

## Основные понятия и правила написания программ.

### Структура программы.

Раздел описаний	}	PROGRAM <имя программы>;
		<раздел описаний>
Текст программы	}	BEGIN
		<операторы>
		END.

### Имя программы.

*Имя программы* может состоять из букв *английского алфавита* и может содержать *цифры*. Начинаться имя должно с *буквы* и не может содержать символы *русского алфавита* и *знаки препинания*. Допускается использовать в имени программы знак подчёркивания «  ».

### Раздел описаний.

В данном разделе описываются:

- |                |                  |
|----------------|------------------|
| 1. Модули.     | 2. Константы.    |
| 3. Типы.       | 4. Метки.        |
| 5. Переменные. | 6. Подпрограммы. |

### Модули.

*Модуль* – файл с дополнительными командами. Подключается к программе с помощью служебного слова **USES**. Пример:  
**USES CRT, GRAPH;**

### Константы.

*Константами* называются элементы данных, значения которых в процессе выполнения программы *не изменяются*.  
Описываются с помощью служебного слова **CONST**. Пример:

```
CONST C = 5;
```

### Типы.

*Типы* – это специальные конструкции языка, которые рассматриваются компилятором как *образцы* для создания других элементов программы, такие как *переменные, функции, массивы*. Также может использоваться для *переопределения* уже существующего типа данных. Описываются после служебного слова **TYPE**. Пример:

```
TYPE
  REAL = EXTENDED;
  MAS = ARRAY [1..10] OF INTEGER;
```

### Метки.

*Метки* используются для *безусловного* изменения хода выполнения программы. Описываются после служебного слова **LABEL**. Переход к метке осуществляется *оператором* **GOTO**. В программе, после метки, ставится *двоеточие* «:». Пример:

```
...
LABEL
  FIRST;
...
BEGIN
...
  FIRST:
...
  GOTO FIRST;
...
END.
```

### Переменные.

*Переменные*, в отличие от констант, могут *изменять* свои значения в процессе выполнения программы. Описываются после служебного слова **VAR**, с указанием типа переменной через двоеточие «:». Пример:

```
VAR
  C : INTEGER;
  KB, SUM : REAL;
```

## Подпрограммы.

*Подпрограммы* – это специальным образом оформленные фрагменты программы. Они служат для *расчленения* программы на ряд, во многом *независимых*, фрагментов, которые можно исполнять *неограниченное* число раз, и ссылаться на них из *разных* областей программы.

## Текст программы.

- Написание текста основной программы начинается после служебного слова **BEGIN** и заканчивается **END** с точкой.
- Любой исполнительный оператор завершается символом точка с запятой «;». Отсутствие данного знака вызывает ошибку с кодом **85**.
- Комментарии, в программе, записываются в фигурных скобках «{ }».

Пример написания программы:

```
PROGRAM MyFirstProgram;  
BEGIN  
  WRITELN ('Это моя первая программа.');
```

```
END.
```

Блок-схема алгоритма программы:

