЕСТВЕ ОПЕРАНДОВ.

ЧИСЛА В ПРОГРАММЕ МОЖНО ОБОЗНАЧАТЬ В НАТУРАЛЬНОЙ И ЕСТЕСТВЕННОЙ ФОРМЕ.

ЗАПИСЬ ЧИСЛА В НАТУРАЛЬНОЙ ФОРМЕ: <ЗНАК МАНТИССЫ> <МАНТИССА> Е <ЗНАК ПОРЯДКА> <ПОРЯДОК>

ЗАПИСЬ ЧИСЛА В ЕСТЕСТВЕННОЙ ФОРМЕ: <ЗНАК ЧИСЛА> <ЦЕЛАЯ ЧАСТЬ> . <ДРОБНАЯ ЧАСТЬ>

ПРАВИЛА ОБОЗНАЧЕНИЯ ЧИСЕЛ:

- НЕЗНАЧАЩИЕ НУЛИ МОЖНО НЕ УКАЗЫВАТЬ;
- ЗНАК <+> НЕ ОБОЗНАЧАЕТСЯ;
- ПРОБЕЛЫ В ЧИСЛЕ НЕ ДОПУСКАЮТСЯ;
- ЕСЛИ ПРИ ЗАПИСИ В НАТУРАЛЬНОЙ ФОРМЕ В МАНТИССЕ ЧИСЛА УКАЗАНА ТОЧКА, ТО ИНТЕРПРЕТАТОРОМ ОНА ИГНОРИРУЕТСЯ;
- МАНТИССА ЧИСЛА ПРИ ЗАПИСИ В НАТУРАЛЬНОЙ ФОРМЕ, А ТАКЖЕ, ЦЕЛАЯ ПЛЮС ДРОБНАЯ ЧАСТЬ ЧИСЛА В ЕСТЕСТВЕННОЙ ФОРМЕ НЕ ДОЛЖНЫ ПРЕВЫШАТЬ 12-ТИ ЦИФР, ПОРЯДОК НЕ БОЛЕЕ ДВУХ.

ПРИМЕРЫ ОБОЗНАЧЕНИЯ ЧИСЕЛ:

ЧИСЛО	!ЗАПИСЬ В ЕСТЕСТВЕННОЙ ФОРМЕ	!ЗАПИСЬ В НАТУРАЛЬНОЙ ФОРМЕ
+4,5	4.5	45E1
-123,9	-123.9	-1239E3
+0,0045	.0045	45E-2
+0,24 · 10 ²¹	-	24E21
+3,5 · 10 ¹⁴	-	35E15
+0,1 · 10 ⁹⁹	-	1E99
-0,1 · 10 ⁻⁹⁹	-	-1E-99
-117,7 · 10 ⁻²¹	-	-1177E-18

ТЕКСТОВЫЕ КОНСТАНТЫ ЗАПИСЫВАЮТСЯ В ВИДЕ ЦЕПОЧКИ ЗНАКОВ, ЗАКЛЮЧЕННОЙ В АПОСТРОФЫ, ЛИБО В КАВЫЧКИ. ЕСЛИ КОНСТАНТА ЗАКЛЮЧЕНА В КАВЫЧКИ, ТО В НЕЙ МОЖНО ИСПОЛЬЗОВАТЬ АПОСТРОФ И НАОБОРОТ.

2.4. ПЕРЕМЕННЫЕ.

ПЕРЕМЕННЫЕ ОБОЗНАЧАЮТСЯ СВОИМ ИМЕНЕМ (ИЛИ ИДЕНТИФИКАТОРОМ), СОСТОЯЩИМ ИЗ ОДНОЙ ЛАТИНСКОЙ БУКВЫ ИЛИ ОДНОЙ БУКВЫ И ОДНОЙ ЦИФРЫ. СНАЧАЛА ПИШЕТСЯ БУКВА, ПОТОМ ЦИФРА.

ПЕРЕМЕННЫЕ МОГУТ БЫТЬ ТРЕХ ВИДОВ:

- ЧИСЛОВЫЕ ВЕЩЕСТВЕННОГО ТИПА;
- ЧИСЛОВЫЕ ЦЕЛОГО ТИПА (ДВУХБАЙТОВОГО ФОРМАТА);
- ТЕКСТОВЫЕ.

В ЦЕЛОЧИСЛЕННЫХ ПЕРЕМЕННЫХ ПОСЛЕ ИМЕНИ ПЕРЕМЕННОЙ СТАВИТСЯ ЗНАК <%>. ПРИ ИХ ОПРЕДЕЛЕНИИ В ПРОЦЕССЕ РАБОТЫ ПРОГРАММЫ ДРОБНАЯ ЧАСТЬ ПРИ ЕЕ НАЛИЧИИ ИГНОРИРУЕТСЯ. СЛЕДУЕТ ПОМНИТЬ, ЧТО ПРИМЕНЕНИЕ ЦЕЛОЧИСЛЕННЫХ ПЕРЕМЕННЫХ НЕ УВЕЛИЧИВАЕТ СКОРОСТИ ВЫЧИСЛЕНИЙ, А ЛИШЬ ПОЗВОЛЯЕТ ЭКОНОМИТЬ ОЗУ.

В ТЕКСТОВЫХ ПЕРЕМЕННЫХ ПОСЛЕ ИМЕНИ ПЕРЕМЕННОЙ СТАВИТСЯ ЗНАК <\$>, НАПРИМЕР, F5\$. В ТЕКСТОВЫХ ПЕРЕМЕННЫХ МОГУТ СОДЕРЖАТЬСЯ ГРУППЫ ДО 128 СИМВОЛОВ.

ПОД ОДНИМ ИМЕНЕМ МОЖНО ОРГАНИЗОВЫВАТЬ ОДНО- ИЛИ ДВУМЕРНЫЕ МАССИВЫ ПЕРЕМЕННЫХ. В ЭТОМ СЛУЧАЕ К ИМЕНИ ПЕРЕМЕННОЙ ПРИ ЕЕ ОБОЗНАЧЕНИИ ДОБАВЛЯЕТСЯ, СООТВЕТСТВЕННО, ОДНО- ИЛИ ДВУМЕРНЫЙ ИНДЕКС. МАССИВЫ ПЕРЕМЕННЫХ ОРГАНИЗУЮТСЯ ОПЕРАТОРОМ DIM.

ОБРАЩЕНИЕ К ПЕРЕМЕННОЙ НЕ ДОПУСКАЕТСЯ, ЕСЛИ ЕЕ ЗНАЧЕНИЕ НЕ БЫЛО РАНЕЕ ОПРЕДЕЛЕНО. ПРИ ОБРАЩЕНИИ К ПЕРЕМЕННОЙ С ИНДЕКСОМ, ЗАДАННЫЙ РАЗМЕР ИНДЕКСА НЕ ДОЛЖЕН ПРЕВЫШАТЬ МАКСИМАЛЬНОГО, ЗАДАННОГО ОПЕРАТОРОМ DIM.

2.5. ФАЙЛЫ ИНФОРМАЦИОННЫХ СТРОК.

ДЛЯ ОРГАНИЗАЦИИ СИСТЕМНОГО ПРЕДСТАВЛЕНИЯ ДАННЫХ В ЭВМ ПРЕДНАЗНАЧЕНЫ ФАЙЛЫ ИНФОРМАЦИОННЫХ СТРОК (ИЛИ ЗАПИСЕЙ).

ИНФОРМАЦИОННЫЕ СТРОКИ СОДЕРЖАТ В СЕБЕ КАКУЮ-ЛИБО ИНФОРМАЦИЮ, СГРУППИРОВАННУЮ ПО ОПРЕДЕЛЕННОМУ ПРИЗНАКУ. КАЖДАЯ ИНФОРМАЦИОННАЯ СТРОКА СОСТОИТ ИЗ ОТДЕЛЬНЫХ ЭЛЕМЕНТОВ, СОДЕРЖАЩИХ В СЕБЕ ДАННЫЕ РАЗЛИЧНОГО ТИПА (ЧИСЛОВЫЕ И ТЕКСТОВЫЕ).

СОВОКУПНОСТЬ ИНФОРМАЦИОННЫХ СТРОК СОСТАВЛЯЕТ ФАЙЛ (ИЛИ МАССИВ) ИНФОРМАЦИОННЫХ СТРОК. СТРОКИ В ФАЙЛЕ РАСПОЛАГАЮТСЯ ПО ПОРЯДКУ, В СООТВЕТСТВИИ СО СВОИМИ НОМЕРАМИ. ОБРАЩЕНИЕ К СТРОКАМ ДЛЯ ВВОДА ИЛИ ВЫВОДА ИНФОРМАЦИИ ПРОИЗВОДИТСЯ В СООТВЕТСТВИИ С УСТАНОВЛИВАЕМЫМ УКАЗАТЕЛЕМ ФАЙЛА.

ФАЙЛЫ И ЭЛЕМЕНТЫ ИНФОРМАЦИОННЫХ СТРОК ОБОЗНАЧАЮТСЯ СЛУЖЕБНЫМИ СЛОВАМИ, СОСТОЯЩИМИ ИЗ ДВУХ БУКВ: 1-Я БУКВА ОБОЗНАЧАЕТ ТИП ЭЛЕМЕНТА, ВТОРАЯ БУКВА L - ОЗНАЧАЕТ ПРИНАДЛЕЖНОСТЬ ЭЛЕМЕНТА К ФАЙЛУ ИНФОРМАЦИОННЫХ СТРОК (LINE - СТРОКА).

ФАЙЛЫ ОБОЗНАЧАЮТСЯ СЛОВОМ FL (FILE - ФАЙЛ), НОМЕР СТРОКИ СЛОВОМ NL (NUMBER - НОМЕР).

В ОЗУ ОДНОВРЕМЕННО МОЖНО ОРГАНИЗОВАТЬ ДО 10-ТИ ФАЙЛОВ ИНФОРМАЦИОННЫХ СТРОК. В ЭТОМ СЛУЧАЕ К ОБОЗНАЧЕНИЮ ФАЙЛА ИЛИ ЭЛЕМЕНТА СТРОКИ ДОБАВЛЯЕТСЯ ОДНА ЦИФРА - НОМЕР ФАЙЛА ОТ 0 ДО 9.

ДЛЯ ОРГАНИЗАЦИИ ФАЙЛА НЕОБХОДИМО В ОЗУ ВЫДЕЛИТЬ ОБЛАСТЬ ДЛЯ ЕГО РАЗМЕЩЕНИЯ (СМ. "ОПЕРАТОРЫ DIM И COM").

КАЖДОМУ ОРГАНИЗОВАННОМУ ФАЙЛУ СООТВЕТСТВУЕТ СВОЙ РЕГИСТР - УКАЗАТЕЛЬ ФАЙЛА, В КОТОРЫЙ ИНТЕРПРЕТАТОРОМ ЗАНОСИТСЯ АДРЕС КАКОЙ-ЛИБО СТРОКИ, НАЧИНАЯ ОТ НАЧАЛА И ДО КОНЦА ФАЙЛА. ЕСЛИ УКАЗАТЕЛЬ УСТАНОВЛЕН НА КОНЕЦ ФАЙЛА, ТО ОБРАЩЕНИЕ К ФАЙЛУ НЕ ДОПУСКАЕТСЯ. СПОСОБЫ ИЗМЕНЕНИЯ УКАЗАТЕЛЕЙ ФАЙЛОВ ОПИСАНЫ НИЖЕ (СМ. "ПРИСВОЕНИЕ ЗНАЧЕНИЙ").

2.6. ЭЛЕМЕНТЫ ИНФОРМАЦИОННЫХ СТРОК.

В КАЖДОЙ ИНФОРМАЦИОННОЙ СТРОКЕ МОЖЕТ БЫТЬ РАЗМЕЩЕНО 3 ТИПА ЭЛЕМЕНТОВ (КРОМЕ НОМЕРА СТРОКИ), ИМЕЮЩИХ СЛЕДУЮЩИЕ ОБОЗНАЧЕНИЯ:

IL - ЦЕЛОЕ ЧИСЛО В ДИАПАЗОНЕ ОТ 0 ДО 99 (INTEGER - ЦЕЛОЕ);

RL - ВЕЩЕСТВЕННОЕ ЧИСЛО (REAL);

TL - ГРУППА СИМВОЛОВ (TEXT - ТЕКСТ).

КРОМЕ ТИПА ЭЛЕМЕНТА В ЕГО ОБОЗНАЧЕНИИ УКАЗЫВАЕТСЯ ЦИФРА - НОМЕР

ФАЙЛА И ОДНОЭЛЕМЕНТНЫЙ ИНДЕКС, ОЗНАЧАЮЩИЙ ПОРЯДКОВЫЙ НОМЕР ЭЛЕМЕНТА ДАННОГО ТИПА В СТРОКЕ. ИНДЕКС МОЖЕТ ИМЕТЬ ЗНАЧЕНИЯ ОТ 0 ДО 255, Т.Е. В КАЖДОЙ СТРОКЕ МОЖЕТ СОДЕРЖАТЬСЯ ДО 255 ЭЛЕМЕНТОВ КАЖДОГО ТИПА. ДЛИНА ИНФОРМАЦИОННОЙ СТРОКИ ОГРАНИЧЕНА ЛИШЬ РАЗМЕРОМ ОБЛАСТИ ОЗУ, ВЫДЕЛЕННОЙ ДЛЯ ФАЙЛА.

ЭЛЕМЕНТЫ TL ЗАНИМАЮТ В ОЗУ КОЛИЧЕСТВО БАЙТ, РАВНОЕ КОЛИЧЕСТВУ СОДЕРЖАЩИХСЯ В НИХ СИМВОЛОВ, ЭЛЕМЕНТЫ ТИПА IL ПО ОДНОМУ, А ТИПА RL ПО 8 БАЙТ. НОМЕР СТРОКИ ЗАНИМАЕТ 2 БАЙТА. ЭЛЕМЕНТЫ TL, НЕ СОДЕРЖАЩИЕ СИМВОЛОВ, И ЭЛЕМЕНТЫ IL И RL, РАВНЫЕ НУЛЮ ИНТЕРПРЕТАТОРОМ В ИНФОРМАЦИОННЫЕ СТРОКИ НЕ ЗАНОСЯТСЯ.

КРОМЕ ИНФОРМАЦИИ ПОЛЬЗОВАТЕЛЯ ИНТЕРПРЕТАТОР РАЗМЕЩАЕТ В СТРОКАХ СЛУЖЕБНУЮ ИНФОРМАЦИЮ (РАЗДЕЛИТЕЛЬНЫЕ ЗНАКИ), НЕ ДОСТУПНУЮ ПОЛЬЗОВАТЕЛЮ. О НАЛИЧИИ ЭТОЙ ИНФОРМАЦИИ НЕОБХОДИМО ЗНАТЬ ПРИ РАСЧЕТЕ ПОТРЕБНОЙ ОБЛАСТИ ОЗУ. ОБЪЕМ ЭТОЙ ИНФОРМАЦИИ ОПРЕДЕЛЯЕТСЯ ОПЫТНЫМ ПУТЕМ.

ПРИМЕРЫ ОБОЗНАЧЕНИЯ ЭЛЕМЕНТОВ ИНФОРМАЦИОННЫХ СТРОК:

RL(5) - ЧИСЛО ВЕЩЕСТВЕННОГО ТИПА С ПОРЯДКОВЫМ НОМЕРОМ 5 В ФАЙЛЕ FL0;

IL7(J) - ЦЕЛОЕ ЧИСЛО С ПОРЯДКОВЫМ НОМЕРОМ J В ФАЙЛЕ FL7;

TL9(I+K) - ТЕКСТОВЫЙ ЭЛЕМЕНТ СТРОКИ С ПОРЯДКОВЫМ НОМЕРОМ I+J В ФАЙЛЕ FL9.

2.7. РЕГИСТРЫ.

ЧАСТЬ ОПЕРАТИВНОЙ ИНФОРМАЦИИ МОЖНО РАСПОЛАГАТЬ В СПЕЦИАЛЬНЫХ РЕГИСТРАХ, ОРГАНИЗУЕМЫХ ИНТЕРПРЕТАТОРОМ В СЛУЖЕБНОЙ ЗОНЕ ОЗУ.

РЕГИСТРЫ ПРЕДНАЗНАЧЕНЫ ДЛЯ ОБМЕНА ПАРАМЕТРАМИ С ВНЕШНИМИ ПОДПРОГРАММАМИ И НЕКОТОРЫМИ ОПЕРАТОРАМИ. РЕГИСТРЫ МОЖНО ИСПОЛЬЗОВАТЬ, ТАКЖЕ, ПО УСМОТРЕНИЮ ПОЛЬЗОВАТЕЛЯ В КАЧЕСТВЕ ЧИСЛОВЫХ И ТЕКСТОВЫХ ПЕРЕМЕННЫХ.

ИНТЕРПРЕТАТОР ОРГАНИЗУЕТ ДВА ТИПА РЕГИСТРОВ:

- ДЛЯ ЗАПИСИ ЧИСЛОВЫХ ПЕРЕМЕННЫХ ВЕЩЕСТВЕННОГО ТИПА;
- ДЛЯ ЗАПИСИ ТЕКСТОВЫХ ПЕРЕМЕННЫХ.

ЧИСЛОВЫЕ РЕГИСТРЫ ОБОЗНАЧАЮТСЯ СЛОВОМ RR И ОДНОЙ ЦИФРОЙ - НОМЕРОМ РЕГИСТРА. ВСЕГО ИМЕЕТСЯ 4 РЕГИСТРА RR, Т.Е. RR0..RR3.

ТЕКСТОВЫЕ РЕГИСТРЫ ОБОЗНАЧАЮТСЯ СЛОВОМ TR И ОДНОЙ ЦИФРОЙ. В КАЖДОМ РЕГИСТРЕ МОЖНО РАЗМЕСТИТЬ ГРУППУ ДО 128 СИМВОЛОВ. ВСЕГО ИМЕЕТСЯ 2 РЕГИСТРА TR - TR0 И TR1.

В ИМЕНАХ РЕГИСТРОВ ЦИФРУ 0 МОЖНО НЕ ОБОЗНАЧАТЬ, Т.Е. ЗАПИСЬ TR0 АНАЛОГИЧНА TR, RR0 АНАЛОГИЧНА RR.

2.8. СЛОВА.

ЛЮБОЕ ОБОЗНАЧЕНИЕ, ПРИНЯТОЕ В ЯЗЫКЕ, СОСТОЯЩЕЕ ИЗ ДВУХ ИЛИ БОЛЕЕ ЛАТИНСКИХ БУКВ НАЗЫВАЕТСЯ СЛОВОМ (КРОМЕ ВЫРАЖЕНИЙ В АПОСТРОФАХ). К СЛОВАМ ОТНОСЯТСЯ:

- ИМЕНА ОПЕРАТОРОВ;
- ИМЕНА СЛУЖЕБНЫХ СЛОВ, ОПРЕДЕЛЯЮЩИХ ПАРАМЕТРЫ ОПЕРАТОРОВ;

- ИМЕНА ФУНКЦИЙ;
- ИМЕНА ФАЙЛОВ И ИХ ЭЛЕМЕНТОВ;
- ИМЕНА РЕГИСТРОВ;
- СЛОВО PI ДЛЯ ОБОЗНАЧЕНИЯ КОНСТАНТЫ "ПИ".

ПРИ ОБОЗНАЧЕНИИ СЛОВ СУЩЕСТВУЮТ СЛЕДУЮЩИЕ ПРАВИЛА:

- ПРОБЕЛЫ ВНУТРИ СЛОВ НЕ ДОПУСКАЮТСЯ;
- ПРОБЕЛЫ МЕЖДУ ОТДЕЛЬНЫМИ СЛОВАМИ, А ТАКЖЕ МЕЖДУ СЛОВАМИ И СТОЯЩИМИ ЗА НИМИ ПЕРЕМЕННЫМИ МОЖНО НЕ УКАЗЫВАТЬ;
- СЛОВО ОБЯЗАТЕЛЬНО ДОЛЖНО БЫТЬ ОТДЕЛЕНО ПРОБЕЛОМ ОТ ВПЕРЕДИ СТОЯЩЕЙ ПЕРЕМЕННОЙ, ОБОЗНАЧЕННОЙ ОДНОЙ БУКВОЙ;
- ПРОБЕЛ НЕ ДОПУСКАЕТСЯ МЕЖДУ ИМЕНЕМ ФУНКЦИИ И ОТКРЫВАЮЩЕЙ СКОБКОЙ ЕЕ АРГУМЕНТА.

АНГЛИЙСКИЕ ТЕРМИНЫ, ПРИМЕНЯЕМЫЕ В СЛОВАХ, ИМЕЮТ СЛЕДУЮЩИЙ ПЕРЕВОД:

AND	- И
CALL	- ВЫЗВАТЬ
CHANGE	- ПРЕОБРАЗОВАТЬ
COM	- COMMON - ОБЩИЙ
COMPILE	- ТРАНСЛИРОВАТЬ
CLEAR	- ОЧИСТИТЬ
DATA	- ДАННЫЕ
DIM	- DIMENSION - РАЗМЕР
ELSE	- ИНАЧЕ
END	- КОНЕЦ
FOR	- ДЛЯ
GOSUB	- ПОДПРОГРАММА
GOTO	- GO TO - ИДТИ К
IF	- ЕСЛИ
INPUT	- ВВЕСТИ
LET	- ПУСТЬ
LIST	- СПИСОК, ПЕРЕЧЕНЬ
LOAD	- ЗАГРУЗИТЬ
NEXT	- СЛЕДУЮЩИЙ
ON	- НА
OR	- ИЛИ
PRINT	- НАПИСАТЬ
READ	- ПРОЧЕСТЬ
REM	- REMARK - КОММЕНТАРИЙ
REN	- RENAME - ПЕРЕИМЕНОВАТЬ
RESTORE	- ВОССТАНОВИТЬ
RETURN	- ВОЗВРАТИТЬСЯ
REW	- REWIND - ПЕРЕМОТАТЬ НАЗАД
RUN	- РАБОТАТЬ
SAVE	- ХРАНИТЬ
SCREEN	- ЭКРАН
SET	- УСТАНОВИТЬ
SIZE	- РАЗМЕР
SKIP	- ПЕРЕСКОЧИТЬ
START	- ПУСК
STEP	- ШАГ
STOP	- СТОП
THEN	- ТОГДА
USING	- ИСПОЛЬЗУЯ

2.9. ИНДЕКСНЫЕ ВЫРАЖЕНИЯ И ИНДЕКСЫ.

ДЛЯ ОБОЗНАЧЕНИЯ ИНДЕКСОВ, А ТАКЖЕ, ДЛЯ ОБОЗНАЧЕНИЯ НЕКОТОРЫХ ЦЕЛОЧИСЛЕННЫХ ПАРАМЕТРОВ В ДАННОЙ ВЕРСИИ ЯЗЫКА ИСПОЛЗУЮТСЯ ИНДЕКСНЫЕ ВЫРАЖЕНИЯ, ПРЕДСТАВЛЯЮЩИЕ СОБОЙ УПРОЩЕННЫЕ АРИФМЕТИЧЕСКИЕ ВЫРАЖЕНИЯ. ИНДЕКСНЫЕ ВЫРАЖЕНИЯ МОГУТ СОСТОЯТЬ НЕ БОЛЕЕ, ЧЕМ ИЗ ДВУХ ПЕРЕМЕННЫХ, ИЛИ КОНСТАНТ, РАЗДЕЛЕННЫХ МЕЖДУ СОБОЙ ЗНАКАМИ <+>, <-> ИЛИ <*>.

НЕ ДОПУСКАЕТСЯ ИСПОЛЬЗОВАНИЕ В ИНДЕКСНЫХ ВЫРАЖЕНИЯХ ПЕРЕМЕННЫХ С ИНДЕКСОМ, Т.Е. ВЛОЖЕНИЕ ИНДЕКСОВ ДРУГ В ДРУГА. ВЕЛИЧИНА, КАК ОТДЕЛЬНЫХ ПЕРЕМЕННЫХ ИЛИ КОНСТАНТ, СОСТАВЛЯЮЩИХ ИНДЕКСНОЕ ВЫРАЖЕНИЕ, ТАК И ВСЕГО ИНДЕКСНОГО ВЫРАЖЕНИЯ ДОЛЖНА НАХОДИТЬСЯ В ПРЕДЕЛАХ ОТ 0 ДО 255. ДРОБНАЯ ЧАСТЬ, ПРИ ЕЕ НАЛИЧИИ, ИНТЕРПРЕТАТОРОМ ОТБРАСЫВАЕТСЯ.

ИНДЕКСЫ МОГУТ БЫТЬ ОДНОМЕРНЫМИ И ДВУМЕРНЫМИ. ОДНОМЕРНЫЕ ИНДЕКСЫ ОБОЗНАЧАЮТСЯ ОДНИМ ИНДЕКСНЫМ ВЫРАЖЕНИЕМ, ДВУМЕРНЫЕ - ДВУМЯ, РАЗДЕЛЕННЫМИ ЗАПЯТОЙ. ИНДЕКС ЗАКЛЮЧАЕТСЯ В КРУГЛЫЕ СКОБКИ. ЕСЛИ ПОСЛЕДНЕЕ ИНДЕКСНОЕ ВЫРАЖЕНИЕ В ДВУМЕРНОМ ИНДЕКСЕ РАВНО НУЛЮ, ТО ЕГО МОЖНО НЕ УКАЗЫВАТЬ, ЕСЛИ ВСЕ ВЫРАЖЕНИЯ В ИНДЕКСЕ РАВНЫ НУЛЮ, ТО ИНДЕКС МОЖНО НЕ ОБОЗНАЧАТЬ.

2.10. ФУНКЦИИ.

В АРИФМЕТИЧЕСКИХ И ТЕКСТОВЫХ ВЫРАЖЕНИЯХ ПРИМЕНЯЮТСЯ СТАНДАРТНЫЕ И СПЕЦИАЛЬНЫЕ ФУНКЦИИ.

СТАНДАРТНЫЕ ФУНКЦИИ ПРЕДСТАВЛЕНЫ В СЛЕДУЮЩЕЙ ТАБЛИЦЕ:

ЗАПИСЬ НА ЯЗЫКЕ !	ЗНАЧЕНИЕ ФУНКЦИИ	ПРИМЕЧАНИЕ
ABS		
ATN		ЗНАЧЕНИЕ В РАДИАНАХ
COS		АРГУМЕНТ В РАДИАНАХ
DEG		ПЕРЕВОД РАДИАН В ГРАДУСЫ
EXP		
EXT		
INT		ОТБРАСЫВАНИЕ ДРОБН. ЧАСТИ
LGT		
LOG		
RAD		ПЕРЕВОД ГРАД. В РАДИАНЫ
SIN		АРГУМЕНТ В РАДИАНАХ
SQR		
TAN		АРГУМЕНТ В РАДИАНАХ

СПЕЦИАЛЬНЫЕ ФУНКЦИИ ПОДРАЗДЕЛЯЮТСЯ НА ЧИСЛОВЫЕ И ТЕКСТОВЫЕ. В АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЯХ ДОПУСКАЕТСЯ ПРИМЕНЯТЬ ТОЛЬКО ЧИСЛОВЫЕ, В ТЕКСТОВЫХ ТОЛЬКО ТЕКСТОВЫЕ СПЕЦФУНКЦИИ. АРГУМЕНТ СПЕЦФУНКЦИИ, КАК ПРАВИЛО, МОЖЕТ БЫТЬ ОБОЗНАЧЕН ОДНОЙ ПЕРЕМЕННОЙ ИЛИ ДРУГОЙ СПЕЦФУНКЦИЕЙ. ПРИМЕНЕНИЕ АРИФМЕТИЧЕСКИХ И ТЕКСТОВЫХ ВЫРАЖЕНИЙ В АРГУМЕНТАХ СПЕЦФУНКЦИЙ НЕ ДОПУСКАЕТСЯ.

К ЧИСЛОВЫМ СПЕЦФУНКЦИЯМ ОТНОСЯТСЯ:

А). ФУНКЦИЯ VAL.

СЛУЖИТ ДЛЯ ПЕРЕВОДА ЧИСЕЛ, ЗАПИСАННЫХ В СИМВОЛЬНОМ ВИДЕ, В ВЕЩЕСТВЕННУЮ ФОРМУ. АРГУМЕНТ МОЖЕТ БЫТЬ ОБОЗНАЧЕН ТЕКСТОВОЙ ПЕРЕМЕННОЙ ЛИБО СПЕЦФУНКЦИЕЙ SEG.

Б). ФУНКЦИЯ LEN.

ОПРЕДЕЛЯЕТ ДЛИНУ ТЕКСТОВОГО ЭЛЕМЕНТА. АРГУМЕНТ МОЖЕТ БЫТЬ ОБОЗНАЧЕН ТЕКСТОВОЙ ПЕРЕМЕННОЙ ИЛИ КОНСТАНТОЙ, ЛИБО СПЕЦФУНКЦИЕЙ STR.

В). ФУНКЦИЯ SIZE.

ОПРЕДЕЛЯЕТ ВЕЛИЧИНУ СВОБОДНОЙ ОБЛАСТИ ПОЛЬЗОВАТЕЛЯ. ЕСЛИ АРГУМЕНТ ФУНКЦИИ ОБОЗНАЧЕН ИМЕНЕМ ФАЙЛА FL, ТО ФУНКЦИЯ РАВНА КОЛИЧЕСТВУ СВОБОДНЫХ БАЙТ В ДАННОМ ФАЙЛЕ. ЕСЛИ АРГУМЕНТ НЕ ОБОЗНАЧЕН, ТО ФУНКЦИЯ РАВНА ВЕЛИЧИНЕ СВОБОДНОЙ ОБЛАСТИ ПОЛЬЗОВАТЕЛЯ.

Г). ФУНКЦИЯ LNT.

ОПРЕДЕЛЯЕТ СОСТОЯНИЕ СЧЕТЧИКА СТРОК ПРИНТЕРА. АРГУМЕНТ ОТСУТСТВУЕТ.

К ТЕКСТОВЫМ СПЕЦФУНКЦИЯМ ОТНОСЯТСЯ:

А). ФУНКЦИЯ STR.

ФУНКЦИЯ ОБРАТНАЯ К VAL. ПЕРЕВОДИТ ЧИСЛО ИЗ ВЕЩЕСТВЕННОЙ ФОРМЫ В СИМВОЛЬНУЮ. АРГУМЕНТ ФУНКЦИИ ОБОЗНАЧАЕТСЯ ЧИСЛОВОЙ ПЕРЕМЕННОЙ. ЕСЛИ МОДУЛЬ АРГУМЕНТА ИМЕЕТ ПОРЯДОК ВЫШЕ 9, ТО ФУНКЦИЯ РАВНА СИМВОЛУ <*>.

Б). ФУНКЦИЯ SEG.

ВЫДЕЛЯЕТ СЕГМЕНТ ИЗ ТЕКСТОВОЙ ПЕРЕМЕННОЙ, НАЧИНАЯ С СИМВОЛА С НОМЕРОМ N1 ПО СИМВОЛ С НОМЕРОМ N2. ФУНКЦИЯ ЗАПИСЫВАЕТСЯ В СЛЕДУЮЩЕМ ВИДЕ:

SEG (<ТЕКСТОВАЯ ПЕРЕМЕННАЯ>, <N1>-<N2>)

СИМВОЛЫ НУМЕРУЮТСЯ ПО ПОРЯДКУ, НАЧИНАЯ С ЕДИНИЦЫ. ЕСЛИ В АРГУМЕНТЕ УКАЗАН ТОЛЬКО N1, ТО ВЫДЕЛЯЕТСЯ ОДИН СИМВОЛ С УКАЗАННЫМ НОМЕРОМ, ЕСЛИ УКАЗАН N1 И ТИРЕ, ТО ВЫДЕЛЯЕТСЯ СЕГМЕНТ С N1 ДО КОНЦА ПЕРЕМЕННОЙ.

АРГУМЕНТЫ ФУНКЦИЙ ОБЯЗАТЕЛЬНО ДОЛЖНЫ БЫТЬ ЗАКЛЮЧЕНЫ В КРУГЛЫЕ СКОБКИ. ПРОБЕЛ МЕЖДУ ИМЕНЕМ ФУНКЦИИ И ОТКРЫВАЮЩЕЙ СКОБКОЙ НЕ ДОПУСКАЕТСЯ. ЕСЛИ СПЕЦФУНКЦИЯ НЕ ИМЕЕТ АРГУМЕНТА, ТО СКОБКИ НЕ ОБОЗНАЧАЮТСЯ.

2.11. АРИФМЕТИЧЕСКИЕ ВЫРАЖЕНИЯ.

АРИФМЕТИЧЕСКИЕ ВЫРАЖЕНИЯ ПРЕДСТАВЛЯЮТ СОБОЙ НАБОР ЧИСЛОВЫХ ЭЛЕМЕНТОВ ЯЗЫКА, РАЗДЕЛЕННЫХ МЕЖДУ СОБОЙ АРИФМЕТИЧЕСКИМИ ЗНАКАМИ. В АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЯХ ДОПУСКАЮТСЯ СЛЕДУЮЩИЕ ЧИСЛОВЫЕ ЭЛЕМЕНТЫ:

- ЧИСЛОВЫЕ КОНСТАНТЫ;
- ЧИСЛОВЫЕ ПЕРЕМЕННЫЕ БЕЗ ИНДЕКСА И С ИНДЕКСОМ;
- РЕГИСТРЫ RR;
- ЭЛЕМЕНТЫ ИНФОРМАЦИОННЫХ СТРОК NL, IL, RL;
- СТАНДАРТНЫЕ И ЧИСЛОВЫЕ СПЕЦИАЛЬНЫЕ ФУНКЦИИ;

ПЕРЕД КАЖДЫМ ИЗ ЭТИХ ЭЛЕМЕНТОВ МОЖЕТ СТОЯТЬ ЗНАК МИНУС.

КОНСТАНТУ "ПИ" МОЖНО ОБОЗНАЧИТЬ СЛОВОМ <PI>.

АРИФМЕТИЧЕСКИЕ ЗНАКИ ИМЕЮТ СЛЕДУЮЩИЕ ОБОЗНАЧЕНИЯ:

- <+> - ПЛЮС;
- <-> - МИНУС;
- <*> - УМНОЖЕНИЕ;
- </> - ДЕЛЕНИЕ;
- <^> - ВОЗВЕДЕНИЕ В СТЕПЕНЬ.

ПРИОРИТЕТ АРИФМЕТИЧЕСКИХ ДЕЙСТВИЙ В ПОРЯДКЕ УБЫВАНИЯ СЛЕДУЮЩИЙ:

- ВОЗВЕДЕНИЕ В СТЕПЕНЬ;
- УМНОЖЕНИЕ И ДЕЛЕНИЕ;
- СЛОЖЕНИЕ И ВЫЧИТАНИЕ.

ДЕЙСТВИЯ, ПРИНАДЛЕЖАЩИЕ ОДНОМУ РАНГУ, Т.Е. С ОДИНАКОВЫМ ПРИОРИТЕТОМ, ВЫПОЛНЯЮТСЯ: ВОЗВЕДЕНИЕ В СТЕПЕНЬ СПРАВА НАЛЕВО, ОСТАЛЬНЫЕ СЛЕВА НАПРАВО, НАПРИМЕР:

$A/B * C$ АНАЛОГИЧНО $(A/B) * C$

$A \wedge B \wedge C$ АНАЛОГИЧНО $A \wedge (B \wedge C)$

ДЛЯ ИЗМЕНЕНИЯ ПОРЯДКА ВЫПОЛНЕНИЯ ДЕЙСТВИЙ ИСПОЛЬЗУЮТСЯ КРУГЛЫЕ СКОБКИ. ПЕРЕД ОТКРЫВАЮЩИМИ СКОБКАМИ (ЗА ИСКЛЮЧЕНИЕМ СКОБОК АРГУМЕНТОВ ФУНКЦИЙ) МОЖЕТ СТОЯТЬ ЗНАК МИНУС.

ПРИ ПРОГРАММИРОВАНИИ АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ НЕОБХОДИМО ПРЕДСТАВЛЯТЬ ВОЗМОЖНОСТИ ЭВМ В ОПЕРАЦИЯХ НАД ВЕЩЕСТВЕННЫМИ ЧИСЛАМИ С ЦЕЛЬЮ НЕДОПУЩЕНИЯ НЕКОРРЕКТНЫХ СИТУАЦИЙ В ПРОЦЕССЕ ВЫЧИСЛЕНИЙ, ТАКИХ КАК, ПРЕВЫШЕНИЕ ДИАПАЗОНА ЧИСЕЛ, НЕДОПУСТИМЫЙ АРГУМЕНТ ФУНКЦИИ, ДЕЛЕНИЕ НА НОЛЬ И Т.Д.

ИНТЕРПРЕТАТОР ВЫПОЛНЯЕТ ВОЗВЕДЕНИЕ В СТЕПЕНЬ СЛЕДУЮЩИМ ОБРАЗОМ: ЕСЛИ ПОКАЗАТЕЛЬ СТЕПЕНИ ЦЕЛОЕ ЧИСЛО В ДИАПАЗОНЕ ОТ 0 ДО 40, ТО ВЫЧИСЛЕНИЯ ПРОИЗВОДЯТСЯ ПУТЕМ УМНОЖЕНИЯ, ПРИ ЭТОМ ЧИСЛО, ВОЗВОДИМОЕ В СТЕПЕНЬ, МОЖЕТ БЫТЬ РАВНЫМ ИЛИ МЕНЬШЕ НУЛЯ; В ОСТАЛЬНЫХ СЛУЧАЯХ ВЫЧИСЛЕНИЯ ПРОИЗВОДЯТСЯ С ПОМОЩЬЮ ЛОГАРИФМОВ И ЧИСЛО, ВОЗВОДИМОЕ В СТЕПЕНЬ, ДОЛЖНО БЫТЬ БОЛЬШЕ НУЛЯ.

ДЛЯ ОКРУГЛЕНИЯ РЕЗУЛЬТАТА ВЫЧИСЛЕНИЯ АРИФМЕТИЧЕСКОГО ВЫРАЖЕНИЯ, В КОНЦЕ ВЫРАЖЕНИЯ МОЖНО ПОСТАВИТЬ ЦИФРУ В КВАДРАТНЫХ СКОБКАХ, ОЗНАЧАЮЩУЮ ЧИСЛО ЗНАКОВ ПОСЛЕ ТОЧКИ НА КОТОРОЕ НЕОБХОДИМО ОКРУГЛИТЬ РЕЗУЛЬТАТ. ЕСЛИ НЕОБХОДИМО ОТБРОСИТЬ ЛИШНИЕ ЦИФРЫ БЕЗ ОКРУГЛЕНИЯ, ТО ПЕРЕД ЦИФРОЙ В КВАДРАТНЫХ СКОБКАХ, СТАВИТСЯ БУКВА Е.

ПРИМЕРЫ АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ:

$A * B + C$

$(A2 + X9\%) / Y$ [2]

$N1 (X1\% * I\%, J\% - 7) * X1 / (2 - A5)$ [E6]

$-X + Y / - (A + B) \wedge 2 + 1258E - 9$

$RR8 + NL * 2 + I\%$

$A * IL5 (28 + J) - IL2 (4)$

$RL5 (2) * RL5 (3) / (A\% (I, J) - A\% (I, 0))$ [3]

$SIN (A + B) - COS (PI / 6) + TAN (DEG (20))$ [9]

$1900 + VAL (SEG (D\$, 4 - 5)) - G$

$LEN (R2\$ (I + K\%, J)) + C\%$

$N * 1024 - SIZE (FL1)$

И Т.П.

2.12. ТЕКСТОВЫЕ ВЫРАЖЕНИЯ.

ТЕКСТОВЫЕ ВЫРАЖЕНИЯ ПРЕДСТАВЛЯЮТ СОБОЙ ОДИН ТЕКСТОВЫЙ ЭЛЕМЕНТ ИЛИ НЕСКОЛЬКО, РАЗДЕЛЕННЫХ ЗНАКАМИ <+>.

ЭЛЕМЕНТ ТЕКСТОВОГО ВЫРАЖЕНИЯ МОЖЕТ БЫТЬ ОБОЗНАЧЕН ТЕКСТОВОЙ ПЕРЕМЕННОЙ, ТЕКСТОВОЙ КОНСТАНТОЙ ИЛИ ТЕКСТОВОЙ СПЕЦФУНКЦИЕЙ.

ЗНАК <+> ОЗНАЧАЕТ ПОСЛЕДОВАТЕЛЬНОЕ ПРИСОЕДИНЕНИЕ ОДНОГО ЭЛЕМЕНТА К ДРУГОМУ. ОБЩЕЕ КОЛИЧЕСТВО СИМВОЛОВ В ТЕКСТОВОМ ВЫРАЖЕНИИ НЕ ДОЛЖНО ПРЕВЫШАТЬ 128.

ПРИМЕРЫ ТЕКСТОВЫХ ВЫРАЖЕНИЙ:

TL(8)+SEG(A\$,I%-)

A\$+'ИВАНОВ'

TL7+', '+TL7(1)+', '+TL7(2)+', '+TL7(3)

TR2+F1\$(I+K,J-1)

'ПРЕДПРИЯТИЕ '+STR(J)

И Т.П.

2.13. ПРОГРАММА.

ПРОГРАММА СОСТОИТ ИЗ ПРОНУМЕРОВАННЫХ СТРОК. НОМЕР СТРОКИ ОБОЗНАЧАЕТСЯ ЦЕЛЫМ ЧИСЛОМ В ДИАПАЗОНЕ ОТ 1 ДО 32767. РЕКОМЕНДУЕТСЯ СТРОКАМ ПРИСВАИВАТЬ НОМЕРА С ИНТЕРВАЛОМ, ЧТОБЫ БЫЛА ВОЗМОЖНОСТЬ МЕЖДУ НИМИ ВСТАВИТЬ НОВЫЕ СТРОКИ.

СТРОКИ ПРОГРАММЫ СОСТОЯТ ИЗ ОПЕРАТОРОВ. МАКСИМАЛЬНАЯ ДЛИНА СТРОКИ ПРОГРАММЫ - 80 СИМВОЛОВ, ВКЛЮЧАЯ КОНЕЧНЫЙ СИМВОЛ <ПЕРЕВОД СТРОКИ>. ОПЕРАТОРЫ В СТРОКЕ ПРОГРАММЫ ОТДЕЛЯЮТСЯ ДРУГ ОТ ДРУГА ДВОЕТОЧИЯМИ, ЛИБО СИМВОЛАМИ <_>. В СЛУЧАЕ ПРИМЕНЕНИЯ ОПЕРАТОРА IF РАЗДЕЛИТЕЛЯМИ МЕЖДУ ОПЕРАТОРАМИ МОГУТ БЫТЬ СЛУЖЕБНЫЕ СЛОВА THEN И ELSE.

ЛИШНИЕ РАЗДЕЛИТЕЛЬНЫЕ ЗНАКИ, А ТАКЖЕ, ЛИШНИЕ ПРОБЕЛЫ ИНТЕРПРЕТАТОРОМ ИГНОРИРУЮТСЯ.

В НАЧАЛЕ КАЖДОГО ОПЕРАТОРА СТАВИТСЯ ЕГО ИМЯ, ЗАТЕМ СЛЕДУЕТ "ТЕЛО ОПЕРАТОРА". ИМЯ ОПЕРАТОРА LET МОЖНО НЕ ОБОЗНАЧАТЬ. МОЖНО НЕ УКАЗЫВАТЬ, ТАКЖЕ, ИМЯ ОПЕРАТОРА GOTO, А ЕСЛИ ОН СТОИТ СРАЗУ ПОСЛЕ НОМЕРА СТРОКИ, ТО ВМЕСТО НЕГО МОЖНО УКАЗАТЬ РАЗДЕЛИТЕЛЬ.

КОММЕНТАРИИ В ПРОГРАММЕ ОБОЗНАЧАЮТСЯ ПОСЛЕ СЛОВА REM. КОММЕНТАРИЙ МОЖЕТ СОДЕРЖАТЬ ЛЮБЫЕ СИМВОЛЫ, ЗАКЛЮЧАТЬ ЕГО В КАВЫЧКИ ИЛИ АПОСТРОФЫ НЕ ОБЯЗАТЕЛЬНО.

В ТЕКСТЕ ПРОГРАММЫ МОЖЕТ БЫТЬ ОРГАНИЗОВАН БЛОК ДАННЫХ. ДАННЫЕ ОБОЗНАЧАЮТСЯ ПОСЛЕ СЛОВА DATA И МОГУТ СОСТОЯТЬ ИЗ ТЕКСТОВЫХ ИЛИ ЧИСЛОВЫХ КОНСТАНТ. ОТДЕЛЬНЫЕ ЗНАЧЕНИЯ РАЗДЕЛЯЮТСЯ ДВОЕТОЧИЕМ, НЕОТРИЦАТЕЛЬНЫЕ ЧИСЛА И ТЕКСТОВЫЕ КОНСТАНТЫ В АПОСТРОФАХ ИЛИ КАВЫЧКАХ МОЖНО РАЗДЕЛЯТЬ ПРОБЕЛАМИ. ТЕКСТОВЫЕ КОНСТАНТЫ МОЖНО НЕ ЗАКЛЮЧАТЬ В АПОСТРОФЫ, ОДНАКО, В ЭТОМ СЛУЧАЕ НАЧАЛЬНЫЕ ПРОБЕЛЫ В НИХ БУДУТ ПРОПУЩЕНЫ. БЛОК ДАННЫХ МОЖНО РАСПОЛАГАТЬ В ЛЮБОМ МЕСТЕ ПРОГРАММЫ. СЧИТЫВАНИЕ ИНФОРМАЦИИ ИЗ БЛОКА ПРОИЗВОДИТСЯ ОПЕРАТОРОМ READ.

ПОСЛЕ СЛОВ DATA И REM В ДАННОЙ СТРОКЕ ПРОГРАММЫ НИКАКИЕ ОПЕРАТОРЫ НЕ ДОПУСКАЮТСЯ.

ВЫПОЛНЕНИЕ ПРОГРАММЫ ПРОИСХОДИТ ПО ПОРЯДКУ ВОЗРАСТАНИЯ НОМЕРОВ СТРОК И ЗАКАНЧИВАЕТСЯ ПОСЛЕ ВЫПОЛНЕНИЯ СТРОКИ С НАИБОЛЬШИМ НОМЕРОМ.

В ПРОГРАММЕ НЕ ДОПУСКАЕТСЯ ПРИМЕНЕНИЕ ОПЕРАТОРОВ CLEAR И LOAD, КОРРЕКТИРУЮЩИХ ТЕКСТ ПРОГРАММЫ.

3. ПРИСВАИВАНИЕ ЗНАЧЕНИЙ.

3.1. ОПЕРАТОРЫ DIM И SOM.

ОПЕРАТОР DIM ИСПОЛЬЗУЕТСЯ ДЛЯ ОРГАНИЗАЦИИ МАССИВОВ ПЕРЕМЕННЫХ. В ОПЕРАТОРЕ УКАЗЫВАЕТСЯ ИМЯ МАССИВА И МАКСИМАЛЬНОЕ ЗНАЧЕНИЕ ИНДЕКСА ПЕРЕМЕННОЙ. МАССИВ МОЖЕТ БЫТЬ ОДНО И ДВУМЕРНЫМ. ОТДЕЛЬНЫЕ ОБОЗНАЧЕНИЯ МАССИВОВ РАЗДЕЛЯЮТСЯ ЗАПЯТЫМИ.

ПРИ ОБРАЗОВАНИИ МАССИВОВ ТЕКСТОВЫХ ПЕРЕМЕННЫХ ПОСЛЕ ИНДЕКСА, ЧЕРЕЗ ЗНАК РАВЕНСТВА УКАЗЫВАЕТСЯ ЧИСЛО, ОЗНАЧАЮЩЕЕ МАКСИМАЛЬНУЮ ДЛИНУ ТЕКСТОВЫХ ПЕРЕМЕННЫХ. ДИАПАЗОН ДЛИНЫ ПЕРЕМЕННОЙ ОТ 1 ДО 128. ЕСЛИ ДЛИНА ПЕРЕМЕННОЙ НЕ ОБОЗНАЧЕНА, ТО ИНТЕРПРЕТАТОРОМ ОНА ПРИНИМАЕТСЯ РАВНОЙ 128.

ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА DIM ВСЕ ЗНАЧЕНИЯ ПЕРЕМЕННЫХ В МАССИВАХ РАВНЫ НУЛЮ (ТЕКСТОВЫЕ НЕ СОДЕРЖАТ СИМВОЛОВ).

ПЕРЕМЕННЫЕ БЕЗ ИНДЕКСОВ ПРЕДСТАВЛЯЮТСЯ В ОЗУ, КАК МАССИВЫ ПЕРЕМЕННЫХ С МАКСИМАЛЬНЫМ ИНДЕКСОМ (0,0). ПОЭТОМУ ОПЕРАТОР DIM МОЖНО ИСПОЛЬЗОВАТЬ ДЛЯ ОПРЕДЕЛЕНИЯ ПРОСТЫХ ПЕРЕМЕННЫХ, НАПРИМЕР, ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА DIM A,B,C,I%,J% В ОЗУ БУДУТ ОПРЕДЕЛЕНА ПЕРЕЧИСЛЕННЫЕ ПЕРЕМЕННЫЕ СО ЗНАЧЕНИЯМИ, РАВНЫМИ НУЛЮ. ОПЕРАТОР DIM УДОБНО ИСПОЛЬЗОВАТЬ ДЛЯ ОПРЕДЕЛЕНИЯ ПРОСТЫХ ТЕКСТОВЫХ ПЕРЕМЕННЫХ, ИМЕЮЩИХ ЗАДАННУЮ ДЛИНУ. НАПРИМЕР, ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА DIM F2\$=80,F3\$=80 БУДУТ ОТВЕДЕНЫ ОБЛАСТИ ОЗУ РАЗМЕРОМ ПО 80 БАЙТ ДЛЯ ПЕРЕМЕННЫХ F2\$ И F3\$, ЗНАЧЕНИЯ КОТОРЫХ БУДУТ РАВНЫ НУЛЮ (Т.Е. НЕ СОДЕРЖАТ СИМВОЛОВ).

ОПЕРАТОР SOM ИСПОЛЬЗУЕТСЯ ДЛЯ ОРГАНИЗАЦИИ ФАЙЛОВ ИНФОРМАЦИОННЫХ СТРОК. В ОПЕРАТОРЕ УКАЗЫВАЕТСЯ ИМЯ ФАЙЛА, И ЗА НИМ В КРУГЛЫХ СКОБКАХ ЧИСЛО - КОЛИЧЕСТВО К-БАЙТ, ОТВОДИМЫХ ДЛЯ ФАЙЛА. МАКСИМАЛЬНЫЙ РАЗМЕР ФАЙЛА - 16 К-БАЙТ.

ПРИМЕРЫ ОПЕРАТОРОВ DIM И SOM.

DIM A(15,20),A7%(M,N),F\$(10,N+20)=2

SOM FL(8),FL1(4),FL9(6)

И Т.П.

3.2. ОПЕРАТОР LET.

ОПЕРАТОР ВЫПОЛНЯЕТ СЛЕДУЮЩИЕ ФУНКЦИИ:

- А). ПРИСВАИВАНИЕ ЗНАЧЕНИЙ ЧИСЛОВЫМ ПЕРЕМЕННЫМ;
- Б). ПРИСВАИВАНИЕ ЗНАЧЕНИЙ ТЕКСТОВЫМ ПЕРЕМЕННЫМ;
- В). ИЗМЕНЕНИЕ РЕГИСТРОВ-УКАЗАТЕЛЕЙ СТРОК В ФАЙЛАХ FL;
- Г). ВВОД ПУСТЫХ СТРОК В ФАЙЛЫ FL;
- Д). ПЕРЕЗАПИСЬ СТРОК ИЗ ОДНОГО ФАЙЛА FL В ДРУГОЙ;

ОПЕРАТОР СОДЕРЖИТ В СЕБЕ РАВЕНСТВО, В КОТОРОМ В ЛЕВОЙ ЧАСТИ УКАЗЫВАЕТСЯ ОПРЕДЕЛЯЕМЫЙ ОПЕРАНД, В ПРАВОЙ ПРИСВАИВАЕМОЕ ЗНАЧЕНИЕ. ЕСЛИ НЕОБХОДИМО ПРИСВОИТЬ ОДНО И ТО ЖЕ ЗНАЧЕНИЕ НЕСКОЛЬКИМ ОПЕРАНДАМ, ТО ЭТИ ОПЕРАНДЫ УКАЗЫВАЮТСЯ В ЛЕВОЙ ЧАСТИ РАВЕНСТВА ЧЕРЕЗ ЗАПЯТУЮ.

В ЗАВИСИМОСТИ ОТ ВЫПОЛНЯЕМОЙ ФУНКЦИИ, В ЛЕВОЙ И ПРАВОЙ ЧАСТЯХ РАВЕНСТВА МОГУТ СОДЕРЖАТЬСЯ ЭЛЕМЕНТЫ, СОГЛАСНО СЛЕДУЮЩЕЙ ТАБЛИЦЕ:

ФУНКЦИЯ!	ЛЕВАЯ ЧАСТЬ	ПРАВАЯ ЧАСТЬ	ДЕЙСТВИЕ
А)	ЧИСЛОВАЯ ПЕРЕМЕННАЯ (РЕГИСТР RR, ЭЛ.-Т ИНФ. СТРОКИ IL, RL)	АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ	ПРИСВАИВАНИЕ ЗНАЧЕНИЯ
Б)	ТЕКСТОВАЯ ПЕРЕМЕННАЯ (РЕГИСТР TR, ЭЛЕМЕНТ TL)	ТЕКСТОВОЕ ВЫРАЖЕНИЕ	ТО ЖЕ
В)	НОМЕР СТРОКИ	АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ	УСТАНОВКА УКАЗАТЕЛЯ НА СТРОКУ С ЗАДАННЫМ ИЛИ НАИМЕНЕЕ ПРЕВЫШАЮЩИМ ЗАДАННЫМ НОМЕРОМ.
Г)	ОБОЗНАЧЕНИЕ ФАЙЛА	ТО ЖЕ	ВВОД В ФАЙЛ ПУСТОЙ СТРОКИ С ЗАДАННЫМ НОМЕРОМ
Д)	ОБОЗНАЧЕНИЕ ФАЙЛА <I>	ОБОЗНАЧЕНИЕ ФАЙЛА <J>	ПЕРЕЗАПИСЬ СТРОКИ ИЗ ФАЙЛА J В ФАЙЛ I

ПРИ ВВОДЕ СТРОК В СЛУЧАЯХ Г) И Д) ПРЕЖНЕЕ ЗНАЧЕНИЕ СТРОКИ С ДАННЫМ НОМЕРОМ СТИРАЕТСЯ. УКАЗАТЕЛЬ ФАЙЛА УСТАНОВЛИВАЕТСЯ НА ВВЕДЕННУЮ СТРОКУ.

ПРИМЕРЫ ОПЕРАТОРА LET:

- ФУНКЦИЯ А) LET R9=A+B%*C[2]
 LET Q1,Q2=SIN(PI*D)
- ФУНКЦИЯ Б) LET N\$='ИВАНОВ'
 LET S\$(I,J)=A\$+B\$+TL2(I)
- ФУНКЦИЯ В) LET NL=15
 LET NL,NL8=RR1*100+I%
- ФУНКЦИЯ Г) FL=NL2+D
- ФУНКЦИЯ Д) FL1,FL2,FL3=FL9

ПРИСВОЕНИЕ ЗНАЧЕНИЙ ЭЛЕМЕНТАМ ИНФОРМАЦИОННЫХ СТРОК С ПОМОЩЬЮ ОПЕРАТОРА LET В ОБЩЕМ СЛУЧАЕ ПРОИЗВОДИТСЯ В ДВА ЭТАПА:

- ЕСЛИ СТРОКА С ТРЕБУЕМЫМ НОМЕРОМ УЖЕ СУЩЕСТВУЕТ, ТО УСТАНОВЛИВАЕТСЯ УКАЗАТЕЛЬ СТРОК ДАННОГО ФАЙЛА (ФУНКЦИЯ В), ЕСЛИ СТРОКИ С ТРЕБУЕМЫМ НОМЕРОМ НЕТ, ЛИБО НЕОБХОДИМО СТЕРЕТЬ СУЩЕСТВУЮЩУЮ СТРОКУ, ТО В ФАЙЛ ВВОДИТСЯ ПУСТАЯ СТРОКА (ФУНКЦИЯ Г);
- ПРИСВАИВАЕТСЯ ЗНАЧЕНИЕ ЭЛЕМЕНТУ (ФУНКЦИИ А. И Б.).

3.3. ОПЕРАТОР INPUT.

ОПЕРАТОР ПРЕДНАЗНАЧЕН ДЛЯ ВВОДА ДАННЫХ В ПРОЦЕССЕ РАБОТЫ ПРОГРАММЫ ПОЛЬЗОВАТЕЛЯ. В ОПЕРАТОРЕ УКАЗЫВАЕТСЯ СПИСОК ОПРЕДЕЛЯЕМЫХ ЭЛЕМЕНТОВ, КОТОРЫЙ МОЖЕТ СОСТОЯТЬ ИЗ ЧИСЛОВЫХ ИЛИ ТЕКСТОВЫХ ПЕРЕМЕННЫХ ИЛИ ЭЛЕМЕНТОВ СТРОК. ПЕРЕМЕННЫЕ В СПИСКЕ РАЗДЕЛЯЮТСЯ ЗАПЯТЫМИ. НЕПОСРЕДСТВЕННО ПОСЛЕ ИМЕНИ ОПЕРАТОРА МОЖНО УКАЗАТЬ КОММЕНТАРИЙ, ОБОЗНАЧАЕМЫЙ В ВИДЕ ТЕКСТОВОЙ КОНСТАНТЫ. КОММЕНТАРИЙ ВЫВОДИТСЯ НА ДИСПЛЕЙ ПЕРЕД ВВОДОМ ЗНАЧЕНИЙ; ЕСЛИ КОММЕНТАРИЙ ОТСУТСТВУЕТ, ТО НА ДИСПЛЕЙ ВЫВОДИТСЯ ЗНАК <?>.

ВВОДИМЫЕ ЗНАЧЕНИЯ ДОЛЖНЫ СОСТОЯТЬ ИЗ ЧИСЛОВЫХ ИЛИ ТЕКСТОВЫХ КОНСТАНТ, КОТОРЫЕ РАЗДЕЛЯЮТСЯ МЕЖДУ СОБОЙ ДВОЕТОЧИЕМ.

ЧИСЛОВЫМ ЭЛЕМЕНТАМ СПИСКА ВВОДА ДОЛЖНЫ СООТВЕТСТВОВАТЬ ЧИСЛОВЫЕ КОНСТАНТЫ, ТЕКСТОВЫМ - ТЕКСТОВЫЕ. ТЕКСТОВЫЕ КОНСТАНТЫ ЗАКЛЮЧАТЬ В АПОСТРОФЫ ИЛИ КАВЫЧКИ НЕ ОБЯЗАТЕЛЬНО, ОДНАКО, В ЭТОМ СЛУЧАЕ НАЧАЛЬНЫЕ ПРОБЕЛЫ В КОНСТАНТАХ ВВЕДЕНА НЕ БУДУТ.

ПРИ ВВОДЕ ЧИСЛОВЫЕ ДАННЫЕ МОЖНО ОБОЗНАЧАТЬ УПРОЩЕННЫМИ АРИФМЕТИЧЕСКИМИ ВЫРАЖЕНИЯМИ, В КОТОРЫХ НЕ ДОПУСКАЕТСЯ ПРИМЕНЕНИЕ СЛУЖЕБНЫХ СЛОВ.

РАЗДЕЛИТЕЛЬНЫЕ ЗНАКИ МЕЖДУ ОТДЕЛЬНЫМИ КОНСТАНТАМИ МОЖНО ЗАМЕНЯТЬ ПРОБЕЛАМИ, ЗА ИСКЛЮЧЕНИЕМ СЛЕДУЮЩИХ СПЕЦИАЛЬНЫХ СЛУЧАЕВ:

- РАЗДЕЛИТЕЛЬ ДОЛЖЕН БЫТЬ УКАЗАН ЗА ТЕКСТОВОЙ КОНСТАНТОЙ, НЕ ЗАКЛЮЧЕННОЙ В АПОСТРОФЫ, ЕСЛИ ЗА НЕЙ УКАЗАНЫ ДРУГИЕ ДАННЫЕ;
- РАЗДЕЛИТЕЛЬ ДОЛЖЕН БЫТЬ ОБОЗНАЧЕН ВСЛЕД ЗА ЧИСЛОВОЙ КОНСТАНТОЙ, ЕСЛИ СТОЯЩЕЕ ЗА НЕЙ ЗНАЧЕНИЕ НАЧИНАЕТСЯ С АРИФМЕТИЧЕСКОГО ЗНАКА.

ЕСЛИ ДАННЫХ УКАЗАНО МЕНЬШЕ, ЧЕМ ЭЛЕМЕНТОВ В СПИСКЕ ВВОДА, ТО ОСТАВШИЕСЯ ПЕРЕМЕННЫЕ НЕ ОПРЕДЕЛЯЮТСЯ ИЛИ У НИХ ОСТАЮТСЯ ПРЕЖНИЕ ЗНАЧЕНИЯ. ЕСЛИ ДАННЫХ ПРИ ВВОДЕ УКАЗАНО БОЛЬШЕ, ТО ИНТЕРПРЕТАТОР ВЫДАЕТ СООБЩЕНИЕ ОБ ОШИБКЕ.

ЕСЛИ ИНТЕРПРЕТАТОРОМ ОБНАРУЖЕНА ОШИБКА ВО ВВОДИМЫХ ДАННЫХ, ТО НА ЭКРАН ВЫДАЕТСЯ СООБЩЕНИЕ "ПОВТОРИТЕ ВВОД", ПОСЛЕ ЧЕГО НЕОБХОДИМО НАБРАТЬ НОВЫЕ ДАННЫЕ.

ОПЕРАТОРОМ INPUT, КРОМЕ ПОЭЛЕМЕНТНОГО ВВОДА ДАННЫХ, ОПИСАННОГО ВЫШЕ, МОЖНО ПРИМЕНЯТЬ "ПОСТРОЧНЫЙ" ВВОД ДАННЫХ В МАССИВЫ ИНФОРМАЦИОННЫХ СТРОК. ДЛЯ ЭТОГО В ОПЕРАТОРЕ УКАЗЫВАЕТСЯ ИМЯ ФАЙЛА И ЗА НИМ, В КВАДРАТНЫХ СКОБКАХ, СПИСОК ВВОДИМЫХ ЭЛЕМЕНТОВ СТРОК, КОТОРЫЙ МОЖЕТ СОСТОЯТЬ ИЗ ЭЛЕМЕНТОВ IL, RL, TL. АРИФМЕТИЧЕСКИЕ ВЫРАЖЕНИЯ В ДАННЫХ ПРИМЕНЯТЬ НЕ ДОПУСКАЕТСЯ. НОМЕР ФАЙЛА В ОБОЗНАЧЕНИЯХ В СПИСКЕ ВВОДА УКАЗЫВАТЬ НЕ ОБЯЗАТЕЛЬНО.

ВВОДИМЫЕ ДАННЫЕ ПРИ ПОСТРОЧНОМ ВВОДЕ ДОЛЖНЫ НАЧИНАТЬСЯ С НОМЕРА ВВОДИМОЙ СТРОКИ. ИНТЕРПРЕТАТОР НАХОДИТ В ФАЙЛЕ УКАЗАННУЮ СТРОКУ И ПЕРЕПИСЫВАЕТ ЕЕ ИЗ ФАЙЛА В СПЕЦИАЛЬНЫЙ БУФЕР В СЛУЖЕБНОЙ ЗОНЕ ОЗУ. СТРОКА ДОЛЖНА УМЕЩАТЬСЯ В БУФЕР. РАЗМЕР БУФЕРА - 320 БАЙТ. ЕСЛИ СТРОКА С УКАЗАННЫМ НОМЕРОМ В ФАЙЛЕ НЕ НАЙДЕНА, ТО ИНТЕРПРЕТАТОР СОЗДАЕТ В БУФЕРЕ НОВУЮ СТРОКУ. ПОСЛЕ ВВОДА ВСЕХ ЗНАЧЕНИЙ СТРОКА ИЗ БУФЕРА ПЕРЕПИСЫВАЕТСЯ В ФАЙЛ, ПОСЛЕ ЧЕГО ИНТЕРПРЕТАТОР ЗАПРАШИВАЕТ ДАННЫЕ ДЛЯ СЛЕДУЮЩЕЙ СТРОКИ.

ПРИ ПОСТРОЧНОМ ВВОДЕ ОДНИМ ОПЕРАТОРОМ INPUT МОЖНО ВВЕСТИ НЕОГРАНИЧЕННОЕ ЧИСЛО СТРОК. ВЫХОД ИЗ ОПЕРАТОРА ОСУЩЕСТВЛЯЕТСЯ, ЕСЛИ ВМЕСТО ВВОДА ДАННЫХ БУДЕТ НАЖАТА КЛАВИША <ПС>.

3.4. ОПЕРАТОР LINPUT.

ОПЕРАТОР ПРИМЕНЯЕТСЯ ДЛЯ ВВОДА ТЕКСТОВОЙ ИНФОРМАЦИИ. В ОПЕРАТОРЕ УКАЗЫВАЮТСЯ ЧЕРЕЗ ЗАПЯТУЮ ОПРЕДЕЛЯЕМЫЕ ТЕКСТОВЫЕ ПЕРЕМЕННЫЕ. ПРИ ВВОДЕ ДАННЫХ КАЖДОЙ ПЕРЕМЕННОЙ ПРИСВАИВАЕТСЯ ЗНАЧЕНИЕ ОДНОЙ НАБРАННОЙ СТРОКИ, ПРИ ЭТОМ, ПЕРЕМЕННОЙ ПРИСВАИВАЕТСЯ ЗНАЧЕНИЕ ВСЕХ СИМВОЛОВ, СОСТАВЛЯЮЩИХ СТРОКУ, ВКЛЮЧАЯ НАЧАЛЬНЫЕ ПРОБЕЛЫ, ДВОЕТОЧИЯ, АПОСТРОФЫ И Т. Д.

КОММЕНТАРИИ В ОПЕРАТОРЕ НЕ ДОПУСКАЮТСЯ. ЗНАК <?> ПЕРЕД ВВОДОМ ЗНАЧЕНИЙ НА ДИСПЛЕЙ НЕ ВЫВОДИТСЯ.

3.5. ОПЕРАТОР READ.

В ТЕКСТЕ ПРОГРАММЫ МОЖЕТ БЫТЬ ОРГАНИЗОВАН БЛОК ДАННЫХ. СЧИТЫВАНИЕ ИНФОРМАЦИИ ИЗ БЛОКА ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ READ.

В ОПЕРАТОРЕ ЧЕРЕЗ ЗАПЯТУЮ УКАЗЫВАЮТСЯ ОПРЕДЕЛЯЕМЫЕ ЭЛЕМЕНТЫ, ПРЕДСТАВЛЯЮЩИЕ СОБОЙ ЧИСЛОВЫЕ ИЛИ ТЕКСТОВЫЕ ПЕРЕМЕННЫЕ. СПИСОК ЗНАЧЕНИЙ В БЛОКЕ ДОЛЖЕН СООТВЕТСТВОВАТЬ СПИСКУ ОПРЕДЕЛЯЕМЫХ ЭЛЕМЕНТОВ В READ АНАЛОГИЧНО ВВОДУ ДАННЫХ В ОПЕРАТОРЕ INPUT.

ВЫБОР ЗНАЧЕНИЙ ИЗ БЛОКА ПРОИСХОДИТ В СООТВЕТСТВИИ С УКАЗАТЕЛЕМ БЛОКА ДАННЫХ. ПОСЛЕ СЧИТЫВАНИЯ ОЧЕРЕДНОГО ЗНАЧЕНИЯ УКАЗАТЕЛЬ ПЕРЕМЕЩАЕТСЯ НА СЛЕДУЮЩЕЕ ЗНАЧЕНИЕ. КОЛИЧЕСТВО СЧИТЫВАНИЙ НЕ ДОЛЖНО ПРЕВЫШАТЬ КОЛИЧЕСТВО ЗНАЧЕНИЙ.

УСТАНОВКА УКАЗАТЕЛЯ БЛОКА ДАННЫХ НА НАЧАЛО ПРОИЗВОДИТСЯ ОПЕРАТОРОМ RESTORE И АВТОМАТИЧЕСКИ ИНТЕРПРЕТАТОРОМ ПРИ ЛЮБОЙ КОРРЕКТИРОВКЕ ПРОГРАММЫ И ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА RUN.

3.6. ОПЕРАТОР CLEAR.

ОПЕРАТОР ПРИМЕНЯЕТСЯ ДЛЯ ОЧИСТКИ ОЗУ. ОПЕРАТОР, В ЗАВИСИМОСТИ ОТ СВОЕГО СОДЕРЖИМОГО ВЫПОЛНЯЕТ СЛЕДУЮЩИЕ ДЕЙСТВИЯ:

- ПУСТОЙ ОПЕРАТОР СТИРАЕТ ВСЮ ПРОГРАММУ ПОЛЬЗОВАТЕЛЯ;
- ЕСЛИ УКАЗАНЫ НОМЕРА СТРОК <N1>-<N2>: СТИРАЕТ СЕГМЕНТ ПРОГРАММЫ СО СТРОКИ N1 ПО СТРОКУ N2. ЕСЛИ УКАЗАН N1 И ТИРЕ, ТО СТИРАЕТСЯ СЕГМЕНТ ОТ N1 ДО КОНЦА ПРОГРАММЫ, ЕСЛИ ТОЛЬКО N1, ТО СТРОКА С НОМЕРОМ N1;
- CLEAR DIM ИСКЛЮЧАЕТ ИЗ ОЗУ ВСЕ ПЕРЕМЕННЫЕ (МАССИВЫ ПЕРЕМЕННЫХ), ОБОЗНАЧЕННЫЕ ЛАТИНСКИМИ БУКВАМИ;
- CLEAR<СПИСОК ПЕРЕМЕННЫХ> ИСКЛЮЧАЕТ ИЗ ОЗУ УКАЗАННЫЕ МАССИВЫ ПЕРЕМЕННЫХ.
- CLEAR COM ИСКЛЮЧАЕТ ИЗ ОЗУ ВСЕ ФАЙЛЫ FL;
- CLEAR FL<I>(<N1>-<N2>) ОЧИЩАЕТ ФАЙЛ FL<I> С N1-Й ПО N2-Ю СТРОКУ. ЕСЛИ N1 И N2 НЕ УКАЗАНЫ, ТО ВЕСЬ ФАЙЛ. ОБЛАСТЬ ФАЙЛА ПРИ ЭТОМ ОСТАЕТСЯ. ИСКЛЮЧЕНИЕ ОБЛАСТИ ФАЙЛА ПРОИЗВОДИТСЯ ОПЕРАТОРОМ COM;
- CLEAR RR ОЧИЩАЕТ РЕГИСТРЫ RR;
- CLEAR TR ОЧИЩАЕТ РЕГИСТРЫ TR.

4. ИСПОЛЬЗОВАНИЕ НМЛ.

ИНТЕРПРЕТАТОР ОРГАНИЗУЕТ ЗАПИСЬ НА НМЛ СЛЕДУЮЩИХ ТИПОВ ИНФОРМАЦИИ:

- ПРОГРАММЫ И ИХ СЕГМЕНТЫ;
- ФАЙЛЫ ИНФОРМАЦИОННЫХ СТРОК И ИХ СЕГМЕНТЫ;
- ТЕКСТОВЫЕ ФАЙЛЫ;
- МАССИВЫ ПЕРЕМЕННЫХ.

ЗАПИСЬ ОСУЩЕСТВЛЯЕТСЯ ФАЙЛАМИ. ФАЙЛ ДАННЫХ МОЖЕТ ИМЕТЬ ИДЕНТИФИКАТОР, СОСТОЯЩИЙ ИЗ ТЕКСТОВОЙ КОНСТАНТЫ И ОДНОМЕРНОГО ИНДЕКСА. ТЕКСТОВАЯ КОНСТАНТА МОЖЕТ СОДЕРЖАТЬ ДО 15-ТИ СРАВНИВАЕМЫХ ЗНАКОВ. ИДЕНТИФИКАТОР ЗАПИСЫВАЕТСЯ В НАЧАЛЬНОМ БЛОКЕ ФАЙЛА. ФАЙЛ ЗАПИСЫВАЕТСЯ БЛОКАМИ ПО 256 БАЙТ. КОЛИЧЕСТВО БЛОКОВ, УМЕЩАЕМЫХ НА КАССЕТУ, ЗАВИСИТ ОТ СКОРОСТИ ПРОТЯЖКИ И СКОРОСТИ ПЕРЕДАЧИ ИНФОРМАЦИИ И ОПРЕДЕЛЯЕТСЯ ОПЫТНЫМ ПУТЕМ.

КОЛИЧЕСТВО БЛОКОВ В ОДНОМ ФАЙЛЕ ОПРЕДЕЛЯЕТСЯ ПО ФОРМУЛЕ:

$$M = \text{INT}((N+263)/252) + K$$

ГДЕ: M - КОЛИЧЕСТВО ЭЛЕМЕНТАРНЫХ БЛОКОВ ПО 256 БАЙТ;
N - РАЗМЕР ФАЙЛА, БАЙТ;
K - КОЛИЧЕСТВО ДУБЛЕЙ.

ПРИМЕНЯЕМАЯ ЛЕНТА ДОЛЖНА ИМЕТЬ ПРОЗРАЧНЫЙ РАКОРД.

ЗАПИСЬ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ SAVE. ФОРМАТ ОПЕРАТОРА:

SAVE<K><ИДЕНТИФИКАТОР><ЗАПИСЫВАЕМЫЕ ДАННЫЕ>

K - КОЛИЧЕСТВО ДУБЛЕЙ (ЦЕЛОЕ ЧИСЛО). ПО УМОЛЧАНИЮ K = 2.

В ИДЕНТИФИКАТОРЕ УКАЗЫВАЕТСЯ СНАЧАЛА ТЕКСТОВАЯ КОНСТАНТА, ЗАТЕМ ИНДЕКС. ИДЕНТИФИКАТОР МОЖНО НЕ УКАЗЫВАТЬ ИЛИ УКАЗАТЬ ТОЛЬКО ИНДЕКС ИЛИ ТОЛЬКО КОНСТАНТУ.

ЕСЛИ ЗАПИСЫВАЕМЫЕ ДАННЫЕ НЕ ОБОЗНАЧЕНЫ, ТО ЗАПИСЫВАЕТСЯ ВСЯ ПРОГРАММА, ЕСЛИ УКАЗАНЫ ГРАНИЧНЫЕ ЗНАЧЕНИЯ СТРОК, ТО СЕГМЕНТ ПРОГРАММЫ. ДЛЯ ЗАПИСИ ФАЙЛА ИНФОРМАЦИОННЫХ СТРОК НЕОБХОДИМО УКАЗАТЬ ИМЯ ФАЙЛА, ЗА КОТОРЫМ В СКОБКАХ МОЖНО УКАЗАТЬ ГРАНИЧНЫЕ ЗНАЧЕНИЯ СТРОК ДЛЯ ЗАПИСИ СЕГМЕНТА ФАЙЛА. ДЛЯ ЗАПИСИ МАССИВА ПЕРЕМЕННЫХ НЕОБХОДИМО УКАЗАТЬ ИМЯ МАССИВА.

ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА SAVE В РЕГИСТР RR0 ЗАНОСИТСЯ КОЛИЧЕСТВО БАЙТ ЗАПИСАННОЙ ИНФОРМАЦИИ.

ОПЕРАТОРОМ SAVE END ЗАПИСЫВАЕТСЯ ЗАПИСЬ "КОНЕЦ ЛЕНТЫ".

ЗАГРУЗКА ИНФОРМАЦИИ С НМЛ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ LOAD. В ОПЕРАТОРЕ УКАЗЫВАЕТСЯ ИДЕНТИФИКАТОР ФАЙЛА И ОБЛАСТЬ ОЗУ ДЛЯ ЗАГРУЗКИ ФАЙЛА.

ЕСЛИ ИДЕНТИФИКАТОР В LOAD НЕ УКАЗАН, ТО ЗАГРУЖАЕТСЯ ЛЮБОЙ ФАЙЛ, ЕСЛИ УКАЗАНА ТОЛЬКО ТЕКСТОВАЯ КОНСТАНТА, ТО СРАВНЕНИЕ ПРОИСХОДИТ ТОЛЬКО ПО КОНСТАНТЕ, ЕСЛИ ТОЛЬКО ИНДЕКС, ТО ТОЛЬКО ПО ИНДЕКСУ.

ФАЙЛЫ ИНФОРМАЦИОННЫХ СТРОК ЗАГРУЖАЮТСЯ В ПРЕДВАРИТЕЛЬНО ПОДГОТОВЛЕННЫЕ ДЛЯ НИХ ОПЕРАТОРОМ SOM ОБЛАСТИ. ДЛЯ ЭТОГО В ОПЕРАТОРЕ УКАЗЫВАЕТСЯ ИМЯ ФАЙЛА. ОСТАЛЬНАЯ ИНФОРМАЦИЯ ЗАГРУЖАЕТСЯ В СВОБОДНУЮ ОБЛАСТЬ ПОЛЬЗОВАТЕЛЯ, ПРИ ЭТОМ НИКАКИХ ОБОЗНАЧЕНИЙ В ОПЕРАТОРЕ НЕ ТРЕБУЕТСЯ. ТИП ДАННЫХ ИНТЕРПРЕТАТОР ОПРЕДЕЛЯЕТ АВТОМАТИЧЕСКИ.

ПРИ ЗАГРУЗКЕ ФАЙЛОВ ИНФОРМАЦИОННЫХ СТРОК СТИРАЕТСЯ ПРЕЖНЕЕ СОДЕРЖИМОЕ ФАЙЛОВ. ПРИ ЗАГРУЗКЕ ПРОГРАММЫ СТРОКИ ПО ПОРЯДКУ ВСТАВЛЯЮТСЯ В ПРОГРАММУ ПОЛЬЗОВАТЕЛЯ В СООТВЕТСТВИИ СО СВОИМИ НОМЕРАМИ. МАССИВ ПЕРЕМЕННЫХ ЗАПИСЫВАЕТСЯ СО СВОИМ ИМЕНЕМ, ПОЭТОМУ ПОСЛЕ ЗАГРУЗКИ ОН ВСТАВЛЯЕТСЯ В ОБЛАСТЬ DIM, ИСКЛЮЧАЯ МАССИВ С ТАКИМ ЖЕ ИМЕНЕМ.

ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА LOAD, ЕСЛИ ФАЙЛ БЫЛ НАЙДЕН, ТО В РЕГИСТР TR ЗАНОСИТСЯ ТЕКСТОВАЯ КОНСТАНТА, В РЕГИСТР RR0 - ИНДЕКС ИДЕНТИФИКАТОРА. В РЕГИСТР TR1 ЗАНОСИТСЯ ТИП ДАННЫХ, В РЕГИСТР RR1 - КОЛИЧЕСТВО БАЙТ В ФАЙЛЕ, В РЕГИСТР RR2 ЗАНОСИТСЯ ПОКАЗАТЕЛЬ КАЧЕСТВА ЗАПИСИ - ПРОЦЕНТ ПРАВИЛЬНО СЧИТАННЫХ БЛОКОВ.

ОБОЗНАЧЕНИЯ ТИПОВ ДАННЫХ, ЗАНОСИМЫЕ В РЕГИСТР TR1 ИМЕЮТ СЛЕДУЮЩУЮ РАСШИФРОВКУ:

VAS - ПРОГРАММА;

FL - ФАЙЛ ИНФОРМАЦИОННЫХ СТРОК ТИПА FL;

FT - ТЕКСТОВЫЙ ФАЙЛ;

<ИМЯ ПЕРЕМЕННОЙ> - МАССИВ ПЕРЕМЕННОЙ.

ПРОПУСК ФАЙЛОВ НА ЛЕНТЕ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ SKIP. В ОПЕРАТОРЕ УКАЗЫВАЕТСЯ ЧИСЛО - КОЛИЧЕСТВО ФАЙЛОВ, КОТОРОЕ НЕОБХОДИМО ПРОПУСТИТЬ. ПО УМОЛЧАНИЮ КОЛИЧЕСТВО ФАЙЛОВ - 1. ЕСЛИ ЛЕНТА УСТАНОВЛЕНА НА СЕРЕДИНЕ ФАЙЛА, ТО ОН ПРИНИМАЕТСЯ ЗА ПОЛНЫЙ, НАПРИМЕР, ЕСЛИ ЛЕНТА УСТАНОВЛЕНА НА СЕРЕДИНЕ ФАЙЛА, ТО ПУСТОЙ ОПЕРАТОР SKIP ПРОПУСТИТ ЛЕНТУ ДО КОНЦА ФАЙЛА И ВЫЗОВЕТ ОСТАНОВ HML.

ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА SKIP ПРОИСХОДИТ ЗАПОЛНЕНИЕ РЕГИСТРОВ АНАЛОГИЧНО ОПЕРАТОРУ LOAD, ЗА ИСКЛЮЧЕНИЕМ РЕГИСТРА RR2. В РЕГИСТРЫ ЗАНОСИТСЯ ИНФОРМАЦИЯ ИЗ ЗАГОЛОВКА ПОСЛЕДНЕГО ПРОПУЩЕННОГО ФАЙЛА. ЕСЛИ НИ ОДИН ЗАГОЛОВОК НЕ БЫЛ ПРОЧИТАН, ТО РЕГИСТРЫ НЕ ЗАПОЛНЯЮТСЯ.

РАБОТА ОПЕРАТОРОВ LOAD И SKIP ПРЕКРАЩАЕТСЯ ПО ПРОЧТЕНИЮ ЗАПИСИ "КОНЕЦ ЛЕНТЫ". ПРИ ЭТОМ ВЫРАБАТЫВАЕТСЯ ПРИЗНАК "КОНЕЦ ЛЕНТЫ", СОХРАНЯЕМЫЙ ИНТЕРПРЕТАТОРОМ ДО СЛЕДУЮЩЕГО ОБРАЩЕНИЯ К HML. ПО УКАЗАННОМУ ПРИЗНАКУ МОЖНО ОРГАНИЗОВАТЬ УСЛОВНЫЙ ПЕРЕХОД ОПЕРАТОРОМ IF END THEN.

ПЕРЕМОТКА ЛЕНТЫ НАЗАД ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ REW.

ОПЕРАТОР ОСУЩЕСТВЛЯЕТ ПЕРЕМОТКУ ЛЕНТЫ В НАЧАЛО. ВРЕМЯ ПЕРЕМОТКИ МОЖЕТ БЫТЬ ОГРАНИЧЕНО УКАЗАНИЕМ В ОПЕРАТОРЕ ИНДЕКСНОГО ВЫРАЖЕНИЯ, ОЗНАЧАЮЩЕГО ВРЕМЯ ПЕРЕМОТКИ В СЕКУНДАХ.

ПРИ СЧИТЫВАНИИ ИНФОРМАЦИИ С HML ОПЕРАТОРАМИ LOAD И SKIP ПОСЛЕ ПРАВИЛЬНОЙ ЗАГРУЗКИ ОЧЕРЕДНОГО БЛОКА НА ЭВМ ВКЛЮЧАЕТСЯ ИНДИКАТОР OP, КОТОРЫЙ ЗАТЕМ ВЫКЛЮЧАЕТСЯ ПЕРЕД СЧИТЫВАНИЕМ СЛЕДУЮЩЕГО БЛОКА.

ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА LOAD ПОСЛЕ НАХОЖДЕНИЯ ЗАДАННОГО ФАЙЛА НА ЭВМ ВКЛЮЧАЕТСЯ ИНДИКАТОР OM.

ПРИ ЗАПИСИ ИЛИ СЧИТЫВАНИИ ИНФОРМАЦИИ, ЕСЛИ ФАЙЛ НЕ УМЕЩАЕТСЯ НА ЛЕНТУ, ТО ПРИ ВЫХОДЕ НА ПРОЗРАЧНЫЙ РАКОРД ИНТЕРПРЕТАТОР ДЕЛАЕТ ОСТАНОВ, ВО ВРЕМЯ КОТОРОГО МОЖНО ВЫПОЛНИТЬ СЛЕДУЮЩИЕ ОПЕРАЦИИ:

- УСТАНОВИТЬ ДРУГУЮ КАСSETУ И ПРОДОЛЖИТЬ ЗАПИСЬ ИЛИ СЧИТЫВАНИЕ НАЖАТИЕМ НА ДИСПЛЕЕ КЛАВИШИ SU/Q (УПР/Q);

- ПРЕКРАТИТЬ ЗАПИСЬ ИЛИ СЧИТЫВАНИЕ И ПРОДОЛЖИТЬ РАБОТУ ПРОГРАММЫ НАЖАТИЕМ КЛАВИШИ SU/E (УПР/E). ПРАВИЛЬНАЯ ЗАГРУЗКА ФАЙЛА В ЭТОМ СЛУЧАЕ ВОЗМОЖНА, ЕСЛИ БЫЛ ПОЛНОСТЬЮ СЧИТАН ОДИН ДУБЛЬ;

- ПЕРЕЙТИ В РЕЖИМ "ДИАЛОГ" НАЖАТИЕМ КЛАВИШИ СУ/С (УПР/С). ЗАГРУЖЕННАЯ ЧАСТЬ ФАЙЛА ПРИ ЭТОМ СТИРАЕТСЯ.

5. ВЫВОД ИНФОРМАЦИИ НА ПЕЧАТЬ.

5.1. ОБЩИЕ СВЕДЕНИЯ.

ИНТЕРПРЕТАТОР ПОЗВОЛЯЕТ ВЫВОДИТЬ ИНФОРМАЦИЮ НА ПЕЧАТАЮЩЕЕ УСТРОЙСТВО ИЛИ ДИСПЛЕЙ. УСТРОЙСТВО ВЫВОДА ИНФОРМАЦИИ ОПРЕДЕЛЯЕТСЯ ЛОГИЧЕСКИМ НОМЕРОМ КАНАЛА ВЫВОДА. ДИСПЛЕЙ ИМЕЕТ НОМЕР КАНАЛА - 0, ПРИНТЕР - 1. НОМЕР КАНАЛА ОБОЗНАЧАЕТСЯ СИМВОЛОМ <#> И СЛЕДУЮЩИМ ЗА НИМ ИНДЕКСНЫМ ВЫРАЖЕНИЕМ. ЕСЛИ ПОСЛЕ ОБОЗНАЧЕНИЯ НОМЕРА КАНАЛА СТОИТ ЗАПЯТАЯ, ТО ИНТЕРПРЕТАТОРОМ ОНА ИГНОРИРУЕТСЯ.

ШИРИНА ПЕЧАТИ ОПРЕДЕЛЯЕТСЯ ПРИМЕНЯЕМЫМИ УСТРОЙСТВАМИ. ПОЗИЦИИ ПЕЧАТИ ПО ШИРИНЕ ЛИСТА ИМЕЮТ НОМЕРА ОТ 0 ДО МАКСИМАЛЬНОЙ. ИНФОРМАЦИЯ, ВЫВОДИМАЯ В ПОЗИЦИЯХ С НОМЕРАМИ 80 И ВЫШЕ НА ДИСПЛЕЕ НЕ ОТОБРАЖАЕТСЯ.

ЧИСЛА ВЫВОДЯТСЯ НА ПЕЧАТЬ В НАТУРАЛЬНОЙ ИЛИ ЕСТЕСТВЕННОЙ ФОРМЕ. ЗНАК "ПЛЮС" НЕ ВЫВОДИТСЯ.

ЧИСЛО, ВЫВОДИМОЕ В НАТУРАЛЬНОЙ ФОРМЕ ИМЕЕТ СЛЕДУЮЩИЙ ВИД:

<ЗНАК ЧИСЛА><12 ЦИФР МАНТИССЫ> Е <ЗНАК ПОРЯДКА> <ДВЕ ЦИФРЫ ПОРЯДКА>

ВМЕСТО ЗНАКА "ПЛЮС" ВЫВОДИТСЯ ПРОБЕЛ.

ФОРМАТ ПЕЧАТИ ЧИСЛА В НАТУРАЛЬНОМ ВИДЕ ИМЕЕТ ОБОЗНАЧЕНИЕ !E!.

ФОРМАТ ПЕЧАТИ ЧИСЛА В ЕСТЕСТВЕННОМ ВИДЕ ИМЕЕТ ОБОЗНАЧЕНИЕ !N1,N2!, ГДЕ N1- ЧИСЛО ЗНАКОВ ДО ТОЧКИ, ВКЛЮЧАЯ ЗНАК "МИНУС"; N2 - ЧИСЛО ЦИФР ПОСЛЕ ТОЧКИ. N1 И N2 МОГУТ ИМЕТЬ ЗНАЧЕНИЯ ОТ 0 ДО 9. ЕСЛИ N2 РАВНО НУЛЮ, ТО ТОЧКА НА ПЕЧАТЬ НЕ ВЫВОДИТСЯ. НЕЗНАЧАЩИЕ НУЛИ: ДО ТОЧКИ ЗАМЕНЯЮТСЯ ПРОБЕЛАМИ; ПОСЛЕ ТОЧКИ ВЫВОДЯТСЯ НА ПЕЧАТЬ. ЧИСЛО ПРИ ПЕЧАТИ ОКРУГЛЯЕТСЯ ДО N2 ЗНАКОВ ПОСЛЕ ТОЧКИ. ЕСЛИ ЧИСЛО НЕ УМЕЩАЕТСЯ В ЗАДАННЫЙ ФОРМАТ, ТО ОНО ПЕЧАТАЕТСЯ В НАТУРАЛЬНОМ ВИДЕ.

КРОМЕ УКАЗАННЫХ ВЫШЕ ФОРМАТОВ ИНТЕРПРЕТАТОР ОРГАНИЗУЕТ ПЕЧАТЬ ЧИСЕЛ В ФОРМАТЕ !A!, ОЗНАЧАЮЩЕМ АВТОМАТИЧЕСКУЮ УСТАНОВКУ ФОРМАТА ПЕЧАТИ ЧИСЕЛ, В ЗАВИСИМОСТИ ОТ КОЛИЧЕСТВА СОДЕРЖАЩИХСЯ В ЧИСЛЕ ЗНАЧАЩИХ ЦИФР. ДО ТОЧКИ И ПОСЛЕ ТОЧКИ МОЖЕТ БЫТЬ ИЗОБРАЖЕНО НЕ БОЛЕЕ 9-ТИ ЦИФР.

5.2. ОПЕРАТОР SET.

ОПЕРАТОР ПРЕДНАЗНАЧЕН ДЛЯ УСТАНОВЛЕНИЯ ПАРАМЕТРОВ ПЕЧАТИ. С ПОМОЩЬЮ ОПЕРАТОРА ОСУЩЕСТВЛЯЮТСЯ СЛЕДУЮЩИЕ ФУНКЦИИ:

А) РАЗМЕТКА ПОЛЯ ПЕЧАТИ. ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ ВИДА

SET TAB (<НОМЕРА ПОЗИЦИЙ>)

НОМЕРА ПОЗИЦИЙ ОБОЗНАЧАЮТСЯ ЧЕРЕЗ ЗАПЯТУЮ, НАЧИНАЯ С МЕНЬШЕЙ. ПРИ УСТАНОВЛЕНИИ ОЧЕРЕДНОЙ МЕТКИ ОПЕРАТОРОМ СТИРАЮТСЯ ВСЕ ПОСЛЕДУЮЩИЕ, Т.Е. С БОЛЬШИМИ НОМЕРАМИ. ОПЕРАТОР SET TAB (0) СТИРАЕТ ВСЕ МЕТКИ. МАКСИМАЛЬНОЕ КОЛИЧЕСТВО МЕТОК - 15. ПЕРЕХОД К МЕТКЕ ПРИ ПЕЧАТИ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ PRINT.

Б) ОГРАНИЧЕНИЕ ДЛИНЫ СТРАНИЦЫ, ВЫВОДИМОЙ НА ПРИНТЕР. ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ SET LNT (<КОЛИЧЕСТВО СТРОК>). ПОСЛЕ ВЫВОДА УКАЗАННОГО КОЛИЧЕСТВА СТРОК НА ПРИНТЕР ПРОИЗОЙДЕТ ОСТАНОВ, ВО ВРЕМЯ КОТОРОГО: ЕСЛИ БУДЕТ НАЖАТА НА ДИСПЛЕЕ КЛАВИША <ПС>, ТО НА ПЕЧАТЬ ВЫВЕДЕТСЯ ЕЩЕ ОДНА СТРОКА; ЕСЛИ ЛЮБАЯ ДРУГАЯ, ТО ЕЩЕ ОДНА СТРАНИЦА. ЕСЛИ АРГУМЕНТ ФУНКЦИИ LNT НЕ УКАЗАН ИЛИ РАВЕН НУЛЮ, ТО ДЛИНА ЛИСТА НЕ ОГРАНИЧИВАЕТСЯ.

В ОПЕРАТОРЕ SET ВСЕ ПАРАМЕТРЫ МОЖНО ОБОЗНАЧАТЬ ИНДЕКСНЫМИ ВЫРАЖЕНИЯМИ.

5.3. ОПЕРАТОР PRINT.

ОПЕРАТОР МОЖЕТ СОДЕРЖАТЬ В СЕБЕ СЛЕДУЮЩИЕ ЭЛЕМЕНТЫ:

- А) НОМЕР КАНАЛА ВЫВОДА. УСТАНОВЛИВАЕТСЯ КАНАЛ ВЫВОДА;
- Б) ФОРМАТ ПЕЧАТИ ЧИСЕЛ. УСТАНОВЛИВАЕТСЯ РЕГИСТР ФОРМАТА ПЕЧАТИ ЧИСЕЛ;
- В) АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ. ПРОИЗВОДИТСЯ ПЕЧАТЬ РЕЗУЛЬТАТА АРИФМЕТИЧЕСКОГО ВЫРАЖЕНИЯ В ЗАДАННОМ ФОРМАТЕ;
- Г) ТЕКСТОВОЕ ВЫРАЖЕНИЕ. ПРОИЗВОДИТСЯ ПЕЧАТЬ СИМВОЛОВ ТЕКСТОВОГО ВЫРАЖЕНИЯ;
- Д) ЗАПЯТАЯ. ВЫВОДИТ НА ПЕЧАТЬ ОДИН ПРОБЕЛ И УСТАНОВЛИВАЕТ ПОЗИЦИЮ ПЕЧАТИ НА БЛИЖАЙШУЮ МЕТКУ, УСТАНОВЛЕННУЮ ОПЕРАТОРОМ SET. ЕСЛИ МЕТКИ НЕТ, ТО ПРОИЗВОДИТСЯ ПЕРЕВОД СТРОКИ;
- Е) ТОЧКА С ЗАПЯТОЙ. ВЫЗЫВАЕТ ПЕЧАТЬ ОДНОГО ПРОБЕЛА;
- Ж) СЛОВО ТАВ. ЗА СЛОВОМ ТАВ УКАЗЫВАЕТСЯ ЦЕЛОЕ ЧИСЛО, ЛИБО В СКОБКАХ ЛЮБОЕ АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ, ОБОЗНАЧАЮЩЕЕ НОМЕР ПОЗИЦИИ ПЕЧАТИ, К КОТОРОЙ НУЖНО ПЕРЕЙТИ.

НАЗНАЧАЕМЫЕ НОМЕР КАНАЛА ВЫВОДА И ФОРМАТ ПЕЧАТИ ЧИСЕЛ ДЕЙСТВУЮТ ВО ВСЕХ ПОСЛЕДУЮЩИХ ОПЕРАТОРАХ PRINT ДО ИХ ЗАМЕНЫ ИЛИ ДО ВЫХОДА В РЕЖИМ "ДИАЛОГ".

ЕСЛИ В КОНЦЕ ОПЕРАТОРА СТОИТ АРИФМЕТИЧЕСКОЕ ИЛИ ТЕКСТОВОЕ ВЫРАЖЕНИЕ, ТО ПОСЛЕ ПЕЧАТИ ВСЕХ ДАННЫХ ПРОИЗВОДИТСЯ ПЕРЕВОД СТРОКИ. ПЕРЕВОД СТРОКИ, ТАКЖЕ, ВЫВОДИТСЯ ПУСТЫМ ОПЕРАТОРОМ.

5.3. ОПЕРАТОР LIST.

ОПЕРАТОР ИСПОЛЬЗУЕТСЯ ДЛЯ ПЕЧАТИ ТЕКСТА ПРОГРАММЫ. ИМЕЕТ СЛЕДУЮЩИЙ ФОРМАТ:

LIST<НОМЕР КАНАЛА ВЫВОДА>,<СПИСОК НОМЕРОВ СТРОК>

ЕСЛИ СПИСОК НОМЕРОВ СТРОК ОТСУТСТВУЕТ, ТО РАСПЕЧАТЫВАЕТСЯ ВСЯ ПРОГРАММА. СПИСОК МОЖЕТ СОДЕРЖАТЬ ОБОЗНАЧЕНИЯ СЕГМЕНТОВ В ВИДЕ <N1>-<N2>, ОЗНАЧАЮЩИХ ПЕЧАТЬ СЕГМЕНТА С N1-Й ПО N2-Ю СТРОКУ.

НАЗНАЧЕННЫЙ НОМЕР КАНАЛА ВЫВОДА ВЫВОДА ДЕЙСТВУЕТ ТОЛЬКО В ОДНОМ ОПЕРАТОРЕ.

5.5. МАКЕТЫ ПЕЧАТИ ИНФОРМАЦИИ.

ДЛЯ ПЕЧАТИ ТАБЛИЦ УДОБНО ИСПОЛЬЗОВАТЬ МАКЕТЫ ВЫВОДА ДАННЫХ, ПРЕДСТАВЛЯЮЩИЕ СОВОЙ ЗАПИСАННЫЕ В ТЕКСТОВЫХ ЭЛЕМЕНТАХ ЯЗЫКА СПЕЦИАЛЬНЫЕ ОБОЗНАЧЕНИЯ ФОРМАТОВ ВЫВОДА ЧИСЕЛ И ГРУПП СИМВОЛОВ.

ФОРМАТ ВЫВОДА ЧИСЛА ОБОЗНАЧАЕТСЯ СИМВОЛАМИ <#> И ТОЧКОЙ. СИМВОЛЫ <#> УКАЗЫВАЮТ ПОЗИЦИИ ПЕЧАТИ ЦИФР ЧИСЛА, ТОЧКА - ПОЗИЦИЮ ПЕЧАТИ ТОЧКИ. НАПРИМЕР, ФОРМАТ ПЕЧАТИ ЧИСЛА !3.2! ОБОЗНАЧАЕТСЯ В ВИДЕ <###.##>. ДО ТОЧКИ МОЖЕТ БЫТЬ ОБОЗНАЧЕНО ОТ 0 ДО 9 ЗНАКОВ, ПОСЛЕ ТОЧКИ ОТ 1 ДО 9 ЗНАКОВ <#>. ПРИ ПЕЧАТИ ЦЕЛЫХ ЧИСЕЛ ТОЧКА НЕ ОБОЗНАЧАЕТСЯ.

ФОРМАТ ВЫВОДА ТЕКСТОВЫХ ДАННЫХ ОБОЗНАЧАЕТСЯ: 1-Я ПОЗИЦИЯ СИМВОЛОМ <'> (АПОСТРОФ), ВСЕ ПОСЛЕДУЮЩИЕ БУКВАМИ <L>.

МАКЕТ МОЖЕТ СОДЕРЖАТЬ В СЕБЕ ОДНОВРЕМЕННО ОБОЗНАЧЕНИЯ ЧИСЛОВЫХ И ТЕКСТОВЫХ ФОРМАТОВ ПЕЧАТИ ДАННЫХ. КРОМЕ СИМВОЛОВ, ОБОЗНАЧАЮЩИХ ФОРМАТЫ ПЕЧАТИ, В МАКЕТЕ МОГУТ СОДЕРЖАТЬСЯ ЛЮБЫЕ ДРУГИЕ СИМВОЛЫ, КОТОРЫЕ ВЫВОДЯТСЯ НА ПЕЧАТЬ В СООТВЕТСТВУЮЩИХ ПОЗИЦИЯХ ПОЛЯ ПЕЧАТИ.

ПЕЧАТЬ ДАННЫХ В СООТВЕТСТВИИ С МАКЕТОМ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ PRINT С УКАЗАНИЕМ В НЕМ СЛУЖЕБНОГО СЛОВА USING. ФОРМАТ ОПЕРАТОРА:

```
PRINT USING <ТЕКСТОВЫЙ ЭЛЕМЕНТ>, <СПИСОК ВЫВОДА>
```

ТЕКСТОВЫЙ ЭЛЕМЕНТ (ЭЛЕМЕНТ TL, TR, ТЕКСТОВАЯ ПЕРЕМЕННАЯ ИЛИ КОНСТАНТА) СОДЕРЖИТ В СЕБЕ МАКЕТ ВЫВОДА ДАННЫХ, СПИСОК ВЫВОДА - ОБОЗНАЧЕНИЯ ДАННЫХ В СООТВЕТСТВИИ С МАКЕТОМ. ТЕКСТОВЫМ ФОРМАТАМ В МАКЕТЕ ДОЛЖНЫ СООТВЕТСТВОВАТЬ ТЕКСТОВЫЕ ДАННЫЕ В СПИСКЕ, ЧИСЛОВЫМ ФОРМАТАМ - ЧИСЛОВЫЕ. СПИСОК ВЫВОДА МОЖЕТ СОСТОЯТЬ ИЗ ТЕКСТОВЫХ И АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ, ОТДЕЛЬНЫЕ ЗНАЧЕНИЯ В СПИСКЕ МОЖНО РАЗДЕЛЯТЬ ЗАПЯТЫМИ. ДО СЛОВА USING В ОПЕРАТОРЕ PRINT МОГУТ БЫТЬ УКАЗАНЫ ЛЮБЫЕ ДРУГИЕ ЭЛЕМЕНТЫ, ДОПУСТИМЫЕ В ОПЕРАТОРЕ, ПОСЛЕ СПИСКА ВЫВОДА НИКАКИЕ ДРУГИЕ ОБОЗНАЧЕНИЯ НЕ ДОПУСКАЮТСЯ.

ЕСЛИ ТЕКСТОВЫЙ ЭЛЕМЕНТ СПИСКА МЕНЬШЕ СООТВЕТСТВУЮЩЕГО ТЕКСТОВОГО ФОРМАТА, ТО ПРИ ВЫВОДЕ ЭЛЕМЕНТ СПРАВА ДОПОЛНЯЕТСЯ ПРОБЕЛАМИ, ЕСЛИ БОЛЬШЕ, ТО ОСТАВШИЕСЯ СИМВОЛЫ НЕ ВЫВОДЯТСЯ. ЕСЛИ ЧИСЛО НЕ УМЕЩАЕТСЯ В ЗАДАННЫЙ ЧИСЛОВОЙ ФОРМАТ, ТО ВМЕСТО НЕГО ПЕЧАТАЕТСЯ СООТВЕТСТВУЮЩЕЕ ДЛИНЕ ФОРМАТА КОЛИЧЕСТВО СИМВОЛОВ <*>.

ЕСЛИ ВЫВОДИМЫХ ДАННЫХ В СПИСКЕ УКАЗАНО МЕНЬШЕ, ЧЕМ ФОРМАТОВ В МАКЕТЕ, ТО ОСТАВШИЕСЯ СИМВОЛЫ МАКЕТА ВЫВОДЯТСЯ НА ПЕЧАТЬ, ФОРМАТЫ ПРИ ЭТОМ ЗАМЕНЯЮТСЯ ПРОБЕЛАМИ. ЕСЛИ ДАННЫХ УКАЗАНО БОЛЬШЕ, ТО МАКЕТ ПОВТОРЯЕТСЯ. ЕСЛИ НЕОБХОДИМО ПРОПУСТИТЬ КАКОЙ-ЛИБО ФОРМАТ БЕЗ ПЕЧАТИ, ТО В СООТВЕТСТВУЮЩЕМ МЕСТЕ СПИСКА ВМЕСТО ДАННЫХ ОБОЗНАЧАЕТСЯ ЗАПЯТАЯ.

ПОСЛЕ ВЫВОДА ВСЕГО СПИСКА НА ПЕЧАТЬ ВЫВОДИТСЯ ПЕРЕВОД СТРОКИ.

ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ МАКЕТОВ.

1. ТРЕБУЕТСЯ РАСПЕЧАТАТЬ В ОДНУ СТРОКУ:

- СЛОВО "ИТОГО:" ;
- 10 СИМВОЛОВ ПЕРЕМЕННОЙ Z\$(I,J) ;
- ПЕРЕМЕННЫЕ S И S1 В ФОРМАТЕ !5.1!;
- РЕЗУЛЬТАТ АРИФМЕТИЧЕСКОГО ВЫРАЖЕНИЯ S1/S*100 В ФОРМАТЕ !3.1!.
- МЕЖДУ ОТДЕЛЬНЫМИ ДАННЫМИ ОБОЗНАЧИТЬ ОДИН ПРОБЕЛ.

ЗАДАННУЮ СТРОКУ РАСПЕЧАТЫВАЕТ СЛЕДУЮЩИЙ СЕГМЕНТ ПРОГРАММЫ:

```
100 M1$= "ИТОГО: 'LLLLLLLLLL #####.# #####.# ###.##"
```

```
110 PRINT USING M1$,Z$(I,J),S,S1,S1/S*100
```

2. ТРЕБУЕТСЯ РАСПЕЧАТАТЬ В ОДНУ СТРОКУ ПО ПОРЯДКУ ПЕРЕМЕННЫЕ X,Y,X1,Y1,X2,Y2. ПЕРЕМЕННЫЕ X,X1,X2 РАСПЕЧАТАТЬ В ФОРМАТЕ !3.0!, ПЕРЕМЕННЫЕ Y,Y1,Y2 В ФОРМАТЕ !0.3!.

ПЕЧАТЬ ОСУЩЕСТВЛЯЕТСЯ СЛЕДУЮЩЕЙ СТРОКОЙ ПРОГРАММЫ:

```
700 PRINT USING '### .### 'X,Y,X1,Y1,X2,Y2
```

6. УПРАВЛЕНИЕ ХОДОМ РАБОТЫ ПРОГРАММЫ.

6.1. ОПЕРАТОР IF.

ОПЕРАТОР ОРГАНИЗУЕТ УСЛОВНЫЕ ПЕРЕХОДЫ. В ОПЕРАТОРЕ СОДЕРЖИТСЯ КАКОЕ-ЛИБО УСЛОВИЕ. ЕСЛИ УСЛОВИЕ ВЫПОЛНЯЕТСЯ, ТО ВЫПОЛНЯЕТСЯ ОПЕРАТОР, СТОЯЩИЙ ЗА СЛОВОМ THEN, ЕСЛИ НЕТ, ТО СТОЯЩИЙ ЗА СЛОВОМ ELSE. СЛОВА THEN И ELSE ДОЛЖНЫ РАСПОЛАГАТЬСЯ В ОДНОЙ СТРОКЕ С ОПЕРАТОРОМ IF. ЕСЛИ СЛОВО ELSE ОТСУТСТВУЕТ, ТО ПРИ НЕВЫПОЛНЕНИИ УСЛОВИЯ ВЫПОЛНЯЕТСЯ СЛЕДУЮЩАЯ СТРОКА ПРОГРАММЫ.

ВСЛЕД ЗА СЛОВАМИ THEN И ELSE МОГУТ РАСПОЛАГАТЬСЯ ЛЮБЫЕ ОПЕРАТОРЫ, КОЛИЧЕСТВО КОТОРЫХ ОГРАНИЧЕНО ДЛИНОЙ СТРОКИ ПРОГРАММЫ. ОПЕРАТОР GOTO ВМЕСТО СЛОВА THEN ПРИМЕНЯТЬ НЕ ДОПУСКАЕТСЯ.

В ОПЕРАТОРЕ IF МОЖЕТ РАСПОЛАГАТЬСЯ НЕСКОЛЬКО УСЛОВИЙ, РАЗДЕЛЕННЫХ МЕЖДУ СОБОЙ СЛОВАМИ OR ИЛИ AND. ЕСЛИ УСЛОВИЯ РАЗДЕЛЕННЫ СЛОВАМИ OR, ТО ПЕРЕХОД К ОПЕРАТОРУ, СТОЯЩЕМУ ЗА СЛОВОМ THEN ОСУЩЕСТВЛЯЕТСЯ ЕСЛИ ИСТИННО ХОТЯ БЫ ОДНО ИЗ УСЛОВИЙ. ЕСЛИ УСЛОВИЯ РАЗДЕЛЕННЫ СЛОВАМИ AND, ТО УКАЗАННЫЙ ПЕРЕХОД ОСУЩЕСТВЛЯЕТСЯ ЕСЛИ ИСТИННЫ ВСЕ УСЛОВИЯ.

УСЛОВИЕ МОЖЕТ БЫТЬ ЗАДАНО:

А). РЕЗУЛЬТАТОМ СРАВНЕНИЯ АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ. В ЭТОМ СЛУЧАЕ УКАЗЫВАЮТСЯ ДВА АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЯ, РАЗДЕЛЕННЫЕ ОДНИМ ИЛИ ДВУМЯ ЗНАКАМИ СООТНОШЕНИЙ: = , < ИЛИ > . УСЛОВИЕ ВЫПОЛНЯЕТСЯ ЕСЛИ СПРАВЕДЛИВ ХОТЯ БЫ ОДИН ИЗ ЗНАКОВ СООТНОШЕНИЙ.

Б). РЕЗУЛЬТАТОМ СРАВНЕНИЯ ТЕКСТОВЫХ ВЫРАЖЕНИЙ. УКАЗЫВАЮТСЯ ЧЕРЕЗ ЗНАК "РАВНО" ДВА ТЕКСТОВЫХ ВЫРАЖЕНИЯ. УСЛОВИЕ ИСТИННО, ЕСЛИ ВСЕ СИМВОЛЫ ВО ВТОРОМ ВЫРАЖЕНИИ СООТВЕТСТВУЮТ СИМВОЛАМ В ПЕРВОМ ВЫРАЖЕНИИ. СРАВНЕНИЕ ПРОИЗВОДИТСЯ ПО КОЛИЧЕСТВУ СИМВОЛОВ, СОДЕРЖАЩЕМУСЯ В ПЕРВОМ ВЫРАЖЕНИИ.

В). СОСТОЯНИЕМ ПРИЗНАКА "КОНЕЦ ЛЕНТЫ". ОПЕРАТОР ИМЕЕТ ВИД:

IF END THEN

Г). СОСТОЯНИЕМ УКАЗАТЕЛЯ ФАЙЛА, УКАЗЫВАЮЩЕГО НА КОНЕЦ ФАЙЛА. ОПЕРАТОР ИМЕЕТ ВИД: IF END FL THEN

6.2. ОРГАНИЗАЦИЯ ЦИКЛОВ.

ЦИКЛЫ ОРГАНИЗУЮТСЯ ОПЕРАТОРАМИ FOR И NEXT.

ОПЕРАТОР FOR ЗАПИСЫВАЕТСЯ В СЛЕДУЮЩЕМ ВИДЕ:

FOR <A> = <P> TO <Q> STEP <D>

ГДЕ: А - ИМЯ УПРАВЛЯЮЩЕЙ ПЕРЕМЕННОЙ;
Р - НАЧАЛЬНОЕ ЗНАЧЕНИЕ;
Q - ГРАНИЧНОЕ ЗНАЧЕНИЕ;
D - ШАГ ИЗМЕНЕНИЯ УПРАВЛЯЮЩЕГО ЭЛЕМЕНТА.

В КАЧЕСТВЕ УПРАВЛЯЮЩЕЙ ПЕРЕМЕННОЙ МОЖЕТ БЫТЬ ИСПОЛЬЗОВАНА ЧИСЛОВАЯ ПЕРЕМЕННАЯ БЕЗ ИНДЕКСА, ЛИБО РЕГИСТР RR. ОСТАЛЬНЫЕ ПАРАМЕТРЫ В ОПЕРАТОРЕ МОГУТ БЫТЬ ОБОЗНАЧЕНЫ АРИФМЕТИЧЕСКИМИ ВЫРАЖЕНИЯМИ.

ЕСЛИ ШАГ НЕ УКАЗАН, ТО ОН ПРИНИМАЕТСЯ РАВНЫМ ЕДИНИЦЕ.

ОПЕРАТОР NEXT ЗАПИСЫВАЕТСЯ В ВИДЕ: NEXT<A> . ЕСЛИ ПОСЛЕ ПРИБАВЛЕНИЯ ШАГА ЗНАЧЕНИЕ УПРАВЛЯЮЩЕГО ЭЛЕМЕНТА ПРИ ПОЛОЖИТЕЛЬНОМ ШАГЕ БОЛЬШЕ ГРАНИЧНОГО, А ПРИ ОТРИЦАТЕЛЬНОМ МЕНЬШЕ ГРАНИЧНОГО, ТО ОПЕРАТОР ОСУЩЕСТВЛЯЕТ ВЫХОД ИЗ ЦИКЛА И ВЫПОЛНЯЕТСЯ СЛЕДУЮЩИЙ ЗА NEXT ОПЕРАТОР, В ОСТАЛЬНЫХ СЛУЧАЯХ ПРОИСХОДИТ ВОЗВРАТ К ОПЕРАТОРУ , СЛЕДУЮЩЕМУ ЗА FOR.

ПРИ РАБОТЕ С ФАЙЛАМИ ИНФОРМАЦИОННЫХ СТРОК ИСПОЛЬЗУЕТСЯ ОПЕРАТОР ВИДА:

FOR FL<I> (<N1>-<N2>)

ГДЕ: I - НОМЕР ФАЙЛА, N1, N2 - ГРАНИЧНЫЕ ЗНАЧЕНИЯ.

ОПЕРАТОР УСТАНОВЛИВАЕТ УКАЗАТЕЛЬ СТРОК ФАЙЛА НА СТРОКУ N1. ОПЕРАТОР NEXT FL<I> УСТАНОВЛИВАЕТ УКАЗАТЕЛЬ НА СЛЕДУЮЩУЮ СТРОКУ, И ЕСЛИ УКАЗАТЕЛЬ ПРИ ЭТОМ УСТАНОВЛИВАЕТСЯ НА СТРОКУ С НОМЕРОМ, БОЛЬШИМ N2 ИЛИ НА КОНЕЦ ФАЙЛА, ТО ПРОИСХОДИТ ВЫХОД ИЗ ЦИКЛА. ЕСЛИ N1 И N2 НЕ ОБОЗНАЧЕНЫ, ТО В ЦИКЛЕ УЧАСТВУЕТ ВСЬ ФАЙЛ.

ЦИКЛЫ ВЫПОЛНЯЮТСЯ, КАК МИНИМУМ, ОДИН РАЗ. НЕ ДОПУСКАЕТСЯ ВЫХОД ИЗ ЦИКЛА ВЫШЕГО УРОВНЯ, ЕСЛИ НЕ ВЫПОЛНЕН ЦИКЛ НИЖЕГО УРОВНЯ. ДЛЯ ПРЕЖДЕВРЕМЕННОГО ВЫХОДА ИЗ ЦИКЛА ИСПОЛЬЗУЕТСЯ ОПЕРАТОР RESTORE FOR, ОСУЩЕСТВЛЯЮЩИЙ ПСЕВДОВОЗВРАТ ИЗ ОДНОГО УРОВНЯ ЦИКЛОВ. МАКСИМАЛЬНЫЙ УРОВЕНЬ ВЛОЖЕНИЯ ЦИКЛОВ - 8.

6.3. БЕЗУСЛОВНЫЕ ПЕРЕХОДЫ И ПОДПРОГРАММЫ.

БЕЗУСЛОВНЫЕ ПЕРЕХОДЫ В ПРОГРАММЕ ОСУЩЕСТВЛЯЮТ ОПЕРАТОРЫ GOTO<N> И ON<N>, ГДЕ N - НОМЕР СТРОКИ ПЕРЕХОДА. В ОПЕРАТОРЕ GOTO НОМЕР СТРОКИ ЗАДАЕТСЯ ЦЕЛЫМ ЧИСЛОМ, В ОПЕРАТОРЕ ON - АРИФМЕТИЧЕСКИМ ВЫРАЖЕНИЕМ.

ОПЕРАТОР ON, КРОМЕ ТОГО, МОЖЕТ БЫТЬ ЗАПИСАН В СЛЕДУЮЩЕМ ВИДЕ:

ON <УПРАВЛЯЮЩИЙ ПАРАМЕТР> GOTO <СПИСОК НОМЕРОВ СТРОК>

ЛИБО

ON <УПРАВЛЯЮЩИЙ ПАРАМЕТР> GOSUB <СПИСОК НОМЕРОВ СТРОК>

ОПЕРАТОР, В ЗАВИСИМОСТИ ОТ ВЕЛИЧИНЫ УПРАВЛЯЮЩЕГО ПАРАМЕТРА, ОПРЕДЕЛЯЮЩЕГО ИНДЕКС НОМЕРА СТРОКИ В СПИСКЕ, ВЫБИРАЕТ СТРОКУ ИЗ СПИСКА И ДЕЛАЕТ К НЕЙ БЕЗУСЛОВНЫЙ ПЕРЕХОД (GOTO), ЛИБО ВЫПОЛНЯЕТ, НАЧИНАЯ С ДАННОЙ СТРОКИ, ПОДПРОГРАММУ (GOSUB). УПРАВЛЯЮЩИЙ ПАРАМЕТР И НОМЕРА СТРОК МОГУТ БЫТЬ ОБОЗНАЧЕНЫ АРИФМЕТИЧЕСКИМИ ВЫРАЖЕНИЯМИ. НОМЕРА СТРОК В СПИСКЕ РАЗДЕЛЯЮТСЯ ЗАПЯТЫМИ.

ПЕРЕХОД К ПОДПРОГРАММЕ ПОЛЬЗОВАТЕЛЯ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ GOSUB<N>, ГДЕ N - НОМЕР ПЕРВОЙ СТРОКИ ПОДПРОГРАММЫ.

ВОЗВРАТ ИЗ ПОДПРОГРАММ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ RETURN. МАКСИМАЛЬНЫЙ УРОВЕНЬ ВЛОЖЕНИЯ ПОДПРОГРАММ - 8.

6.4. ОПЕРАТОР RUN.

ОПЕРАТОР ПО ПОРЯДКУ ВЫПОЛНЯЕТ СЛЕДУЮЩИЕ ДЕЙСТВИЯ:

- ИСКЛЮЧАЕТ ИЗ ОЗУ ОБЛАСТЬ DIM ;
- ОЧИЩАЕТ РЕГИСТРЫ TR И RR ;
- УСТАНОВЛИВАЕТ В НАЧАЛО УКАЗАТЕЛЬ БЛОКА ДАННЫХ ;
- УСТАНОВЛИВАЕТ В ИСХОДНОЕ СОСТОЯНИЕ РАЗМЕТКУ ПОЛЯ ПЕЧАТИ ;
- НАЧИНАЕТ ВЫПОЛНЕНИЕ ПРОГРАММЫ СО СТРОКИ С НАИМЕНЬШИМ НОМЕРОМ.

6.5. ОПЕРАТОРЫ END, STOP И START.

ОПЕРАТОР END ОСУЩЕСТВЛЯЕТ БЕЗУСЛОВНЫЙ ПЕРЕХОД В РЕЖИМ "ДИАЛОГ".

ОПЕРАТОР STOP ПО ПОРЯДКУ ВЫПОЛНЯЕТ СЛЕДУЮЩИЕ ДЕЙСТВИЯ:

- ОПРЕДЕЛЯЕТ АДРЕС СЛЕДУЮЩЕЙ СТРОКИ ПРОГРАММЫ И ЕЕ НОМЕР (N);
- ВЫВОДИТ НА ДИСПЛЕЙ СООБЩЕНИЕ "ОСТАНОВ В СТРОКЕ <N>";
- ЗАПОЛНЯЕТ ВЕКТОР ОСТАНОВА, В КОТОРЫЙ ЗАНОСИТ АДРЕС СТРОКИ N, ТЕКУЩИЙ ФОРМАТ ПЕЧАТИ ЧИСЕЛ И НОМЕР КАНАЛА ВЫВОДА В PRINT, СОДЕРЖИМОЕ СЧЕТЧИКА СТРОК ПРИНТЕРА И СОДЕРЖИМОЕ РЕГИСТРОВ - УКАЗАТЕЛЕЙ СТЕКОВ ЦИКЛОВ И ПОДПРОГРАММ;
- ВЫРАБАТЫВАЕТ "ПРИЗНАК ОСТАНОВА";
- ПЕРЕВОДИТ ИНТЕРПРЕТАТОР В РЕЖИМ "ДИАЛОГ".

ДАЛЕЕ ВОЗМОЖЕН ДИАЛОГ, ПОСЛЕ КОТОРОГО, ЕСЛИ НЕ БЫЛ ПОТЕРЯН "ПРИЗНАК ОСТАНОВА", МОЖЕТ БЫТЬ ОСУЩЕСТВЛЕН ВОЗВРАТ К ПРЕРВАННОМУ СОСТОЯНИЮ ПРОГРАММЫ. ВОЗВРАТ ИЗ ОСТАНОВА ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ START.

"ПРИЗНАК ОСТАНОВА" НЕ СОХРАНЯЕТСЯ ЕСЛИ В ПРОЦЕССЕ ДИАЛОГА ВЫПОЛНЯЛИСЬ СЛЕДУЮЩИЕ ОПЕРАЦИИ:

- КОРРЕКТИРОВКА ПРОГРАММЫ;
- ВЫВОД ИНФОРМАЦИИ НА ПРИНТЕР;
- ИСПОЛЬЗОВАЛИСЬ ОПЕРАТОРЫ GOSUB И FOR.

ОПЕРАТОР STOP МОЖЕТ БЫТЬ ВЫЗВАН ПРИ ВЫПОЛНЕНИИ ПРОГРАММЫ НАЖАТИЕМ ЛЮБОЙ ИЗ КЛАВИШ НА ДИСПЛЕЕ. ПРИ ЭТОМ ВЫРАБАТЫВАЕТСЯ ПРИЗНАК "ЗАПРОС НА ОСТАНОВ" ПРИ КОТОРОМ ВО ВРЕМЯ ПЕРЕХОДА К СЛЕДУЮЩЕЙ СТРОКЕ ПРОГРАММЫ ИНТЕРПРЕТАТОР ВЫПОЛНЯЕТ ОПЕРАТОР STOP.

ОПЕРАТОР END ПРИ ВЫПОЛНЕНИИ ПРОГРАММЫ МОЖЕТ БЫТЬ ВЫЗВАН НАЖАТИЕМ НА ДИСПЛЕЕ КЛАВИШИ СУ/С (УПР/С).

// Конец файла k35a_index-1. Далее идёт текст из файла k35a_index-2:

7. ПРОЧИЕ ОПЕРАТОРЫ.

7.1 ОПЕРАТОР CHANGE.

ОПЕРАТОР ПРЕДНАЗНАЧЕН ДЛЯ ПЕРЕВОДА СИМВОЛОВ В ИХ ДЕСЯТИЧНЫЕ КОДЫ И ОБРАТНО.

В ОПЕРАТОРЕ ЧЕРЕЗ СЛОВО "ТО" УКАЗЫВАЮТСЯ ТЕКСТОВОЕ ВЫРАЖЕНИЕ И ИМЯ МАССИВА ЧИСЛОВЫХ ПЕРЕМЕННЫХ ВЕЩЕСТВЕННОГО ТИПА.

ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА ВИДА:

CHANGE <ТЕКСТОВОЕ ВЫРАЖЕНИЕ> ТО <ИМЯ МАССИВА ПЕРЕМЕННЫХ>

В МАССИВ ПЕРЕМЕННЫХ, НАЧИНАЯ С ПЕРЕМЕННОЙ С ИНДЕКСОМ 1, ЗАПИСЫВАЮТСЯ ПОСЛЕДОВАТЕЛЬНО ВСЕ КОДЫ СИМВОЛОВ ТЕКСТОВОГО ВЫРАЖЕНИЯ, А В ЭЛЕМЕНТ С ИНДЕКСОМ 0 ЗАДАЕТСЯ КОЛИЧЕСТВО СИМВОЛОВ В ТЕКСТОВОМ ВЫРАЖЕНИИ. МАССИВ ПЕРЕМЕННЫХ ЗАДАЕТСЯ ЗАРАНЕЕ ОПЕРАТОРОМ DIM; ОН ДОЛЖЕН БЫТЬ ОДНОМЕРНЫМ. ЕСЛИ В МАССИВЕ МЕНЬШЕ ПЕРЕМЕННЫХ, ЧЕМ СИМВОЛОВ В ТЕКСТОВОМ ВЫРАЖЕНИИ, ТО ОСТАВШИЕСЯ СИМВОЛЫ ЕСЛИ БОЛЬШЕ, ТО ОСТАВШИМСЯ ПЕРЕМЕННЫМ ПРИСВАИВАЕТСЯ ЗНАЧЕНИЕ КОДА ПРОБЕЛА - 32.

ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА ВИДА:

CHANGE <ИМЯ МАССИВА ПЕРЕМЕННЫХ> TO <ТЕКСТОВАЯ ПЕРЕМЕННАЯ>

ВСЕ ЗНАЧЕНИЯ ПЕРЕМЕННЫХ, НАЧИНАЯ С ПЕРЕМЕННОЙ С ИНДЕКСОМ 1, ЗАПИСЫВАЮТСЯ В ТЕКСТОВУЮ ПЕРЕМЕННУЮ. ВМЕСТО СИМВОЛОВ, НЕДОПУСТИМЫХ В АЛФАВИТЕ ЯЗЫКА ЗАПИСЫВАЮТСЯ ПРОБЕЛЫ. ДЛИНУ ТЕКСТОВОЙ ПЕРЕМЕННОЙ МОЖНО ОГРАНИЧИТЬ ЗАПИСЬЮ В ПЕРЕМЕННУЮ С ИНДЕКСОМ 0 СООТВЕТСТВУЮЩЕГО ЗНАЧЕНИЯ.

ПРИМЕР ИСПОЛЬЗОВАНИЯ ОПЕРАТОРА.

В ТЕКСТОВОЙ ПЕРЕМЕННОЙ L2\$ СОДЕРЖИТСЯ ТЕКСТ "ИВАНОВ, ПЕТРОВ". ТРЕБУЕТСЯ ИЗ ЗНАЧЕНИЯ ПЕРЕМЕННОЙ ВЫДЕЛИТЬ СЕГМЕНТ ОТ НАЧАЛА ДО ПЕРВОЙ ЗАПЯТОЙ И ЗАПИСАТЬ ЕГО В ВИДЕ ПЕРЕМЕННОЙ F1\$. ДАННУЮ ЗАДАЧУ ВЫПОЛНЯЕТ СЕГМЕНТ ПРОГРАММЫ:

```
10 DIM A(20)
20 CHANGE L2$ TO A
30 FOR J=1 TO A
40 IF A(J)=44 THEN RESTORE FOR:A=J-1 ELSE NEXT J
50 CHANGE A TO F1$: END
```

7.2. ОПЕРАТОР REN.

ОПЕРАТОР ПРЕДНАЗНАЧЕН ДЛЯ ПЕРЕИМЕНОВАНИЯ ФАЙЛОВ ИНФОРМАЦИОННЫХ СТРОК И МАССИВОВ ПЕРЕМЕННЫХ. ОПЕРАТОР ИМЕЕТ СЛЕДУЮЩИЙ ФОРМАТ:

REN<ИМЯ1>TO<ИМЯ2>

В РЕЗУЛЬТАТЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА ФАЙЛУ, ЛИБО МАССИВУ ПЕРЕМЕННЫХ, С ИМЕНЕМ 1 ПРИСВАИВАЕТСЯ ИМЯ 2. ЕСЛИ В ОЗУ УЖЕ ИМЕЮТСЯ ДАННЫЕ С ИМЕНЕМ 2, ТО ЭТИМ ДАННЫМ ПРИСВАИВАЕТСЯ ИМЯ 1, Т.Е. ПРОИСХОДИТ ОБМЕН ИМЕНАМИ.

ТИП ДАННЫХ, ОБОЗНАЧЕННЫХ ИМЕНЕМ 2, ДОЛЖЕН СООТВЕТСТВОВАТЬ ТИПУ ДАННЫХ С ИМЕНЕМ 1.

ТЕЛО ОПЕРАТОРА REN МОЖЕТ ПОВТОРЯТЬСЯ ЧЕРЕЗ ЗАПЯТУЮ.

ПРИМЕР ИСПОЛЬЗОВАНИЯ ОПЕРАТОРА:

- ТРЕБУЕТСЯ ОПРЕДЕЛИТЬ БОЛЬШЕЕ ИЗ ДВУХ ЗНАЧЕНИЙ ПЕРЕМЕННЫХ A0 И B0 И ПРИСВОИТЬ ЕГО ПЕРЕМЕННОЙ X, МЕНЬШЕЕ ЗНАЧЕНИЕ ПРИСВОИТЬ ПЕРЕМЕННОЙ Y.

ДАННУЮ ЗАДАЧУ ВЫПОЛНЯЕТ СЛЕДУЮЩАЯ СТРОКА ПРОГРАММЫ:

```
100 X=A0:Y=B0:IF X<Y THEN REN X TO Y
```

ОПЕРАТОР REN УДОБНО ИСПОЛЬЗОВАТЬ ДЛЯ ОРГАНИЗАЦИИ ВХОДА В ПОДПРОГРАММЫ. С ЕГО ПОМОЩЬЮ ФАКТИЧЕСКИЕ ПАРАМЕТРЫ МОЖНО ПЕРЕИМЕНОВАТЬ В ФОРМАЛЬНЫЕ И НАОБОРОТ.

7.3. ОПЕРАТОР RESTORE.

ОПЕРАТОР ВЫПОЛНЯЕТ СЛЕДУЮЩИЕ ФУНКЦИИ:

- ПУСТОЙ ОПЕРАТОР УСТАНОВЛИВАЕТ В НАЧАЛО УКАЗАТЕЛЬ БЛОКА ДАННЫХ;
- RESTORE COSUB ДЕЛАЕТ ПСЕВДОВОЗВРАТ ИЗ ОДНОГО УРОВНЯ ПОДПРОГРАММ;
- RESTORE FOR ДЕЛАЕТ ПСЕВДОВОЗВРАТ ИЗ ОДНОГО УРОВНЯ ЦИКЛОВ.

7.4. ОПЕРАТОР CALL.

ОПЕРАТОР ИСПОЛЬЗУЕТСЯ ДЛЯ ОБРАЩЕНИЯ К ВНЕШНИМ ПОДПРОГРАММАМ. БЛОК ВНЕШНИХ ПОДПРОГРАММ ПРИСТАВЛЯЕТСЯ К ТЕКСТУ ПРОГРАММЫ ИНТЕРПРЕТАТОРА ПЕРЕД ЕГО ЗАПУСКОМ. ВСЕ ПОДПРОГРАММЫ В БЛОКЕ ИМЕЮТ СВОИ ИМЕНА, ОБОЗНАЧЕННЫЕ ОДНОЙ ЛАТИНСКОЙ БУКВОЙ ОТ А ДО Z.

В ТЕЛЕ ОПЕРАТОРА CALL УКАЗЫВАЕТСЯ СНАЧАЛА ИМЯ ПОДПРОГРАММЫ, ЗАТЕМ ВСЕ ПАРАМЕТРЫ, НЕОБХОДИМЫЕ ДЛЯ РАБОТЫ ПОДПРОГРАММЫ.

8. РАБОТА С ТЕКСТОВЫМИ ФАЙЛАМИ.

ТЕКСТОВЫЕ ФАЙЛЫ МОГУТ БЫТЬ СОЗДАНЫ С ПОМОЩЬЮ СПЕЦИАЛЬНЫХ ПРОГРАММ ПОЛЬЗОВАТЕЛЯ ИЛИ САМИМ ИНТЕРПРЕТАТОРОМ.

ПРИ СОЗДАНИИ ТЕКСТОВЫХ ФАЙЛОВ ПОЛЬЗОВАТЕЛЕМ, ОНИ ДОЛЖНЫ БЫТЬ ОРГАНИЗОВАНЫ С УЧЕТОМ СЛЕДУЮЩИХ ТРЕБОВАНИЙ:

- ТЕКСТ ДОЛЖЕН СОСТОЯТЬ ИЗ СТРОК ДЛИНОЙ НЕ БОЛЕЕ 80-ТИ СИМВОЛОВ. СТРОКИ ДЛИНОЙ МЕНЕЕ 80-ТИ СИМВОЛОВ ДОЛЖНЫ ЗАКАНЧИВАТЬСЯ КОДОМ <ПС>-00.10 ;
- СИМВОЛ < \ > (05.12) ПРИМЕНЯТЬ НЕ ДОПУСКАЕТСЯ;
- ФАЙЛ ДОЛЖЕН ЗАКАНЧИВАТЬСЯ КОДАМИ, ОБОЗНАЧАЮЩИМИ КОНЕЦ ЗАПИСИ - 00.00, 05.12;
- ФАЙЛ ЗАПИСЫВАЕТСЯ НА НМЛ ОДНИМ БЛОКОМ КОМАНДОЙ ЭВМ 12.03.

ИНТЕРПРЕТАТОР ПОЗВОЛЯЕТ ВЫПОЛНЯТЬ СЛЕДУЮЩИЕ ДЕЙСТВИЯ С ТЕКСТОВЫМИ ФАЙЛАМИ:

А) ВЫВОД ИНФОРМАЦИИ В ФАЙЛ.

ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ PRINT, С УКАЗАНИЕМ В НЕМ НОМЕРА КАНАЛА ВЫВОДА - #2 . ИНФОРМАЦИЯ ВЫВОДИТСЯ В КОНЕЦ ФАЙЛА АНАЛОГИЧНО ЕЕ ВЫВОДУ НА ДИСПЛЕЙ. СИМВОЛ < > ЗАМЕНЯЕТСЯ ДВОЕТОЧИЕМ.

Б) ВВОД ИНФОРМАЦИИ ИЗ ФАЙЛА.

ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРАМИ INPUT #2,<СПИСОК ВВОДА> И LINPUT #2,<СПИСОК ВВОДА>. КОММЕНТАРИЙ В ОПЕРАТОРЕ INPUT НЕ УКАЗЫВАЕТСЯ. ОПЕРАТОРЫ ВВОДЯТ ИЗ ФАЙЛА ОДНУ СТРОКУ АНАЛОГИЧНО ЕЕ ВВОДУ С ДИСПЛЕЯ. ВВОД СТРОК ПРОИЗВОДИТСЯ В СООТВЕТСТВИИ С УКАЗАТЕЛЕМ ФАЙЛА. ПОСЛЕ ВВОДА ОЧЕРЕДНОЙ СТРОКИ УКАЗАТЕЛЬ ПЕРЕМЕЩАЕТСЯ НА СЛЕДУЮЩУЮ СТРОКУ. УСТАНОВКА УКАЗАТЕЛЯ В НАЧАЛО ФАЙЛА ПРОИЗВОДИТСЯ ОПЕРАТОРОМ RESTORE #2 . АНАЛИЗ СОСТОЯНИЯ УКАЗАТЕЛЯ "КОНЕЦ ФАЙЛА" МОЖНО ПРОИЗВЕСТИ ОПЕРАТОРОМ IF END#2 THEN.

ПРИ ПОСТРОЧНОМ ВВОДЕ ИНФОРМАЦИОННЫХ СТРОК ОПЕРАТОРОМ ВИДА INPUT #2,FL[<СПИСОК ВВОДА>] ИЗ ТЕКСТОВОГО ФАЙЛА МОЖНО ВВЕСТИ СРАЗУ НЕСКОЛЬКО СТРОК: ДО ПЕРВОЙ ПУСТОЙ СТРОКИ ИЛИ ДО КОНЦА ФАЙЛА.

В) ЗАПИСЬ ФАЙЛА НА НМЛ.

ЗАПИСЬ ФАЙЛА НА НМЛ ПРОИЗВОДИТСЯ ОПЕРАТОРОМ

SAVE<КОЛ-ВО ДУБЛЕЙ><ИДЕНТИФИКАТОР>FT

ЗАПИСЬ ТЕКСТОВОГО ФАЙЛА ОДНИМ БЛОКОМ КОМАНДОЙ ЭВМ SAVEX МОЖНО ПРОИЗВЕСТИ ВНЕШНЕЙ ПОДПРОГРАММОЙ (СМ.ПРИЛОЖЕНИЕ) .

Г) ЗАГРУЗКА ФАЙЛА С НМЛ.

ЗАГРУЗКА ФАЙЛА ПРОИЗВОДИТСЯ ОПЕРАТОРОМ LOAD<ИДЕНТИФИКАТОР>. НИКАКИЕ ДРУГИЕ ОБОЗНАЧЕНИЯ НЕ ДОПУСКАЮТСЯ. ЗАГРУЗКА ФАЙЛА ЗАПИСАННОГО КОМАНДОЙ SAVEX ПРОИЗВОДИТСЯ ВНЕШНЕЙ ПОДПРОГРАММОЙ (СМ.ПРИЛОЖЕНИЯ).

Д) ВЫВОД ТЕКСТОВОГО ФАЙЛА НА ДИСПЛЕЙ ИЛИ ПРИНТЕР.

ПРОИЗВОДИТСЯ, СООТВЕТСТВЕННО, ОПЕРАТОРАМИ PRINT#0, FT ИЛИ PRINT#1, FT.

Е) ВЫВОД ТЕКСТА ПРОГРАММЫ ПОЛЬЗОВАТЕЛЯ В ФАЙЛ.

ПРОИЗВОДИТСЯ ОПЕРАТОРОМ LIST С УКАЗАНИЕМ В НЕМ НОМЕРА КАНАЛА - #2.

Ж) ВВОД ТЕКСТА ПРОГРАММЫ ИЗ ФАЙЛА И ЕЕ "КОМПИЛЯЦИЯ".

ПРОИЗВОДИТСЯ ОПЕРАТОРОМ COMPILE. ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА ИНТЕРПРЕТАТОР ДЕЛАЕТ БЕЗУСЛОВНЫЙ ПЕРЕХОД В РЕЖИМ "ДИАЛОГ".

ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА СТРОКИ ИЗ ТЕКСТОВОГО ФАЙЛА, В СООТВЕТСТВИИ С УСТАНОВЛИВАЕМЫМ УКАЗАТЕЛЕМ ФАЙЛА ПО ПОРЯДКУ "КОМПИЛИРУЮТСЯ" И ПЕРЕПИСЫВАЮТСЯ В ПРОГРАММУ ПОЛЬЗОВАТЕЛЯ. ЕСЛИ В КАКОЙ- ЛИБО СТРОКЕ ТЕКСТА В ПРОЦЕССЕ КОМПИЛЯЦИИ ИНТЕРПРЕТАТОР НАХОДИТ ОШИБКУ, ТО ВЫПОЛНЕНИЕ ОПЕРАТОРА ПРЕКРАЩАЕТСЯ И НА ДИСПЛЕЙ ВЫВОДИТСЯ СООБЩЕНИЕ ОБ ОШИБКЕ. УКАЗАТЕЛЬ ФАЙЛА ПРИ ЭТОМ УСТАНОВЛИВАЕТСЯ НА СЛЕДУЮЩУЮ СТРОКУ И, ЕСЛИ В ДАННОМ СЛУЧАЕ, ЕЩЕ РАЗ УКАЗАТЬ ОПЕРАТОР COMPILE, ТО КОМПИЛЯЦИЯ ПРОДОЛЖИТСЯ СО СЛЕДУЮЩЕЙ ПОСЛЕ ОШИБОЧНОЙ СТРОКИ. ВЫПОЛНЕНИЕ ОПЕРАТОРА ПРЕКРАЩАЕТСЯ ПРИ ВВОДЕ ИЗ ТЕКСТОВОГО ФАЙЛА ПУСТОЙ СТРОКИ.

ТЕКСТ ПРОГРАММЫ В ТЕКСТОВОМ ФАЙЛЕ МОЖЕТ СОДЕРЖАТЬ СТРОКИ-КОММЕНТАРИИ. В СТРОКАХ-КОММЕНТАРИЯХ НОМЕР СТРОКИ НЕ УКАЗЫВАЕТСЯ, ТЕКСТ СТРОКИ НАЧИНАЕТСЯ СО СЛОВА REM. СТРОКИ-КОММЕНТАРИИ В ПРОГРАММУ ПОЛЬЗОВАТЕЛЯ НЕ ВВОДЯТСЯ.

З) ИСКЛЮЧЕНИЕ ФАЙЛА ИЗ ОЗУ.

ПРОИЗВОДИТСЯ ОПЕРАТОРОМ CLEAR #2. ОПЕРАТОР CLEARFT ИСКЛЮЧАЕТ ОДНУ СТРОКУ ИЗ ФАЙЛА, ТУ, НА КОТОРУЮ УСТАНОВЛЕН УКАЗАТЕЛЬ ФАЙЛА.

И) РЕДАКЦИЯ ФАЙЛА.

В СОСТАВ ПРОГРАММЫ-ИНТЕРПРЕТАТОРА ВКЛЮЧЕНА ПОДПРОГРАММА - ЭКРАННЫЙ РЕДАКТОР ТЕКСТА. С ЕЕ ПОМОЩЬЮ МОЖНО ОСУЩЕСТВИТЬ ВВОД И РЕДАКЦИЮ ТЕКСТОВОГО ФАЙЛА, ИСПОЛЬЗУЯ КЛАВИАТУРУ ДИСПЛЕЯ.

ВЫЗОВ ПОДПРОГРАММЫ-РЕДАКТОРА ПРОИЗВОДИТСЯ ОПЕРАТОРОМ SCREEN. ПРИ ИСПОЛЬЗОВАНИИ ДИСПЛЕЯ 15ИЭ 200*140-017 ПЕРЕД ВЫПОЛНЕНИЕМ ОПЕРАТОРА НЕОБХОДИМО ОБЯЗАТЕЛЬНО УБРАТЬ РЕЖИМ "РОЛИК".

ПРИ ВХОДЕ В ПОДПРОГРАММУ-РЕДАКТОР ИНТЕРПРЕТАТОР ПРОИЗВОДИТ "РАСПАКОВКУ" ФАЙЛА, В ПРОЦЕССЕ КОТОРОЙ КАЖДАЯ СТРОКА ЗАПИСЫВАЕТСЯ В 80 БАЙТ ОЗУ, ВМЕСТО НЕДОСТАЮЩИХ СИМВОЛОВ ЗАПИСЫВАЮТСЯ ПРОБЕЛЫ. ЕСЛИ ФАЙЛ В "РАСПАКОВАННОМ" ВИДЕ НЕ УМЕЩАЕТСЯ В ОЗУ, ТО ИНТЕРПРЕТАТОР ВЫДАЕТ СООБЩЕНИЕ ОБ ОШИБКЕ.

ПОДПРОГРАММА-РЕДАКТОР СТИРАЕТ ИЗОБРАЖЕНИЕ НА ЭКРАНЕ И, ЕСЛИ ТЕКСТОВЫЙ ФАЙЛ УЖЕ ИМЕЕТСЯ, ВЫВОДИТ ЕГО НА ЭКРАН ДЛЯ РЕДАКЦИИ.

ПОДПРОГРАММА-РЕДАКТОР РЕАЛИЗУЕТ ПОЗИТИВНО-ФОНОВЫЙ РЕЖИМ РАБОТЫ ЭВМ, Т.Е. ОДНОВРЕМЕННО С РЕДАКТИРОВАНИЕМ ТЕКСТОВОГО ФАЙЛА (ПОЗИТИВНАЯ ЗАДАЧА) ПРОИСХОДИТ ВЫПОЛНЕНИЕ ПРОГРАММЫ ПОЛЬЗОВАТЕЛЯ (ФОНОВАЯ ЗАДАЧА).

ПРОИЗВЕСТИ ЗАПУСК ОБЕИХ ЗАДАЧ МОЖНО НАБРАВ, НАПРИМЕР, СТРОКУ

SCREEN:GOTO<НОМЕР СТРОКИ ПРОГРАММЫ>

ЛИБО (ЕСЛИ ВЫПОЛНЕНИЕ ПРОГРАММЫ БЫЛО ПРЕРВАНО ДО ЭТОГО ОПЕРАТОРОМ STOP)

SCREEN:START

РЕДАКТИРОВАНИЕ ТЕКСТА ПРЕРЫВАЕТСЯ ПРИ ВЫПОЛНЕНИИ ФОНОВОЙ ЗАДАЧЕЙ ОПЕРАЦИЙ С НМЛ.

ВЫПОЛНЕНИЕ ФОНОВОЙ ЗАДАЧИ ПРЕКРАЩАЕТСЯ В СЛЕДУЮЩИХ СЛУЧАЯХ:

- ПРИ ОКОНЧАНИИ ЗАДАЧИ И ВЫХОДЕ В РЕЖИМ "ДИАЛОГ";
- ПРИ ВЫВОДЕ В ЗАДАЧЕ ИНФОРМАЦИИ НА ДИСПЛЕЙ;
- ПРИ ОБРАЩЕНИИ К РЕДАКТИРУЕМОМУ ТЕКСТОВОМУ ФАЙЛУ;
- ПРИ ВЫВОДЕ НА ПРИНТЕР ОЧЕРЕДНОЙ СТРАНИЦЫ ТЕКСТА.

ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАВИШ РЕДАКТИРОВАНИЯ ЗАВИСИТ ОТ КОНСТРУКЦИИ ДИСПЛЕЯ И УКАЗАНО В ПРИЛОЖЕНИИ.

ВЫХОД ИЗ ПОДПРОГРАММЫ-РЕДАКТОРА ОСУЩЕСТВЛЯЕТСЯ ПОСЛЕ НАЖАТИЯ КЛАВИШИ СУ/А (УПР/А), ПОСЛЕ ЧЕГО ИНТЕРПРЕТАТОР ПРОИЗВОДИТ ОБРАТНУЮ "УПАКОВКУ" ФАЙЛА И, ЕСЛИ ВЫПОЛНЕНИЕ ФОНОВОЙ ЗАДАЧИ ЗАКОНЧЕНО, ТО ПЕРЕХОДИТ В РЕЖИМ "ДИАЛОГ", ЕСЛИ НЕТ, ТО ПРОДОЛЖАЕТ ЕЕ ВЫПОЛНЕНИЕ В ПОЗИТИВНОМ РЕЖИМЕ.

9. ИСПОЛЬЗОВАНИЕ РАСШИРЕННОЙ ПАМЯТИ.

ПРИ ИСПОЛЬЗОВАНИИ ЭВМ С ОЗУ РАЗМЕРОМ 128 К-БАЙТ, ИНТЕРПРЕТАТОР ПОЗВОЛЯЕТ ИСПОЛЬЗОВАТЬ РАСШИРЕННУЮ ПАМЯТЬ ПУТЕМ ПЕРЕКЛЮЧЕНИЯ ОДНОЙ СМЕННОЙ СТРАНИЦЫ ВЕЛИЧИНОЙ 8 К-БАЙТ. ОСНОВНЫЕ СРЕДСТВА ИНТЕРПРЕТАТОРА ПОЗВОЛЯЮТ ЗАГРУЖАТЬ РАСШИРЕННУЮ ПАМЯТЬ ТЕКСТОВЫМИ ФАЙЛАМИ. ДЛЯ РАЗМЕЩЕНИЯ В ЗОНЕ РАСШИРЕННОЙ ПАМЯТИ ДРУГИХ ТИПОВ ДАННЫХ НЕОБХОДИМО ИСПОЛЬЗОВАТЬ ВНЕШНИЕ ПОДПРОГРАММЫ.

РАСШИРЕННАЯ ЗОНА ПАМЯТИ ИНТЕРПРЕТАТОРОМ РАЗДЕЛЯЕТСЯ НА 12 СМЕННЫХ СТРАНИЦ РАЗМЕРОМ ПО 8 К-БАЙТ. КАЖДОЙ СТРАНИЦЕ СООТВЕТСТВУЕТ СВОЙ ЛОГИЧЕСКИЙ НОМЕР КАНАЛА ВВОДА-ВЫВОДА, НАЗНАЧАЕМЫЙ В ДИАПАЗОНЕ ОТ 3 ДО 14.

НА СМЕННЫХ СТРАНИЦАХ МОЖНО ВЫПОЛНЯТЬ ВСЕ ОПЕРАЦИИ С ТЕКСТОВЫМИ ФАЙЛАМИ, ПУТЕМ УКАЗАНИЯ В СООТВЕТСТВУЮЩИХ ОПЕРАТОРАХ НОМЕРА КАНАЛА ВЫВОДА ОТ #3 ДО #14. НЕКОТОРЫЕ ОСОБЕННОСТИ РАБОТЫ ИМЕЮТ СЛЕДУЮЩИЕ ОПЕРАТОРЫ:

- В ОПЕРАТОРЕ SCREEN УКАЗЫВАЕТСЯ НОМЕР КАНАЛА ВЫВОДА. ПО УМОЛЧАНИЮ НОМЕР КАНАЛА - #2;
- ДЛЯ РАСПЕЧАТКИ ФАЙЛА, В ОПЕРАТОРЕ PRINT ПОСЛЕ СЛОВА FT НЕОБХОДИМО УКАЗАТЬ ИНДЕКС, ОЗНАЧАЮЩИЙ НОМЕР КАНАЛА. ПО УМОЛЧАНИЮ ИНДЕКС РАВЕН ДВУМ;
- ДЛЯ ОЧИСТКИ СМЕННОЙ СТРАНИЦЫ НЕОБХОДИМО В ОПЕРАТОРЕ CLEAR УКАЗАТЬ НОМЕР КАНАЛА ДЛЯ ОЧИСТКИ ВСЕЙ СТРАНИЦЫ ИЛИ ОБОЗНАЧЕНИЕ FT (<ИНДЕКСНОЕ ВЫРАЖЕНИЕ>) ДЛЯ ИСКЛЮЧЕНИЯ ОДНОЙ СТРОКИ;
- ДЛЯ ЗАПИСИ ИНФОРМАЦИИ СТРАНИЦЫ НА НМЛ ИСПОЛЬЗУЕТСЯ ОПЕРАТОР ВИДА

SAVE<ИДЕНТИФИКАТОР><НОМЕР КАНАЛА>

- ЗАГРУЗКА ИНФОРМАЦИИ С НМЛ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ ВИДА

LOAD<ИДЕНТИФИКАТОР><НОМЕР КАНАЛА>

ПРИ ЗАГРУЗКЕ ОПЕРАТОРОМ LOAD СТИРАЕТСЯ ПРЕДЫДУЩЕЕ СОДЕРЖИМОЕ НА УКАЗАННОЙ СТРАНИЦЕ.

ПРИ ЗАПИСИ ТЕКСТОВЫХ ФАЙЛОВ НА СМЕННЫЕ СТРАНИЦЫ, СЛЕДУЕТ ПОМНИТЬ, ЧТО МАКСИМАЛЬНЫЙ РАЗМЕР ФАЙЛА - 8 К-БАЙТ, А МАКСИМАЛЬНЫЙ РЕДАКТИРУЕМЫЙ РАЗМЕР ТЕКСТА - 102 СТРОКИ.

РАЗМЕР СВОБОДНОЙ ОБЛАСТИ ОЗУ НА ДАННОЙ СТРАНИЦЕ ОПРЕДЕЛЯЕТСЯ ФУНКЦИЕЙ SIZE В АРГУМЕНТЕ КОТОРОЙ ИНДЕКСНЫМ ВЫРАЖЕНИЕМ УКАЗЫВАЕТСЯ НОМЕР СТРАНИЦЫ.

10. ПОРЯДОК РАБОТЫ С ИНТЕРПРЕТАТОРОМ.

ЗАГРУЗКА И ЗАПУСК ИНТЕРПРЕТАТОРА ОСУЩЕСТВЛЯЕТСЯ В СЛЕДУЮЩЕЙ ПОСЛЕДОВАТЕЛЬНОСТИ:

- ВКЛЮЧАЕТСЯ В СЕТЬ ДИСПЛЕЙ И ЭВМ, ПРИ НЕОБХОДИМОСТИ ПРИНТЕР;
- ПРОГРАММА-ИНТЕРПРЕТАТОР ЗАГРУЖАЕТСЯ В ОЗУ С АДРЕСА 0;
- ПРОИЗВОДИТСЯ ЗАПУСК ИНТЕРПРЕТАТОРА КЛАВИШАМИ [C] И [S].

ПОСЛЕ ЗАПУСКА ИНТЕРПРЕТАТОРА НА ДИСПЛЕЙ ВЫВОДИТСЯ ИМЯ ИНТЕРПРЕТАТОРА И ДВОЕТОЧИЕ, ПОСЛЕ КОТОРОГО НЕОБХОДИМО НАБРАТЬ СТРОКУ ПРОГРАММЫ.

РАБОТУ ИНТЕРПРЕТАТОРА В ЛЮБОЙ МОМЕНТ МОЖНО ПРЕРВАТЬ НАЖАТИЕМ НА ЭВМ КЛАВИШ [Ш] ИЛИ [ШН], ПОСЛЕ ЧЕГО ПРОВЕРИТЬ ЕГО СОХРАННОСТЬ КОНТРОЛЬНОЙ СУММОЙ И ОСУЩЕСТВИТЬ ПОВТОРНЫЙ ЕГО ЗАПУСК. ПРИ ПОВТОРНОМ ЗАПУСКЕ КЛАВИШАМИ [C], [S] СТИРАЕТСЯ ВСЯ ИНФОРМАЦИЯ ПОЛЬЗОВАТЕЛЯ И ИНТЕРПРЕТАТОР УСТАНОВЛИВАЕТСЯ В ИСХОДНОЕ СОСТОЯНИЕ. ДЛЯ СОХРАНЕНИЯ ИНФОРМАЦИИ МОЖНО ОСУЩЕСТВИТЬ ЗАПУСК КЛАВИШАМИ [>], [M], ВЫЗЫВАЮЩИМИ ПЕРЕХОД ИНТЕРПРЕТАТОРА В РЕЖИМ "ДИАЛОГ", ОДНАКО, В ЭТОМ СЛУЧАЕ ВОЗМОЖНА ПОТЕРЯ РАБОТСПОСОБНОСТИ ИНТЕРПРЕТАТОРА.

РАБОТУ ИНТЕРПРЕТАТОРА МОЖНО ПОДРАЗДЕЛИТЬ НА ДВА РЕЖИМА: РЕЖИМ "ДИАЛОГ" И ПРОГРАММНЫЙ.

ПРИЗНАКОМ РЕЖИМА "ДИАЛОГ" ЯВЛЯЕТСЯ ДВОЕТОЧИЕ, ВЫВОДИМОЕ НА ЭКРАН. ПОСЛЕ ДВОЕТОЧИЯ НАБИРАЕТСЯ СТРОКА ПРОГРАММЫ. ВВОД СТРОКИ ЗАКАНЧИВАЕТСЯ СИМВОЛОМ <ПС>. МАКСИМАЛЬНАЯ ДЛИНА СТРОКИ ПРОГРАММЫ - 80 СИМВОЛОВ, ВКЛЮЧАЯ СИМВОЛ <ПС>. ЕСЛИ В НАЧАЛЕ СТРОКИ СТОИТ НОМЕР, ТО ОНА ВВОДИТСЯ В ПРОГРАММУ, ЕСЛИ НОМЕРА В НАЧАЛЕ СТРОКИ НЕТ, ТО ОНА ВЫПОЛНЯЕТСЯ.

ПОСЛЕ ПОЛНОГО НАБОРА СТРОКИ ПРОИЗВОДИТСЯ ЕЕ "КОМПИЛЯЦИЯ", В ПРОЦЕССЕ КОТОРОЙ НОМЕР СТРОКИ ИЗ СИМВОЛЬНОГО ВИДА ПРЕОБРАЗУЕТСЯ В ШЕСТНАДЦАТИРИЧНЫЙ, А ВСЕ СЛУЖЕБНЫЕ СЛОВА ЗАМЕНЯЮТСЯ СПЕЦИАЛЬНЫМИ ОДНОБАЙТНЫМИ КОДАМИ. ЕСЛИ В ТЕКСТЕ СТРОКИ ПРОГРАММЫ В НЕДОПУСТИМОМ МЕСТЕ (Т.Е. НЕ В ТЕКСТОВЫХ КОНСТАНТАХ, И НЕ ПОСЛЕ СЛОВ DATA И REM) ИМЕЮТСЯ РУССКИЕ ИЛИ СТРОЧНЫЕ БУКВЫ, ТО ИНТЕРПРЕТАТОРОМ ОНИ ЗАМЕНЯЮТСЯ НА СООТВЕТСТВУЮЩИЕ ИМ ПРОПИСНЫЕ ЛАТИНСКИЕ. НАПРИМЕР СТРОКУ <PRINT A,B,C> МОЖНО НАБРАТЬ <ПРИНТ А,Б,Ц>.

СТРОКИ ВВОДЯТСЯ В ПРОГРАММУ В "КОМПИЛИРОВАННОМ" ВИДЕ. ЕСЛИ ДО ЭТОГО В ПРОГРАММЕ УЖЕ БЫЛА СТРОКА С ЭТИМ НОМЕРОМ, ТО ОНА ЗАМЕНЯЕТСЯ НА НОВУЮ.

ПЕРЕХОД ИЗ РЕЖИМА "ДИАЛОГ" В ПРОГРАММНЫЙ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРАМИ RUN, GOTO ИЛИ ON. ОТДЕЛЬНУЮ ПОДПРОГРАММУ МОЖНО ВЫПОЛНИТЬ ОПЕРАТОРОМ GOSUB.

ПЕРЕХОД ИЗ ПРОГРАММНОГО РЕЖИМА В "ДИАЛОГ" ОСУЩЕСТВЛЯЕТСЯ ПОСЛЕ ОКОНЧАНИЯ ПРОГРАММЫ, ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРОВ STOP ИЛИ END, А ТАКЖЕ ПРИ ОБНАРУЖЕНИИ ИНТЕРПРЕТАТОРОМ ОШИБКИ. ПРИ ПЕРЕХОДЕ В РЕЖИМ "ДИАЛОГ" ИНТЕРПРЕТАТОР ВЫПОЛНЯЕТ СЛЕДУЮЩИЕ ОПЕРАЦИИ:

- УСТАНОВЛИВАЕТ В НУЛЬ СЧЕТЧИК СТРОК ПРИНТЕРА;
- УСТАНОВЛИВАЕТ ФОРМАТ ПЕЧАТИ ЧИСЕЛ - !A!;
- УСТАНОВЛИВАЕТ НОМЕР КАНАЛА ВЫВОДА - 0;
- УСТАНОВЛИВАЕТ В ИСХОДНОЕ СОСТОЯНИЕ УКАЗАТЕЛИ СТЕКОВ ЦИКЛОВ И ПОДПРОГРАММ.

В РЕЖИМЕ "ДИАЛОГ" НЕ ДОПУСКАЕТСЯ ПРИМЕНЕНИЕ ОПЕРАТОРА INPUT.

С ЦЕЛЮ СОХРАННОСТИ ДАННЫХ И САМОЙ ПРОГРАММЫ-ИНТЕРПРЕТАТОРА ИНТЕРПРЕТАТОРОМ, ПО ВОЗМОЖНОСТИ, БЛОКИРУЮТСЯ ОШИБКИ ПОЛЬЗОВАТЕЛЯ. ПРИ ОБНАРУЖЕНИИ ОШИБКИ НА ДИСПЛЕЙ ВЫДАЕТСЯ СООБЩЕНИЕ ВИДА

ОШИБКА <K> В СТРОКЕ <N>

ГДЕ: K - ШИФР ОШИБКИ, N - НОМЕР СТРОКИ ПРОГРАММЫ ПРИ ВЫПОЛНЕНИИ КОТОРОЙ ОНА ОБНАРУЖЕНА. ВЕРОЯТНЫЕ ПРИЧИНЫ ОШИБОК, В ЗАВИСИМОСТИ ОТ ИХ ШИФРОВ, УКАЗАНЫ В ПРИЛОЖЕНИИ.

ПОСЛЕ СООБЩЕНИЯ ОБ ОШИБКЕ ИНТЕРПРЕТАТОР ПЕРЕХОДИТ В РЕЖИМ "ДИАЛОГ".

11. ДОПОЛНИТЕЛЬНЫЕ СВЕДЕНИЯ О КОНСТРУКЦИИ ИНТЕРПРЕТАТОРА.

11.1. ОПИСАНИЕ СТРУКТУРЫ ИНТЕРПРЕТАТОРА.

ИНТЕРПРЕТАТОР ИМЕЕТ БЛОЧНУЮ СТРУКТУРУ И СОСТОИТ ИЗ ОТДЕЛЬНЫХ БЛОКОВ, РАСПОЛОЖЕНИЕ КОТОРЫХ ПОКАЗАНО НА СЛЕДУЮЩЕМ РИСУНКЕ:

```
-----  
!           БЛОК ОБРАБОТКИ ПРЕРЫВАНИЙ           !  
!-----!  
!           БЛОК ЗАПУСКА                         !  
!-----!  
! М 04 08 !           ГЛАВНЫЙ ПРОГРАММНЫЙ МОДУЛЬ           !  
!-----!  
! М 00 00 !           ПОДПРОГРАММА ОБРАБОТКИ ОШИБОК           !  
!-----!  
! М 00 00 !СЕРВИСНАЯ ПП (ОПРЕДЕЛЕНИЕ АДРЕСОВ ПОДПРОГРАММ) !  
!-----!  
!           БЛОК СЛУЖЕБНЫХ СЛОВ И СООБЩЕНИЙ           !  
!-----!  
!           БЛОК ОСНОВНЫХ ПОДПРОГРАММ ИНТЕРПРЕТАТОРА           !  
!           (НОМЕРА 1..179)                             !  
!-----!  
!           РЕЗЕРВ ДЛЯ ПОДПРОГРАММ (180..222)           !  
!-----!  
!           ВНЕШНИЕ ПОДПРОГРАММЫ (223..248)           !  
!-----!  
!           КОД КОНЦА ПРОГРАММЫ (13 13, 05 14, 05 12)           !  
!-----!
```

БЛОК ОБРАБОТКИ ПРЕРЫВАНИЙ ОБСЛУЖИВАЕТ ПРЕРЫВАНИЯ, ПОСТУПАЮЩИЕ С ДИСПЛЕЯ. ПРЕРЫВАНИЯ В ИНТЕРПРЕТАТОРЕ ИСПОЛЬЗУЮТСЯ ДЛЯ ОРГАНИЗАЦИИ ОСТАНОВОВ ПРОГРАММЫ И ДЛЯ ОРГАНИЗАЦИИ СОВМЕСТНОЙ РАБОТЫ РЕДАКТОРА И ФОНОВОЙ ЗАДАЧИ. ПРЕРЫВАНИЯ ВОЗНИКАЮТ ПО СИГНАЛУ PR8 (УРОВЕНЬ 2). ОСТАЛЬНЫЕ УРОВНИ ПРЕРЫВАНИЙ МОГУТ БЫТЬ ИСПОЛЬЗОВАНЫ ПОЛЬЗОВАТЕЛЕМ. ДЛЯ ЭТОЙ ЦЕЛИ В БЛОКЕ ОБРАБОТКИ ПРЕРЫВАНИЙ ИМЕЮТСЯ ЗАРЕЗЕРВИРОВАННЫЕ ШАГИ ПРОГРАММЫ.

БЛОК ЗАПУСКА ПРЕДНАЗНАЧЕН ДЛЯ УСТАНОВКИ ИНТЕРПРЕТАТОРА В ИСХОДНОЕ СОСТОЯНИЕ. БЛОК ОСУЩЕСТВЛЯЕТ ПОИСК ПОДПРОГРАММ ИНТЕРПРЕТАТОРА И ЗАПОЛНЕНИЕ ТАБЛИЦЫ АДРЕСОВ ПОДПРОГРАММ, РАСПРЕДЕЛЕНИЕ ОЗУ И УСТАНОВКУ СЛУЖЕБНЫХ РЕГИСТРОВ В СООТВЕТСТВИИ С ПРИНЯТЫМ ИСХОДНЫМ (НАЧАЛЬНЫМ) СОСТОЯНИЕМ ИНТЕРПРЕТАТОРА. БЛОК СОХРАНЯЕТСЯ ПОСЛЕ ЗАПУСКА ИНТЕРПРЕТАТОРА, ЭТО ОЗНАЧАЕТ, ЧТО В ЛЮБОЙ МОМЕНТ РАБОТА ИНТЕРПРЕТАТОРА МОЖЕТ БЫТЬ ОСТАНОВЛЕНА И ОСУЩЕСТВЛЕН ПОВТОРНЫЙ ЕГО ЗАПУСК.

СОСТАВНЫМИ ЧАСТЯМИ ПРОГРАММЫ-ИНТЕРПРЕТАТОРА ЯВЛЯЮТСЯ - ГЛАВНЫЙ ПРОГРАММНЫЙ МОДУЛЬ И БЛОК ОСНОВНЫХ ПОДПРОГРАММ. К БЛОКУ ОСНОВНЫХ ПОДПРОГРАММ МОЖЕТ БЫТЬ ДОБАВЛЕН ДОПОЛНИТЕЛЬНЫЙ БЛОК ВНЕШНИХ ПОДПРОГРАММ, ЯВЛЯЮЩИЙСЯ ЛОГИЧЕСКИМ ПРОДОЛЖЕНИЕМ ИНТЕРПРЕТАТОРА.

ВСЕ ОСНОВНЫЕ И ВНЕШНИЕ ПОДПРОГРАММЫ ИМЕЮТ ЛОГИЧЕСКИЕ НОМЕРА (ОТ 1 ДО 248), КОТОРЫЕ МОГУТ БЫТЬ ИСПОЛЬЗОВАНЫ ДЛЯ ПОИСКА ИХ АДРЕСОВ. ЛОГИЧЕСКИЕ НОМЕРА ИСПОЛЬЗУЮТСЯ, ТАКЖЕ, ПРИ ДИАГНОСТИКЕ ОШИБОК.

ВСЕ ПОДПРОГРАММЫ РАСПОЛОЖЕНЫ В ИНТЕРПРЕТАТОРЕ ПО ПОРЯДКУ, В СООТВЕТСТВИИ СО СВОИМИ НОМЕРАМИ. ПЕРЕД КАЖДОЙ ПОДПРОГРАММОЙ ОБЯЗАТЕЛЬНО ДОЛЖЕН СТОЯТЬ РАЗДЕЛИТЕЛЬНЫЙ ЗНАК - КОД 13.13. ПО РАЗДЕЛИТЕЛЬНЫМ ЗНАКАМ БЛОК ЗАПУСКА ОПРЕДЕЛЯЕТ АДРЕСА ПОДПРОГРАММ И ЗАНОСИТ ИХ В ТАБЛИЦУ. ИСКЛЮЧАТЬ КОДЫ 13.13 ИЗ ИНТЕРПРЕТАТОРА НЕ ДОПУСКАЕТСЯ. НЕ ДОПУСКАЕТСЯ, ТАКЖЕ, ИСПОЛЬЗОВАНИЕ ЭТИХ КОДОВ В ТЕКСТЕ ПОДПРОГРАММ.

БЛОЧНАЯ КОНСТРУКЦИЯ ИНТЕРПРЕТАТОРА С РАЗДЕЛИТЕЛЬНЫМИ КОДАМИ ПРЕДОСТАВЛЯЕТ ШИРОКИЕ ВОЗМОЖНОСТИ ДЛЯ ВНЕСЕНИЯ В НЕГО ИЗМЕНЕНИЙ. В ПОДПРОГРАММЫ ИНТЕРПРЕТАТОРА ДОПУСКАЕТСЯ ВСТАВЛЯТЬ ДОПОЛНИТЕЛЬНЫЕ ШАГИ, ИСКЛЮЧАТЬ НЕНУЖНЫЕ. В ПРОГРАММУ-ИНТЕРПРЕТАТОР МОЖНО ВСТАВЛЯТЬ НА ЗАРЕЗЕРВИРОВАННЫЕ МЕСТА НОВЫЕ ПОДПРОГРАММЫ, ИСКЛЮЧАТЬ ПОДПРОГРАММЫ, НЕ ИСПОЛЬЗУЕМЫЕ В КОНКРЕТНЫХ ЗАДАЧАХ. ВСЕ ЭТИ ОПЕРАЦИИ ПРОИЗВОДЯТСЯ БЕЗ ПЕРЕСЧЕТА АДРЕСОВ ПОДПРОГРАММ, ТАК КАК ОНИ ОПРЕДЕЛЯЮТСЯ САМИМ ИНТЕРПРЕТАТОРОМ ПРИ ЕГО ЗАПУСКЕ.

ПРИ ОТЛАДКЕ ИНТЕРПРЕТАТОРА ВО ВРЕМЯ ВНЕСЕНИЯ В НЕГО ИЗМЕНЕНИЙ УДОБНО ПОЛЬЗОВАТЬСЯ СЕРВИСНОЙ ПОДПРОГРАММОЙ 00.00, ОПРЕДЕЛЯЮЩЕЙ АДРЕС ПОДПРОГРАММЫ ПО ЕЕ НОМЕРУ. ДЛЯ ЭТОГО В РЕГИСТР X ЭВМ ПОСЛЕ ЗАГРУЗКИ ИНТЕРПРЕТАТОРА ЗАНОСИТСЯ НОМЕР ТРЕБУЕМОЙ ПОДПРОГРАММЫ И, ДАЛЕЕ, НА КЛАВИАТУРЕ ПРЯМОГО КОДИРОВАНИЯ НАБИРАЕТСЯ КОД СЕРВИСНОЙ ПОДПРОГРАММЫ 00.00. РЕЗУЛЬТАТ - АДРЕС ПОДПРОГРАММЫ БУДЕТ ОТОБРАЖЕН В РЕГИСТРЕ X.

11.2. ПРОСТЕЙШИЕ ВОЗМОЖНОСТИ КОМПОНОВКИ ИНТЕРПРЕТАТОРА.

НЕПРИЯТНЫМ, НО, МОЖНО НАДЕЯТЬСЯ, ВРЕМЕННЫМ НЕДОСТАТКОМ ЯВЛЯЕТСЯ ОТСУТСТВИЕ В ИНТЕРПРЕТАТОРЕ БЛОКА КОМПОНОВКИ, КОТОРЫЙ ПОЗВОЛИЛ БЫ ПОЛЬЗОВАТЕЛЮ БЫСТРО СОЗДАВАТЬ НЕОБХОДИМЫЕ ВЕРСИИ ИНТЕРПРЕТАТОРА, СОРИЕНТИРОВАННЫЕ НА КОНКРЕТНЫЙ НАБОР ВНЕШНИХ УСТРОЙСТВ И КОНКРЕТНЫЙ ТИП РЕШАЕМОЙ ЗАДАЧИ. ОДНАКО, ПРИ НАЛИЧИИ НЕОБХОДИМОЙ КВАЛИФИКАЦИИ ПОЛЬЗОВАТЕЛЬ МОЖЕТ ВЫПОЛНИТЬ НЕКОТОРЫЕ ОПЕРАЦИИ ПО КОМПОНОВКЕ, ИСПОЛЬЗУЯ СРЕДСТВА РЕДАКТИРОВАНИЯ, ИМЕЮЩИЕСЯ В ЭВМ.

С ЦЕЛЬЮ СОВМЕСТИМОСТИ ИНТЕРПРЕТАТОРА С СОСТАВОМ ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА РАЗРАБОТАНО НЕСКОЛЬКО ВЕРСИЙ ИНТЕРПРЕТАТОРА, ОДНАКО НА ПРАКТИКЕ МОЖЕТ ПОЯВИТЬСЯ НЕОБХОДИМОСТЬ СМЕЖНЫХ ВАРИАНТОВ. НИЖЕ В ТАБЛИЦЕ ПРЕДСТАВЛЕНЫ ТИПЫ УСТРОЙСТВ И НОМЕРА ПОДПРОГРАММ, ОБЕСПЕЧИВАЮЩИЕ РАБОТУ ИНТЕРПРЕТАТОРА С ЭТИМИ УСТРОЙСТВАМИ. ПРИ

НЕОБХОДИМОСТИ ПОЛЬЗОВАТЕЛЬ МОЖЕТ ПЕРЕСТАВИТЬ ПОДПРОГРАММЫ ИЗ ОДНОГО ВАРИАНТА ИНТЕРПРЕТАТОРА В ДРУГОЙ, СОЗДАВ ТЕМ САМЫМ НОВУЮ ВЕРСИЮ. НАПРИМЕР, ЕСЛИ В КОМПЛЕКСЕ ЭВМ ИМЕЕТСЯ ДИСПЛЕЙ 15-ИЭ-00-013 И ЭПУ "КОНСУЛ", ТО ДЛЯ РАБОТЫ С ДАННЫМ КОМПЛЕКСОМ НЕОБХОДИМО ПОДПРОГРАММУ НОМЕР 137 ИЗ ВАРИАНТА 32-ВК ПЕРЕСТАВИТЬ НА СВОЕ ЖЕ МЕСТО В ВАРИАНТ 32-ФР.

УСТРОЙСТВО	! ВЕРСИЯ!	НОМЕРА ПОДПРОГРАММ
ЭВМ С ОЗУ 128 К-БАЙТ	128-ФР	21, 24, 40, 48, 53, 95, 138 153, 169..173, 176
ЭВМ С ОЗУ 32 К-БАЙТ	32-ВК	- " -
ЭВМ СО СКОРОСТЬЮ 1 МЛН.Т/СЕК	128-ФР	39, 42, 161
ЭВМ СО СКОРОСТЬЮ 500 ТЫС.Т/СЕК	32-ВК	- " -
АЦПУ "РОБОТРОН"	128-ФР	137
ТЕРМОПЕЧАТАЮЩЕЕ УСТРОЙСТВО	128-ФР	-"-
ЭПУ "КОНСУЛ 260.1"	32-ВК	-"-
ПЕЧ.УСТРОЙСТВО УВВПЧ-30-004	128-ФМ	-"-

С ЦЕЛЬЮ ЭКОНОМИИ ОЗУ ПОЛЬЗОВАТЕЛЮ МОЖНО ИСКЛЮЧИТЬ ИЗ ИНТЕРПРЕТАТОРА НЕИСПОЛЬЗУЕМЫЕ ПОДПРОГРАММЫ. ДЛЯ ЭТОГО ОПРЕДЕЛЯЕТСЯ АДРЕС ПОДПРОГРАММЫ, И НАЧИНАЯ С НАЙДЕННОГО АДРЕСА ИЗ ТЕКСТА ИНТЕРПРЕТАТОРА ИСКЛЮЧАЮТСЯ ШАГИ ДО СЛЕДУЮЩЕГО КОДА 13.13, КОТОРЫЙ ИСКЛЮЧАТЬ НЕ ДОПУСКАЕТСЯ. РЕКОМЕНДУЕТСЯ НА МЕСТЕ ПОДПРОГРАММЫ ОСТАВЛЯТЬ ОДИН ШАГ, НА КОТОРЫЙ НЕОБХОДИМО ЗАПИСАТЬ КОД 00.10 (КОД ПОДПРОГРАММЫ ОБРАБОТКИ ОШИБКИ), БЛОКИРУЮЩИЙ ВХОД В ПОДПРОГРАММУ. В ЭТОМ СЛУЧАЕ ПРИ ОШИБОЧНОМ ВХОДЕ В ОТСУТСТВУЮЩУЮ ПОДПРОГРАММУ ИНТЕРПРЕТАТОР ВЫДАСТ СООБЩЕНИЕ ОБ ОШИБКЕ. ШИФР ОШИБКИ - НОМЕР ОТСУТСТВУЮЩЕЙ ПОДПРОГРАММЫ.

НИЖЕ В ТАБЛИЦЕ УКАЗАНЫ ОТДЕЛЬНЫЕ ФУНКЦИИ И ОПЕРАТОРЫ ИНТЕРПРЕТАТОРА, А ТАКЖЕ, НОМЕРА ОБСЛУЖИВАЮЩИХ ИХ ПОДПРОГРАММ. ЕСЛИ НОМЕРА ПОДПРОГРАММ ЗАКЛЮЧЕНЫ В КВАДРАТНЫЕ СКОБКИ, ТО ИХ БЛОКИРОВАТЬ НЕ ОБЯЗАТЕЛЬНО.

НАИМЕНОВАНИЕ ФУНКЦИЙ, ОПЕРАТОРОВ	!НОМЕРА ПОДПРОГРАММ !	! ПРИМЕРНОЕ ! КОЛ-ВО БАЙТ
CHANGE	105	141
COM	11, 78, 109, 150, [90]	411
COMPILE	104	31
DIM	110	127
INPUT - ПОСТРОЧНЫЙ ВВОД	62	138
LINPUT	120	40
ON	127	42
PRINT FT	138	57
PRINT USING	159, [51]	283
REN	168	99
SCREEN	124, [21, 82, 97, 98, 140, 142, 144, 160, 162]	881
SEG	75	89
SET	118	35
SIZE	152	56
VAL	151	26

ИЗ ТЕКСТА ИНТЕРПРЕТАТОРА МОЖНО, ТАКЖЕ, ИСКЛЮЧИТЬ БЛОК ВНЕШНИХ ПОДПРОГРАММ. ДЛЯ ЭТОГО ВМЕСТО ПП180 ЗАПИСЫВАЕТСЯ КОД КОНЦА ЗАПИСИ - 05.14, 05.12, А НА КЛЮЧЕВОМ ШАГЕ, НОМЕР КОТОРОГО УКАЗАН В ПРИЛОЖЕНИИ ЗАПИСЫВАЕТСЯ КОД 00.10.

11.3. РАСПРЕДЕЛЕНИЕ ОЗУ.

РАСПРЕДЕЛЕНИЕ 32 К-БАЙТ ОСНОВНОЙ ПАМЯТИ ПОКАЗАНО НА СЛЕДУЮЩЕМ РИСУНКЕ

```

- - - - -
! АДРЕС ! НАИМЕНОВАНИЕ ЗОНЫ ОЗУ !
!- - - - -!- - - - -!
! 00.00.00.00 ! ТЕКСТ ПРОГРАММЫ-ИНТЕРПРЕТАТОРА !
!- - - - -!- - - - -!
! (07.12.03.04) ! ТЕКСТ ПРОГРАММЫ ПОЛЬЗОВАТЕЛЯ !
!- - - - -!- - - - -!
! (07.12.03.08) ! ТЕКСТОВЫЙ ФАЙЛ (КАНАЛ #2) !
!- - - - -!- - - - -!
! (07.12.03.12) ! СВОБОДНАЯ ОБЛАСТЬ ПОЛЬЗОВАТЕЛЯ !
!- - - - -!- - - - -!
! (T1) ! ПЕРЕМЕННЫЕ (ОБЛАСТЬ DIM) !
!- - - - -!- - - - -!
! (T2) ! ФАЙЛЫ FL (ОБЛАСТЬ COM) !
!- - - - -!- - - - -!
! 07.08.00.00 !
! . . . ! СЛУЖЕБНАЯ ЗОНА ОЗУ !
! 07.15.15.15 !
- - - - -

```

В ОБЛАСТИ DIM РАЗМЕЩАЕТСЯ ТРИ ТИПА ПЕРЕМЕННЫХ. ОБЛАСТЬ ОЗУ КАЖДОЙ ПЕРЕМЕННОЙ ИЛИ МАССИВА ПЕРЕМЕННЫХ НАЧИНАЕТСЯ С ИДЕНТИФИКАТОРА, СОСТОЯЩЕГО ИЗ 4-Х БАЙТ, СОДЕРЖАЩИХ СЛЕДУЮЩУЮ ИНФОРМАЦИЮ

```

- - - - -
! S ! 0 S - КОД БУКВЫ ИМЕНИ ПЕРЕМЕННОЙ;
!- - - - -! K - КОД ТИПА ПЕРЕМЕННОЙ: 0 - ВЕЩЕСТ-
! K ! I ! 1 ВЕННЫЙ, 4 - ЦЕЛЬНЫЙ, 8 - ТЕКСТОВЫЙ;
!- - - - -! I - ЦИФРА В ИМЕНИ ПЕРЕМЕННОЙ, ЕСЛИ
! N ! 2 ЦИФРЫ НЕТ, ТО I=10;
!- - - - -! N - МАКСИМАЛЬНЫЙ ИНДЕКС СТОЛБЦА;
! M ! 3 M - МАКСИМАЛЬНЫЙ ИНДЕКС СТРОКИ.
- - - - -

```

ПРОСТЫЕ ПЕРЕМЕННЫЕ (БЕЗ ИНДЕКСА) ПРЕДСТАВЛЯЮТСЯ КАК МАССИВЫ ПЕРЕМЕННЫХ С МАКСИМАЛЬНЫМИ ИНДЕКСАМИ, РАВНЫМИ НУЛЮ.

ПРИМЕРЫ ОБОЗНАЧЕНИЯ:

- МАССИВ ПЕРЕМЕННОЙ, ОРГАНИЗОВАННЫЙ ОПЕРАТОРОМ DIMX3%(7,19) БУДЕТ ОБОЗНАЧЕН В ОЗУ ИДЕНТИФИКАТОРОМ, СОСТОЯЩИМ ИЗ КОДОВ 05.08, 04.03, 01.03, 00.07;
- МАССИВ F\$(3,2) БУДЕТ ОБОЗНАЧЕН 04.06, 08.10, 00.02, 00.03;
- ПЕРЕМЕННАЯ Y0 БУДЕТ ОБОЗНАЧЕНА 05.09, 00.00, 00.00, 00.00.

ПОСЛЕ ИДЕНТИФИКАТОРА (В СТОРОНУ УВЕЛИЧЕНИЯ АДРЕСОВ ОЗУ) РАСПОЛАГАЮТСЯ ЗНАЧЕНИЯ МАССИВОВ ПЕРЕМЕННЫХ. ВЕЩЕСТВЕННЫЕ ПЕРЕМЕННЫЕ ЗАНИМАЮТ ПО 8 БАЙТ, ЦЕЛЫЕ ПО 2 БАЙТА, МАССИВ ТЕКСТОВЫХ ПЕРЕМЕННЫХ НАЧИНАЕТСЯ С БАЙТА, ОПРЕДЕЛЯЮЩЕГО ДЛИНУ ПЕРЕМЕННЫХ И ДАЛЕЕ КАЖДАЯ ТЕКСТОВАЯ ПЕРЕМЕННАЯ ЗАНИМАЕТ ОТВЕДЕННОЕ ДЛЯ НЕЕ КОЛИЧЕСТВО БАЙТ. ЕСЛИ ДЛИНА ТЕКСТОВОЙ ПЕРЕМЕННОЙ МЕНЬШЕ ЗАДАННОЙ, ТО ПОСЛЕ ЕЕ ОБОЗНАЧЕНИЯ СТОИТ КОД 00.00.

В ОБЛАСТИ СОМ РАСПОЛОЖЕНЫ ФАЙЛЫ FL. КАЖДЫЙ ФАЙЛ ЗАНИМАЕТ КОЛИЧЕСТВО К-БАЙТ, ОТВЕДЕННОЕ ДЛЯ НЕГО ОПЕРАТОРОМ СОМ.

В СЛУЖЕБНОЙ ЗОНЕ ОЗУ РАЗМЕЩАЕТСЯ БУФЕРНАЯ ПАМЯТЬ, НЕОБХОДИМАЯ ДЛЯ РАБОТЫ ИНТЕРПРЕТАТОРА, СЛУЖЕБНЫЕ РЕГИСТРЫ- УКАЗАТЕЛИ, РЕГИСТРЫ-ПРИЗНАКИ, ТАБЛИЦА АДРЕСОВ ПОДПРОГРАММ, РАБОЧИЙ СТЕК, РЕГИСТРЫ TR И RR, СЛУЖЕБНАЯ ЗОНА ЭВМ.

11.4. РАЗРАБОТКА ВНЕШНИХ ПОДПРОГРАММ.

БЛОКИ ВНЕШНИХ ПОДПРОГРАММ ПРИСТАВЛЯЮТСЯ К ТЕКСТУ ПРОГРАММЫ-ИНТЕРПРЕТАТОРА, ПРИ ЭТОМ ОНИ ЯВЛЯЮТСЯ КАК-БЫ ЕГО ЛОГИЧЕСКИМ ПРОДОЛЖЕНИЕМ. ПОДПРОГРАММЫ С ИМЕНАМИ A..Z ИМЕЮТ, СООТВЕТСТВЕННО, ЛОГИЧЕСКИЕ НОМЕРА В ИНТЕРПРЕТАТОРЕ 223..248.

ОБРАЩЕНИЕ К ПОДПРОГРАММЕ ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРОМ CALL, ЗАПИСЫВАЕМОМ В СЛЕДУЮЩЕМ ВИДЕ:

CALL<ИМЯ ПОДПРОГРАММЫ> [,<ВХОДНЫЕ ПАРАМЕТРЫ>]

ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА CALL ИНТЕРПРЕТАТОР ПО ИМЕНИ ПОДПРОГРАММЫ ОПРЕДЕЛЯЕТ ЕЕ НОМЕР И ОРГАНИЗУЕТ ЕЕ ВЫПОЛНЕНИЕ, ПОСЛЕ ЧЕГО ВЫПОЛНЯЕТСЯ СЛЕДУЮЩИЙ ОПЕРАТОР ПРОГРАММЫ.

НА ВХОДЕ ВО ВНЕШНЮЮ ПОДПРОГРАММУ ИНТЕРПРЕТАТОР УСТАНОВЛИВАЕТ СЛЕДУЮЩЕЕ СОСТОЯНИЕ РЕГИСТРОВ:

- R1 - АДРЕС ПЕРВОГО СИМВОЛА ВХОДНЫХ ПАРАМЕТРОВ, НЕ РАВНОГО ПРОБЕЛУ. ЕСЛИ ПАРАМЕТРОВ НЕТ, ТО В R1 - АДРЕС СЛЕДУЮЩЕГО ЗА ОПЕРАТОРОМ CALL РАЗДЕЛИТЕЛЯ (СИМВОЛЫ <П>, <: >, < > ИЛИ КОД СЛОВА ELSE);

- VD И VP РАВНЫ НУЛЮ.

НА ВЫХОДЕ ИЗ ПОДПРОГРАММЫ В R1 ДОЛЖЕН БЫТЬ ЗАНЕСЕН АДРЕС РАЗДЕЛИТЕЛЯ, СЛЕДУЮЩЕГО ЗА ОПЕРАТОРОМ CALL, VD=VP=0. ПОДПРОГРАММА ДОЛЖНА ЗАКАНЧИВАТЬСЯ КОМАНДОЙ 05.11.

ВО ВНЕШНЕЙ ПОДПРОГРАММЕ МОГУТ ИСПОЛЬЗОВАТЬСЯ БЕЗ ВОССТАНОВЛЕНИЯ НАЧАЛЬНОГО СОСТОЯНИЯ СЛЕДУЮЩИЕ РЕГИСТРЫ И ЗОНЫ ОЗУ:

- РЕГИСТРЫ X, Y, Z;

- РЕГИСТРЫ R0, R2..R12;

- РЕГИСТРЫ T11, T13..T15;

- 70 БАЙТ СТЕКА, НА ВЕРШИНУ КОТОРОГО УКАЗЫВАЕТ R13;

- СВОБОДНАЯ ОБЛАСТЬ ПОЛЬЗОВАТЕЛЯ;

- БОЛЬШОЙ БУФЕР СЛУЖЕБНОЙ ПАМЯТИ: 320 БАЙТ С АДРЕСА 07.08.00.00 (СОДЕРЖИМОЕ T5);

- МАЛЫЙ БУФЕР СЛУЖЕБНОЙ ПАМЯТИ: 16 БАЙТ С АДРЕСА 07.14.15.00 (T4);

- СМЕННЫЕ НЕЗАНЯТЫЕ СТРАНИЦЫ. НОМЕРА КАНАЛОВ 3..14 СООТВЕТСТВУЮТ СТРАНИЦАМ 00..11.

ПРИМЕЧАНИЕ: ПРИ ОПЕРАЦИЯХ СО СМЕННЫМИ СТРАНИЦАМИ НА ВЫХОДЕ ИЗ ПОДПРОГРАММЫ ДОЛЖНО БЫТЬ ВОССТАНОВЛЕНО ИСХОДНОЕ СОСТОЯНИЕ ПОДКЛЮЧАЕМОГО СЕГМЕНТА, Т.Е. 15.14.13.12. ПЕРЕД СМЕНОЙ СТРАНИЦЫ КОД СЕГМЕНТА 15.X.13.12 ДОЛЖЕН БЫТЬ ЗАНЕСЕН В РЕГИСТР T12.

ОБМЕН ПАРАМЕТРАМИ С ВНЕШНИМИ ПОДПРОГРАММАМИ УДОБНО ПРОИЗВОДИТЬ ЧЕРЕЗ СЛУЖЕБНЫЕ РЕГИСТРЫ RR И TR.

РЕГИСТРЫ RR РАСПОЛОЖЕНЫ НАЧИНАЯ С АДРЕСА 07.10.05.08 (Т7) И ЗАНИМАЮТ ПО 8 БАЙТ. В РЕГИСТРАХ СОДЕРЖАТСЯ ВЕЩЕСТВЕННЫЕ ЧИСЛА В ВОСЬМИБАЙТОВОМ ФОРМАТЕ.

РЕГИСТРЫ TR ЗАНИМАЮТ ПО 128 БАЙТ С АДРЕСА 07.09.04.00. В РЕГИСТРАХ РАСПОЛАГАЮТСЯ КОДЫ СИМВОЛОВ, ДОПУСТИМЫХ В АЛФАВИТЕ ЯЗЫКА. ТЕКСТОВЫЕ ДАННЫЕ ДЛИНОЙ МЕНЕЕ 128 СИМВОЛОВ ЗАКАНЧИВАЮТСЯ КОДОМ 00.00.

ВО ВНЕШНЕЙ ПОДПРОГРАММЕ МОГУТ БЫТЬ ИСПОЛЬЗОВАНЫ ПОДПРОГРАММЫ НИЗШЕГО УРОВНЯ, КОТОРЫЕ МОЖНО ВСТАВИТЬ НА РЕЗЕРВНЫЕ МЕСТА В БЛОКЕ ПП. ОБРАЩЕНИЕ К ПОДПРОГРАММЕ НИЗШЕГО УРОВНЯ В ЗАВИСИМОСТИ ОТ ЕЕ НОМЕРА ПРОИЗВОДИТСЯ СЛЕДУЮЩИМИ КОМАНДАМИ:

ПОДПРОГРАММА 180 - 10.14, 06.06;
ПОДПРОГРАММА 181 - 10.14, 06.08;
ПОДПРОГРАММА 182 - 10.14, 06.10
И Т.Д.

11.5. ОПИСАНИЕ НЕКОТОРЫХ ПОДПРОГРАММ ИНТЕРПРЕТАТОРА.

ПРИ РАЗРАБОТКЕ ВНЕШНИХ ПОДПРОГРАММ УДОБНО И ВЫГОДНО ИСПОЛЬЗОВАТЬ НЕКОТОРЫЕ ПОДПРОГРАММЫ (ДАЛЕЕ - ПП) ИНТЕРПРЕТАТОРА. НИЖЕ ДАЕТСЯ ОПИСАНИЕ НАИБОЛЕЕ УПОТРЕБИТЕЛЬНЫХ ПП, КОТОРЫЕ ЛЕГКО МОГУТ БЫТЬ ИСПОЛЬЗОВАНЫ ПОЛЬЗОВАТЕЛЕМ В СВОИХ ПОДПРОГРАММАХ. ОПИСАНИЕ ОСТАЛЬНЫХ ПОДПРОГРАММ НЕ ПРИВОДИТСЯ ВВИДУ ЛИБО ИХ ПРОСТОТЫ, ЛИБО, НАОБОРОТ, СЛОЖНОСТИ ОПИСАНИЯ И ОРГАНИЗАЦИИ ВХОДА-ВЫХОДА. ПП ИНТЕРПРЕТАТОРА В СЛУЧАЕ, ЕСЛИ ОНИ ВЫПОЛНЯЮТ КАКИЕ-ЛИБО ОПЕРАЦИИ, НЕ ПОНЯТНЫЕ ПОЛЬЗОВАТЕЛЮ, ПРИМЕНЯТЬ НЕ РЕКОМЕНДУЕТСЯ, Т.К. ЭТО МОЖЕТ ПРИВЕСТИ К НЕПРЕДСКАЗУЕМЫМ ПОСЛЕДСТВИЯМ.

ПП2. ПЕЧАТЬ ЧИСЛА В ЗАДАННОМ ФОРМАТЕ.

ОБРАЩЕНИЕ К ПП: 10.13, 00.02;

ВХОД: РЕГИСТР X - ЧИСЛО, S1 - ФОРМАТ. ФОРМАТ !N1.N2! ЗАДАЕТСЯ В S1: СТАРШАЯ ТЕТРАДА - N1, МЛАДШАЯ - N2. ДЛЯ ПЕЧАТИ В ФОРМАТЕ !A! - S1=14.01, В ФОРМАТЕ !E! - S1=00.10. УСТРОЙСТВО ВЫВОДА ЗАДАЕТСЯ АНАЛОГИЧНО ПП3 (СМ. НИЖЕ).

ИСПОЛЬЗУЕМЫЕ РЕГИСТРЫ (ДАЛЕЕ - ИР): R1, R5..R12, X, Y, 20 БАЙТ БОЛЬШОГО БУФЕРА.

ПП3. ВЫВОД СИМВОЛА.

ОБРАЩЕНИЕ К ПП: 10.13, 00.04;

ВХОД: S3 - СИМВОЛ, ДОПУСТИМЫЙ В АЛФАВИТЕ. ДЛЯ СТРОЧНЫХ БУКВ ДОЛЖЕН БЫТЬ УСТАНОВЛЕН СТАРШИЙ БИТ S3.

ДЛЯ ВЫВОДА СИМВОЛА НА ПРИНТЕР НЕОБХОДИМО В БАЙТЕ С АДРЕСОМ 07.12.04.14 (СОДЕРЖИМОЕ T3) УСТАНОВИТЬ МЛАДШИЙ БИТ, ДЛЯ ВЫВОДА НА ДИСПЛЕЙ ОЧИСТИТЬ. ПРИ ВХОДЕ ВО ВНЕШНЮЮ ПП ДАННЫЙ БИТ ВСЕГДА ОЧИЩЕН И ДОЛЖЕН БЫТЬ ОЧИЩЕН НА ЕЕ ВЫХОДЕ. ОСТАЛЬНЫЕ БИТЫ В УКАЗАННОМ БАЙТЕ ИЗМЕНЯТЬ НЕ ДОПУСКАЕТСЯ.

ВЫХОД: S3 - ВЫВЕДЕННЫЙ СИМВОЛ, S1 - СЧЕТЧИК ПОЗИЦИЙ УСТРОЙСТВА ВЫВОДА.

ИР: R8..R12.

ПП38. ПОИСК МАССИВА ПЕРЕМЕННОЙ.

ВЫЗОВ: 10.13, 04.10;

ВХОД: R8 - ИДЕНТИФИКАТОР МАССИВА;

ВЫХОД: R10 - АДРЕС МАССИВА: АДРЕС ПЕРВОГО БАЙТА ПОСЛЕ ИДЕНТИФИКАТОРА (ДЛЯ ТЕКСТОВЫХ ПЕРЕМЕННЫХ - АДРЕС БАЙТА, ОПРЕДЕЛЯЮЩЕГО ДЛИНУ ПЕРЕМЕННЫХ). ЕСЛИ МАССИВ ПЕРЕМЕННЫХ НЕ ОПРЕДЕЛЕН, ТО R10=0; R11 - ИНДЕКС МАССИВА: S6 - N, S7 - M; R8 - БЕЗ ИЗМЕНЕНИЙ.

ИР: R8..R11.

ПП43. ВВОД СТРОКИ С ДИСПЛЕЯ.

ВЫЗОВ: 10.13, 05.04;

ВХОД: R1 - АДРЕС ДЛЯ ВВОДА;

ВЫХОД: R1 - НАЧАЛО СТРОКИ; R12 - АДРЕС ПОСЛЕДНЕГО СИМВОЛА - <ПС>.

ИР: R1, R8..R12.

ПП52. СЧИТЫВАНИЕ ТЕКСТОВОГО ВЫРАЖЕНИЯ.

ВЫЗОВ: 10.13, 06.06;

ВХОД: R1 - НАЧАЛО ОБОЗНАЧЕНИЯ (ТЕКСТОВОГО ВЫРАЖЕНИЯ): ПЕРВЫЙ СИМВОЛ, НЕ РАВНЫЙ ПРОБЕЛУ;

ВЫХОД: R1 - НАЧАЛО СЛЕДУЮЩЕГО ОБОЗНАЧЕНИЯ (ДАЛЕЕ БУДЕТ ОБОЗНАЧАТЬСЯ <R1 = R1+D>), РЕЗУЛЬТАТ ТЕКСТОВОГО ВЫРАЖЕНИЯ В БОЛЬШОМ БУФЕРЕ, R11 - НАЧАЛО РЕЗУЛЬТАТА (БУФЕРА), R10 - КОЛИЧЕСТВО СИМВОЛОВ.

ИР: R0..R3, R5..R12, БОЛЬШОЙ БУФЕР.

ПП59. СЧИТЫВАНИЕ ИНДЕКСА. (M,N).

ВЫЗОВ: 10.13, 07.04;

ВХОД: R1 - НАЧАЛО ОБОЗНАЧЕНИЯ (АДРЕС ОТКРЫВАЮЩЕЙ СКОБКИ);

ВЫХОД: R1 = R1+D, R11 - ИНДЕКС: S6 - N, S7 - M;

ИР: R1, R8..R12, X.

ПП65. РАСЧЕТ АРИФМЕТИЧЕСКОГО ВЫРАЖЕНИЯ.

ВЫЗОВ: 10.13, 08.00;

ВХОД: R1 - НАЧАЛО ОБОЗНАЧЕНИЯ;

ВЫХОД: R1 = R1+D, X - РЕЗУЛЬТАТ;

ИР: R0..R3, R6..R12, T11, T15, X, Y, БОЛЬШОЙ БУФЕР, МАЛЫЙ БУФЕР.

ПП89.СЧИТЫВАНИЕ ЧИСЛОВОЙ КОНСТАНТЫ.

ВЫЗОВ: 10.13, 11.00;
ВХОД: R1 - НАЧАЛО ОБОЗНАЧЕНИЯ;
ВЫХОД: R1 = R1+D, X - ЗНАЧЕНИЕ;
ИР: R1, R8..R10, X, МАЛЫЙ БУФЕР.

ПП94.СЧИТЫВАНИЕ ИМЕНИ ПЕРЕМЕННОЙ.

ВЫЗОВ: 10.13, 11.10;
ВХОД: R1 - НАЧАЛО ОБОЗНАЧЕНИЯ;
ВЫХОД: R1 = R1+D, R8 - ИДЕНТИФИКАТОР;
ИР: R1, R8, S3.

ПП100.ОСТАНОВ ФОНОВОЙ ЗАДАЧИ ДО ВЫХОДА ИЗ РЕДАКТОРА.

ВЫЗОВ: 10.13, 12.06;
ИР: S2, R11

ПП134.ОЧИСТКА РЕГИСТРОВ TR.

ВЫЗОВ: 10.14, 00.10;
ВЫХОД: P11 - АДРЕС РЕГИСТРОВ TR;
ИР: R9, R11.

ПП154.ВЫЗОВ РЕГИСТРА TR.

ВЫЗОВ: S1 - НОМЕР РЕГИСТРА;
ВЫХОД: P11 - АДРЕС РЕГИСТРА, P10 - КОЛИЧЕСТВО СИМВОЛОВ;
ИР: R8..R11.

ПОДПРОГРАММА ОБРАБОТКИ ОШИБКИ.

ВЫЗОВ: 00.10;

КОД ПП МОЖНО УКАЗЫВАТЬ В ЛЮБОМ МЕСТЕ ПОДПРОГРАММЫ. ПП ПЕЧАТАЕТ СООБЩЕНИЕ ОБ ОШИБКЕ, ГДЕ ШИФР ОШИБКИ - НОМЕР ПОДПРОГРАММЫ В КОТОРОЙ УКАЗАН КОД 00.10. ПЕРЕД ВХОДОМ В ПП НЕ ОБЯЗАТЕЛЬНО ВОССТАНАВЛИВАТЬ ВД, R1 И R13. ПП ОСУЩЕСТВЛЯЕТ ПЕРЕХОД В РЕЖИМ "ДИАЛОГ".

ПЕРЕХОД В РЕЖИМ "ДИАЛОГ" БЕЗ СООБЩЕНИЯ ОБ ОШИБКЕ МОЖНО ОСУЩЕСТВИТЬ, УКАЗАВ КОДЫ 04.07, 04.08.

// Конец файла k35a_index-2. Далее идёт текст из файла k35b_index-3:

ПРИЛОЖЕНИЯ.

П Р И Л О Ж Е Н И Е N1.

ШИФРЫ ОШИБОК И ВЕРОЯТНЫЕ ПРИЧИНЫ ИХ ВОЗНИКНОВЕНИЯ

- 12 ПРОПУСК ЗАКРЫВАЮЩЕЙ СКОБКИ АРГУМЕНТА СПЕЦФУНКЦИИ ИЛИ ИНДЕКСА
СИНТАКСИЧЕСКАЯ ОШИБКА В ОБОЗНАЧЕНИИ ИНДЕКСА
- 15 ПЕРЕПОЛНЕНИЕ ОЗУ, ФАЙЛА FL ИЛИ БУФЕРА ПРИ ПОСТРОЧНОМ ВВОДЕ
- 18 ВВОДИМЫЕ ДАННЫЕ В INPUT НЕ СООТВЕТСТВУЮТ СПИСКУ ВВОДА
- 20 ПРОПУСК РАЗДЕЛИТЕЛЬНОГО ДВОЕТОЧИЯ
СИНТАКСИЧЕСКАЯ ОШИБКА
- 23 СЛИШКОМ БОЛЬШОЙ ДВУМЕРНЫЙ ИНДЕКС
- 37 ПРИСВОЕНИЕ ТЕКСТОВОМУ ЭЛЕМЕНТУ НЕДОПУСТИМОГО ЗНАЧЕНИЯ
- 40 ОШИБКА НМЛ
- 46 ИНДЕКС ПЕРЕМЕННОЙ БОЛЬШЕ ЗАДАННОГО В DIM
- 47 ФАЙЛ НЕ УМЕЩАЕТСЯ В ОТВЕДЕННУЮ ДЛЯ НЕГО ОБЛАСТЬ ОЗУ
- 48 ВЫВОД ТЕКСТОВОГО ФАЙЛА НА ЗАНЯТУЮ СТРАНИЦУ
- 49 ОШИБКА В ОБОЗНАЧЕНИИ СЛУЖЕБНОГО СЛОВА
АРГУМЕНТ ФУНКЦИИ НЕ ЗАКЛЮЧЕН В СКОБКИ
ПРОПУСК ЗАКРЫВАЮЩЕГО АПОСТРОФА В ТЕКСТОВОЙ КОНСТАНТЕ
- 52 РЕЗУЛЬТАТ ТЕКСТОВОГО ВЫРАЖЕНИЯ БОЛЬШЕ ДОПУСТИМОЙ ДЛИНЫ
НЕДОПУСТИМЫЙ ЭЛЕМЕНТ В ТЕКСТОВОМ ВЫРАЖЕНИИ
- 53 НЕДОПУСТИМЫЙ НОМЕР КАНАЛА ВВОДА-ВЫВОДА
- 54 НЕДОПУСТИМЫЙ НОМЕР РЕГИСТРА TR
- 57 ПЕРЕПОЛНЕНИЕ ОЗУ ПРИ ЗАПИСИ В ОБЛАСТЬ DIM ИЛИ COM
- 60 НЕДОПУСТИМОЕ ЗНАЧЕНИЕ ИНДЕКСНОГО ЭЛЕМЕНТА
НЕДОПУСТИМАЯ ВЕЛИЧИНА УПРАВЛЯЮЩЕГО ПАРАМЕТРА В ON
- 61 ОБРАЩЕНИЕ К ФАЙЛУ FL ПРИ ПОЛОЖЕНИИ УКАЗАТЕЛЯ "КОНЕЦ
ФАЙЛА"
- 65 НЕПРАВИЛЬНОЕ ПРИМЕНЕНИЕ СКОБОК В АРИФМЕТИЧЕСКОМ
ВЫРАЖЕНИИ
- 73 НЕ ОПРЕДЕЛЕНА ЧИСЛОВАЯ ПЕРЕМЕННАЯ
ПРИМЕНЕНИЕ ТЕКСТОВОЙ ПЕРЕМЕННОЙ В АРИФМЕТИЧЕСКОМ
ВЫРАЖЕНИИ
- 74 НЕДОПУСТИМОЕ ПРИСВОЕНИЕ ЗНАЧЕНИЙ
- 75 СИНТАКСИЧЕСКАЯ ОШИБКА В SEG
- 83 НЕДОПУСТИМОЕ ЗНАЧЕНИЕ ИНДЕКСНОГО ВЫРАЖЕНИЯ
- 84 НЕ ОПРЕДЕЛЕН ФАЙЛ FL

- 86 НЕКОРРЕКТНАЯ СИТУАЦИЯ В ПРОЦЕССЕ ВЫЧИСЛЕНИЙ
- 91 НЕДОПУСТИМАЯ ВЕЛИЧИНА ЦЕЛОЧИСЛЕННОЙ ПЕРЕМЕННОЙ
- 95 НЕДОПУСТИМЫЙ НОМЕР КАНАЛА ВВОДА-ВЫВОДА
- 99 НЕДОПУСТИМЫЙ НОМЕР СТРОКИ ПРОГРАММЫ ИЛИ ФАЙЛА FL
- 105 СИНТАКСИЧЕСКАЯ ОШИБКА В CHANGE
- 106 СИНТАКСИЧЕСКАЯ ОШИБКА В ОБОЗНАЧЕНИИ ИМЕНИ ОПЕРАТОРА
- 107 СИНТАКСИЧЕСКАЯ ОШИБКА В FOR
ПРЕВЫШЕНИЕ ДОПУСТИМОГО УРОВНЯ ЦИКЛОВ
- 108 ОШИБКА В CLEAR:
ЧИСТКА ПРОГРАММЫ В ПРОГРАММНОМ РЕЖИМЕ
СИНТАКСИЧЕСКАЯ ОШИБКА
- 109 ОШИБКА В COM
- 110 СИНТАКСИЧЕСКАЯ ОШИБКА В DIM
- 111 INPUT В РЕЖИМЕ "ДИАЛОГ"
НЕДОПУСТИМЫЙ ЭЛЕМЕНТ СПИСКА ПРИ ПОСТРОЧНОМ ВВОДЕ В INPUT
ПРИ ПОСТРОЧНОМ ВВОДЕ ВЫЗЫВАЕМАЯ СТРОКА НЕ ПОМЕЩАЕТСЯ В
БУФЕР
INPUT В НЕДОПУСТИМОМ УРОВНЕ ЦИКЛОВ (НИЖЕ 4)
- 118 СИНТАКСИЧЕСКАЯ ОШИБКА В SET
- 122 ОШИБКА В NEXT: НЕПРАВИЛЬНОЕ ВЛОЖЕНИЕ ЦИКЛОВ
- 124 ОШИБКА В SCREEN: ФАЙЛ В РАСПАКОВАННОМ ВИДЕ НЕ ПОМЕЩАЕТСЯ
В ОЗУ
- 127 НЕДОПУСТИМОЕ ЗНАЧЕНИЕ УПРАВЛЯЮЩЕГО ПАРАМЕТРА В ON
- 128 ПРЕВЫШЕНИЕ ДОПУСТИМОГО УРОВНЯ ПОДПРОГРАММ
- 138 ОШИБКА В PRINTFT:
НЕДОПУСТИМЫЙ НОМЕР СТРАНИЦЫ
НА СТРАНИЦЕ НАХОДЯТСЯ ДАННЫЕ ДРУГОГО ТИПА
- 141 НЕ ЗАДАН МАССИВ ПЕРЕМЕННОЙ ОПЕРАТОРОМ DIM
ИСПОЛЬЗОВАНИЕ ПЕРЕМЕННОЙ С ИНДЕКСОМ В КАЧЕСТВЕ
УПРАВЛЯЮЩЕЙ В FOR
НЕДОПУСТИМОЕ ПРИСВОЕНИЕ ЗНАЧЕНИЙ
- 146 НЕ ОПРЕДЕЛЕНА ТЕКСТОВАЯ ПЕРЕМЕННАЯ
- 149 НЕДОПУСТИМЫЙ НОМЕР РЕГИСТРА RR
- 150 ПЕРЕПОЛНЕНИЕ ФАЙЛА FL
- 151 ОШИБКА В VAL
- 152 ОШИБКА В SIZE
- 153 ОШИБКА ПРИ ОБРАЩЕНИИ К ТЕКСТОВОМУ ФАЙЛУ НА СМЕННОЙ
СТРАНИЦЕ:
НЕ ОПРЕДЕЛЕН ТЕКСТОВЫЙ ФАЙЛ
УКАЗАТЕЛЬ УСТАНОВЛЕН НА КОНЕЦ ФАЙЛА
СМЕННАЯ СТРАНИЦА ЗАНЯТА ДРУГИМ ТИПОМ ДАННЫХ

- 155 НЕДОПУСТИМЫЙ АРГУМЕНТ СПЕЦФУНКЦИИ
СИНТАКСИЧЕСКАЯ ОШИБКА
- 159 СИНТАКСИЧЕСКАЯ ОШИБКА В PRINT USING
- 165 СИНТАКСИЧЕСКАЯ ОШИБКА В PRINT:
НЕДОПУСТИМЫЙ НОМЕР КАНАЛА ВЫВОДА;
ОШИБКА ПРИ ОБОЗНАЧЕНИИ ФОРМАТА ПЕЧАТИ ЧИСЕЛ
- 168 НЕ ОПРЕДЕЛЕНА ПЕРЕМЕННАЯ В REN
СИНТАКСИЧЕСКАЯ ОШИБКА В REN
- 171 НЕ ОПРЕДЕЛЕНА ЗАПИСЫВАЕМЫЕ ДАННЫЕ В SAVE
- 173 НЕДОПУСТИМЫЙ НОМЕР КАНАЛА В LOAD
НЕДОПУСТИМАЯ ЗАГРУЗКА FL В СВОБОДНУЮ ОБЛАСТЬ
ПОЛЬЗОВАТЕЛЯ
ЗАГРУЗКА ПРОГРАММЫ В ПРОГРАММНОМ РЕЖИМЕ
- 176 ПЕРЕПОЛНЕНИЕ ОЗУ ПРИ ЗАПИСИ В ТЕКСТОВЫЙ ФАЙЛ

П Р И Л О Ж Е Н И Е N2.

НАЗНАЧЕНИЕ УПРАВЛЯЮЩИХ СИМВОЛОВ НА ДИСПЛЕЕ.

ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ УПРАВЛЯЮЩИХ СИМВОЛОВ	!	ТИП ДИСПЛЕЯ	!
	!	- - - - -	-!
	!	!15ИЭ 200*	!15ИЭ-00-
	!	*140-017	! -013

А) .В РЕЖИМЕ ДИАЛОГ И ПРИ ВВОДЕ ДАННЫХ
В INPUT.

- КОНЕЦ ВВОДА, ПЕРЕХОД НА НАЧАЛО СЛЕДУЮЩЕЙ СТРОКИ;	ПС	ПС, ВК
- "ЗАВОЙ"	ВК, < > В РУС. РЕГИСТРЕ	ЗБ
- ГОРИЗОНТАЛЬНАЯ ТАБУЛЯЦИЯ С ВВОДОМ ПРОБЕЛОВ (УСТАНОВКА МАРКЕРА НА БЛИЖАЙШУЮ ПОЗИЦИЮ С НОМЕРОМ КРАТНЫМ ВОСЬМИ) .	УПР/Г	ГТ

Б) .В ПРОГРАММНОМ РЕЖИМЕ

- ПЕРЕХОД В РЕЖИМ "ДИАЛОГ"	УПР/С	СУ/С
- ВЫЗОВ ОПЕРАТОРА STOP	ОСТАЛЬНЫЕ	ОСТАЛЬНЫЕ

В) .ПРИ РАБОТЕ С НМЛ В СЛУЧАЕ ВЫХОДА
НА ПРОЗРАЧНЫЙ РАКОРД

- ПЕРЕХОД К РЕЖИМУ "ДИАЛОГ"	УПР/С	СУ/С
- КОНЕЦ РАБОТЫ НМЛ, ПРОДОЛЖЕНИЕ РАБОТЫ ПРОГРАММЫ	УПР/Е	СУ/Е
- ПРОДОЛЖЕНИЕ РАБОТЫ НМЛ	УПР/Q	СУ/Q, ПРМ

Г) .ВО ВРЕМЯ ОСТАНОВА ПРИНТЕРА ПОСЛЕ
ВЫВОДА ОЧЕРЕДНОЙ СТРАНИЦЫ

- ПЕЧАТЬ ЕЩЕ ОДНОЙ СТРОКИ	ПС	ПС
- ПЕЧАТЬ ЕЩЕ ОДНОЙ СТРАНИЦЫ	ОСТАЛЬНЫЕ	ОСТАЛЬНЫЕ

Д) .ПРИ РАБОТЕ С ПОДПРОГРАММОЙ-РЕДАКТОРОМ

- ВЫХОД ИЗ РЕДАКТОРА	УПР/А	СУ/А, С1
- НАЧАЛО ТЕКСТА	УПР/В	СУ/В
- СЛЕДУЮЩАЯ СТРАНИЦА	УПР/Д	СУ/Д, СТР
- НОВАЯ СТРОКА	ПС, < >	ПС
- НАЧАЛО СТРОКИ	-	ВК
- НАЧАЛО ЭКРАНА	< >	< ↖ >
- ГОРИЗОНТАЛЬНАЯ ТАБУЛЯЦИЯ С ВВОДОМ ПРОБЕЛОВ	УПР/І	ГТ
- ГОРИЗОНТАЛЬНАЯ ТАБУЛЯЦИЯ БЕЗ ВВОДА ПРОБЕЛОВ	УПР/Х	ТАБ
- МАРКЕР ВВЕРХ	< >	< ↑ >
- МАРКЕР ВНИЗ	< >	< ↓ >
- МАРКЕР ВЛЕВО	< >	< ← >
- МАРКЕР ВПРАВО	< >	< → >
- СМЫКАНИЕ В СТРОКЕ	УПР/Т	< ←→ >
- РАЗМЫКАНИЕ В СТРОКЕ	УПР/S	< →← >
- ИСКЛЮЧЕНИЕ СТРОКИ	УПР/R	< ⊥ >
- ВСТАВЛЕНИЕ СТРОКИ	УПР/V	< ⊥̄ >
- СТИРАНИЕ СТРОКИ	< >	СТС
- "ЗАБОЙ"	ВК, < > В РУС. РЕГИСТРЕ	ЗБ

ПРИМЕЧАНИЯ: 1) .КЛАВИША < ↓ > ПРИ ПОЛОЖЕНИИ МАРКЕРА В ПОСЛЕДНЕЙ СТРОКЕ ТЕКСТА ИГНОРИРУЕТСЯ. ДЛЯ РАСШИРЕНИЯ ТЕКСТА СЛЕДУЕТ ПОЛЬЗОВАТЬСЯ КЛАВИШЕЙ ПС.

2) .КЛАВИШИ, РАСШИРЯЮЩИЕ ТЕКСТ, ПРИ ОТСУТСТВИИ СВОБОДНОГО МЕСТА В ОЗУ ИГНОРИРУЮТСЯ.

П Р И Л О Ж Е Н И Е N3.

ВАРИАНТЫ ПРОГРАММЫ-ИНТЕРПРЕТАТОРА.

А) .ВАРИАНТ 32-ВК.

РЕДАКЦИЯ НА
КОНТРОЛЬНАЯ СУММА -
КОЛИЧЕСТВО ШАГОВ -
РАЗМЕР ОЗУ ДЛЯ ПОЛЬЗОВАТЕЛЯ (БЕЗ БЛОКА ВНЕШНИХ ПП) -
НОМЕР ШАГА ДЛЯ БЛОКИРОВКИ БЛОКА ВНЕШНИХ ПП

СОСТАВ КОМПЛЕКСА:

- ЭВМ ЭЛЕКТРОНИКА ДЗ-28 С ОЗУ 32 К-БАЙТ И СКОРОСТЬЮ ВЫПОЛНЕНИЯ ОПЕРАЦИЙ 500 ТЫС.ТАКТОВ/СЕК;
- ДИСПЛЕЙ 15ИЭ 200*140-017;
- ЭПУ "КОНСУЛ 260.1".

ДОПОЛНИТЕЛЬНЫЕ ФУНКЦИИ.

В ОПЕРАТОРЕ PRINT МОЖЕТ БЫТЬ УКАЗАНО ОБОЗНАЧЕНИЕ <@1>, ОЗНАЧАЮЩЕЕ ДЛЯ ЭПУ - УСТАНОВКУ КРАСНОЙ ЛЕНТЫ, ДЛЯ ДИСПЛЕЯ - НЕГАТИВНОЕ ИЗОБРАЖЕНИЕ СИМВОЛОВ. НАЗНАЧЕНИЕ ДЕЙСТВУЕТ ДО УКАЗАНИЯ В ОПЕРАТОРЕ PRINT ОБОЗНАЧЕНИЯ <@0>, А ДЛЯ ДИСПЛЕЯ, КРОМЕ ТОГО - ДО ВЫХОДА В РЕЖИМ "ДИАЛОГ".

ОСОБЕННОСТИ РАБОТЫ С ЭПУ "КОНСУЛ".

СИМВОЛ <КАВЫЧКИ> ЗАМЕНЯЕТСЯ НА АПОСТРОФ, СИМВОЛ < > НА ДВОЕТОЧИЕ.

ОСТАНОВ ПЕЧАТИ МОЖНО В ЛЮБОЙ МОМЕНТ ВЫЗВАТЬ ОТЖАТИЕМ КЛАВИШИ "РАСПЕЧАТКА", ПОСЛЕ ЕЕ ПОВТОРНОГО ВКЛЮЧЕНИЯ ПЕЧАТЬ БУДЕТ ПРОДОЛЖЕНА.

Б) .ВАРИАНТ 128-ФР.

РЕДАКЦИЯ НА
КОНТРОЛЬНАЯ СУММА -
КОЛИЧЕСТВО ШАГОВ -
РАЗМЕР ОЗУ ДЛЯ ПОЛЬЗОВАТЕЛЯ (БЕЗ БЛОКА ВНЕШНИХ ПП) -
НОМЕР ШАГА ДЛЯ БЛОКИРОВКИ ВНЕШНИХ ПОДПРОГРАММ -

СОСТАВ КОМПЛЕКСА:

- ЭВМ "ЭЛЕКТРОНИКА ДЗ-28" С ОЗУ 128 К-БАЙТ И СКОРОСТЬЮ ВЫПОЛНЕНИЯ ОПЕРАЦИЙ 1 МЛН.ТАКТОВ/СЕК;
- ДИСПЛЕЙ 15ИЭ-00-013;
- ПРИНТЕР: АЦПУ "РОБОТРОН" ИЛИ ТЕРМОПЕЧАТАЮЩЕЕ УСТРОЙСТВО.

ДОПОЛНИТЕЛЬНЫЕ ФУНКЦИИ.

1) .В ОПЕРАТОРЕ PRINT МОЖЕТ БЫТЬ УКАЗАНА ФУНКЦИЯ PLOT, В АРГУМЕНТЕ КОТОРОЙ ДВУМЯ ИНДЕКСНЫМИ ВЫРАЖЕНИЯМИ МОГУТ БЫТЬ УКАЗАНЫ НОМЕР СТРОКИ И НОМЕР СТОЛБЦА, НА КОТОРЫЕ НУЖНО УСТАНОВИТЬ МАРКЕР ДИСПЛЕЯ. ПРИ НАЗНАЧЕННОМ НОМЕРЕ КАНАЛА ВЫВОДА, НЕ РАВНОМ НУЛЮ, ФУНКЦИЯ ИГНОРИРУЕТСЯ.

2) .ОПЕРАТОР CLEAR #0 СТИРАЕТ ИЗОБРАЖЕНИЕ НА ЭКРАНЕ И УСТАНОВЛИВАЕТ МАРКЕР В НАЧАЛО ЭКРАНА.

В) .ВАРИАНТ 128-ФМ.

РЕДАКЦИЯ НА
КОНТРОЛЬНАЯ СУММА -
КОЛИЧЕСТВО ШАГОВ -
РАЗМЕР ОЗУ ДЛЯ ПОЛЬЗОВАТЕЛЯ (БЕЗ БЛОКА ВНЕШНИХ ПП) -
НОМЕР ШАГА ДЛЯ БЛОКИРОВКИ БЛОКА ВНЕШНИХ ПП -

СОСТАВ КОМПЛЕКСА:

- ЭВМ "ЭЛЕКТРОНИКА ДЗ-28" С ОЗУ 128 К-БАЙТ И СКОРОСТЬЮ ВЫПОЛНЕНИЯ ОПЕРАЦИЙ 1 МЛН.ТАКТОВ/СЕК;
- ДИСПЛЕЙ 15ИЭ-00-013;
- ПРИНТЕР: УСТРОЙСТВО УВВПЧ-30-004.

ДОПОЛНИТЕЛЬНЫЕ ФУНКЦИИ.

АНАЛОГИЧНО ВАРИАНТУ 128-ФР.

ПРИМЕЧАНИЯ К ПРИЛОЖЕНИЮ N3:

1) .ВАРИАНТЫ 128-ФР И 128-ФМ МОЖНО ИСПОЛЬЗОВАТЬ НА ЭВМ С ОЗУ 32 К-БАЙТ, НО СЛЕДУЕТ УЧЕСТЬ, ЧТО ОБРАЩЕНИЕ К СМЕННЫМ СТРАНИЦАМ НЕ ЗАБЛОКИРОВАНО.

2) .ВАРИАНТ 32-ВК НЕ РЕКОМЕНДУЕТСЯ ПРИМЕНЯТЬ НА ЭВМ СО СКОРОСТЬЮ ВЫПОЛНЕНИЯ ОПЕРАЦИЙ 1 МЛН.ТАКТОВ/СЕК, Т.К. ВОЗМОЖНЫ СБОИ В РАБОТЕ НМЛ.

П Р И Л О Ж Е Н И Е N4.

ОПИСАНИЕ БЛОКА ВНЕШНИХ ПП.

ПОДПРОГРАММА А.

ПОДПРОГРАММА ПРЕДНАЗНАЧЕНА ДЛЯ ВЫРАВНИВАНИЯ ТЕКСТОВ ПО ШИРИНЕ СТРАНИЦЫ. ПРИ ВХОДЕ В ПОДПРОГРАММУ В РЕГИСТРЫ TR И TR1 ЗАНОСИТСЯ ТЕКСТ СТРОКИ: В РЕГИСТРЕ TR - НАЧАЛО, В РЕГИСТРЕ TR1 - КОНЕЦ. В ТЕЛЕ ОПЕРАТОРА CALL ИНДЕКСНЫМ ВЫРАЖЕНИЕМ УКАЗЫВАЕТСЯ ПАРАМЕТР, ОЗНАЧАЮЩИЙ КОЛИЧЕСТВО СИМВОЛОВ В СТРОКЕ. ВЕЛИЧИНА ДАННОГО ПАРАМЕТРА, А ТАКЖЕ, ДЛИНА СИМВОЛОВ В КАЖДОМ ИЗ РЕГИСТРОВ НЕ ДОЛЖНА ПРЕВЫШАТЬ 128.

НА ВЫХОДЕ ПОДПРОГРАММЫ:

- В РЕГИСТРЕ TR - СФОРМИРОВАННАЯ ТЕКСТОВАЯ СТРОКА ЗАДАННОЙ ДЛИНЫ, В КОТОРОЙ КОНЕЧНЫЕ ПРОБЕЛЫ РАВНОМЕРНО РАСПРЕДЕЛЕНА МЕЖДУ ОТДЕЛЬНЫМИ СЛОВАМИ В СТРОКЕ;
- В РЕГИСТРЕ TR1 НАХОДИТСЯ ОСТАТОК СТРОКИ.

ПРИМЕР ИСПОЛЬЗОВАНИЯ ПОДПРОГРАММЫ:

УСЛОВИЕ: ТРЕБУЕТСЯ РАСПЕЧАТАТЬ ТЕКСТОВЫЙ ФАЙЛ СТРОКАМИ ДЛИНОЙ ПО 64 СИМВОЛА, ПРИЧЕМ СТРОКИ, НАЧИНАЮЩИЕСЯ С ТРЕХ И БОЛЕЕ ПРОБЕЛОВ, ИЛИ СИМВОЛА <ТИРЕ>, А ТАКЖЕ ПУСТЫЕ СТРОКИ НАЧИНАТЬ ПЕЧАТАТЬ С НОВОЙ СТРОКИ.

ДАННУЮ ЗАДАЧУ ВЫПОЛНЯЕТ СЛЕДУЮЩИЙ СЕГМЕНТ ПРОГРАММЫ:

```

8100REM <ПП ПЕЧАТИ ТЕКСТА В ЗАДАННОМ ФОРМАТЕ>
8105D=64:REM <УСТАНОВКА ПАРАМЕТРА - ШИРИНЫ ПЕЧАТИ>
8110CLEAR TR:RESTORE #2
8120IF END #2 THEN GOSUB 8200:END ELSE LINPUT#2,Z$
8130IF'   '=Z$ORZ$=' 'OR'-' '=Z$THEN8300
8140IFLEN(TR)+LEN(TR1)<D THEN8170
8160CALLA,D:PRINT TR:TR=' ':8140
8170IFTR=' 'THENTR=TR1:8190
8180IFLEN(TR1)>0THENTR=TR+' '+TR1
8190TR1=Z$:Z$=' ':8120
8200IF LEN(TR)+LEN(TR1)>D THEN CALLA,D:8220
8205IFLEN(TR)>0ANDLEN(TR1)>0THENTR=TR+' '+TR1:TR1=' ':8220
8210IFTR+TR1=' 'THENRETURN
8215IFTR=' 'THENTR=TR1:TR1=' '
8220PRINT TR:TR=TR1:TR1=' ':8200
8300GOSUB8200:IFZ$=' 'THEN PRINT:8120 ELSE 8140

```

ПОДПРОГРАММА В.

ПОДПРОГРАММА ОСУЩЕСТВЛЯЕТ ЗАПИСЬ ТЕКСТОВОГО ФАЙЛА (КАНАЛ #2) НА НМЛ КОМАНДОЙ SAVEХ. ПОСЛЕ ВЫПОЛНЕНИЯ ПП В РЕГИСТРЕ RR - КОЛИЧЕСТВО БАЙТ, В РЕГИСТРЕ RR1 - КОНТРОЛЬНАЯ СУММА ЗАПИСАННОГО ФАЙЛА.

ПОДПРОГРАММА С.

ПОДПРОГРАММА ЗАГРУЖАЕТ ТЕКСТОВЫЙ ФАЙЛ С НМЛ, ИСПОЛЬЗУЯ КОМАНДУ LOADХ. ПОСЛЕ ВЫПОЛНЕНИЯ ПП В РЕГИСТРЕ RR - КОЛИЧЕСТВО БАЙТ, В РЕГИСТРЕ RR1 - КОНТРОЛЬНАЯ СУММА ФАЙЛА. ПРИ ОШИБКЕ НМЛ RR=RR1=0. ЕСЛИ ФАЙЛ НЕ УМЕЩАЕТСЯ В ОЗУ, ТО СТИРАЕТСЯ ВСЯ ИНФОРМАЦИЯ ПОЛЬЗОВАТЕЛЯ И ОСУЩЕСТВЛЯЕТСЯ ПОВТОРНЫЙ ЗАПУСК ИНТЕРПРЕТАТОРА.

ПОДПРОГРАММА D.

ОСУЩЕСТВЛЯЕТ ПЕРЕЗАПИСЬ ИНФОРМАЦИИ НА СМЕННУЮ СТРАНИЦУ. ОБРАЩЕНИЕ К ПП ИМЕЕТ СЛЕДУЮЩИЙ ВИД:

```
CALLD,<ОБОЗНАЧЕНИЕ ДАННЫХ> ТО <НОМЕР КАНАЛА>
```

ОБОЗНАЧЕНИЕ ДАННЫХ АНАЛОГИЧНО ОПЕРАТОРУ SAVE. ЕСЛИ ДАННЫЕ НЕ ОБОЗНАЧЕНЫ (Т.Е. CALLD,ТО <НОМЕР КАНАЛА>), ТО НА СМЕННУЮ СТРАНИЦУ ПЕРЕПИСЫВАЕТСЯ ПРОГРАММА.

ПОДПРОГРАММА E.

ОСУЩЕСТВЛЯЕТ ПЕРЕЗАПИСЬ ИНФОРМАЦИИ СО СМЕННОЙ СТРАНИЦЫ. ОБРАЩЕНИЕ К ПП:

```
CALLE,<НОМЕР КАНАЛА>ТО<ОБЛАСТЬ ДЛЯ ЗАПИСИ ДАННЫХ>
```

ОБЛАСТЬ ДЛЯ ЗАПИСИ ДАННЫХ ОБОЗНАЧАЕТСЯ АНАЛОГИЧНО ОПЕРАТОРУ LOAD. ДЛЯ ЗАПИСИ В СВОБОДНУЮ ЗОНУ ОЗУ ОБРАЩЕНИЕ К ПП:

```
CALLE,<НОМЕР КАНАЛА>
```

ПОДПРОГРАММА F.

ОСУЩЕСТВЛЯЕТ ПЕРЕМОТКУ ЛЕНТЫ ВПЕРЕД. ВРЕМЯ ПЕРЕМОТКИ МОЖЕТ БЫТЬ ОГРАНИЧЕНО УКАЗАНИЕМ ВРЕМЕНИ В СЕКУНДАХ, АНАЛОГИЧНО REW.

ПОДПРОГРАММА G.

ВСЕ ПРОПИСНЫЕ РУССКИЕ БУКВЫ В ТЕКСТОВОМ ФАЙЛЕ (КАНАЛ #2), СТОЯЩИЕ НЕ ПОСЛЕ ТОЧКИ (ПРОБЕЛЫ И СИМВОЛЫ <ПС> ИГНОРИРУЮТСЯ), ЗАМЕНЯЮТСЯ НА СООТВЕТСТВУЮЩИЕ ИМ СТРОЧНЫЕ.

// Конец файла k35b_index-3.

// ПОЯСНЕНИЯ:

// 1. В исходном тексте в файле с индексом 3 на магнитной ленте "k35, сторона Б" обозначения клавиш в скобках < > отсутствовали. Вместо обозначений в скобках присутствовало по два пробела. Можно предположить, что подразумевалось, что пользователь сам вписывает в распечатку нужные обозначения клавиш редактирования дисплея 15ИЭ-00-013, так как в принтере для них нет стандартных символов. В данной распечатке такие символы я (Sinus) вставил внутрь угловых скобок < > по результатам экспериментов с редактированием программой "Форматер" в имитаторе ДЗ-28 (версия 16.02.2023).

// На стр. 4 в примерах чисел не было степеней, т.е., например, вместо $+0,24 \cdot 10^{21}$ было просто $+0,24 \cdot 10$
// Степени и знак умножения (точку) я добавил вручную. И удалил много пустых строк по всему тексту.

// 2. В исходном тексте в описании перехода из режима останова ЭВМ к режиму "Диалог с Бейсиком-План" нажатием (на пульте ЭВМ) кнопок [C], [>], [M] символ > обозначает кнопку ▷. При работе в имитаторе указанной последовательности кнопок соответствует последовательность ПК-клавиш <C>, <T>, <M> в режиме View > Machine (этот режим можно выбрать в "меню" окна имитатора).

// 3. Для того чтобы в имитаторе ДЗ-28 (во всяком случае в его нынешней версии, 16.02.2023) с Бейсиком-План из файла k35b_Basic_plan_KP-192147_N-12621.txt сделать доступным "вывод на печать" (т.е. в txt-файл, в том числе листинги программ и распечатки текстов Форматером), надо выполнить следующие действия:

// а) после загрузки Бейсика-План в имитатор (как обычно, ПК-клавишами <C>, <L>), но до его запуска (клавишей <S>), следует на шаге 07681 заменить байт 0805 байтом 0010. Т.е. надо перейти в режим "В" (клавишей <V> или <W>), включить установку номера шага (клавишей <N>), набрать цифровыми ПК-клавишами 5-значное число 07681, затем набрать 4-значное число 0010 и нажать <Enter>, вернуть (клавишей <R>) ЭВМ в режим "Р", нажать <C>, и тогда уже запустить Бейсик-План клавишей <S>.

// б) включить клавишей <Home> имитацию ТПУ и включить "печать маленьких букв как в МХТИ-128С". Т.е. в режиме имитатора View > Display нажать <F1>, <P_{лат}>, и нажать "Да" в появившемся окне с вопросом о включении "mhti-128c print mode".

// По-видимому, указанная замена байта 0805 байтом 0010 нужна потому, что данная версия Бейсика-План, – 128 ФМ, – работает с печатающим устройством УВВПЧ-30-004, и в нём код 0805 служит командой печати содержимого буфера печати. А в имитаторе имитируется работа с термопечатающим устройством ТПУ; в нём аналогичной командой печати содержимого буфера печати является код 0010.

// 4. Для работы с Бейсиком-План в имитаторе ДЗ-28 следует сбросить в ноль параметры служебной строки имитируемого дисплея. Для этого достаточно в режиме имитатора View > Display нажать <F8>, <F9>, <F8>. Подробно о работе с имитатором написано в pdf "Руководство к имитатору" в архиве с самим имитатором.

Sinus 5.08.2023.