

Структура программы на языке Паскаль

Программа – это последовательность инструкций компьютеру приводящих к конечному результату за конечное число шагов.

Программа, написанная на алгоритмическом языке, переводится на язык машинных команд:

```
program <имя>;  
<описательная часть>;  
<раздел функций и процедур>;  
begin  
<исполнительная часть>;  
end.
```

Зарезервированные слова:

- program <имя> – необязательная строка;
- begin – начало;
- end – конец.

<имя> - присваивается составителем программы (строится по правилам составления переменных).

Описательная часть программы

Все переменные, используемые в программе, должны быть описаны.

Описание начинается со служебного слова var.

```
program ff;  
var  
i,n: integer;  
x,y,z: real;  
begin;  
...
```

Список переменных от типа отделяется - “:”, одно описание от другого - “;”, список переменных перечисляется через - “,”. Если в программе используются метки, то они описываются с помощью служебного слова label. Метки могут быть числовые и символьные.

В программах на Паскале можно использовать константы, которые описываются с помощью служебного слова const.

```
const  
n=100;  
...
```

Переменная-константа (n), более в программе не описывается, ее тип определяется присвоенным ей числовым значением.

Для определения пользовательского типа данных используют служебное слово - type.

```
type
in=integer;
var
a,b: in;
...
```

С помощью служебного слова `uses` можно подключать к программе стандартные библиотечные модули. Стандартные модули объединяют функции определенного назначения и, в случае необходимости, подключаются к программе.

Например, для использования функции очистки экрана (`clrscr`), к программе подключают стандартный модуль `crt`.

```
uses
crt;
...
```

Исполнительная часть программы

Выполнение программы начинается именно с исполнительной части.

Отдельные инструкции, входящие в программу, называются операторами. Операторы отделяются один от другого – “;” .

Операторы бывают трех типов:

- пустой оператор;
- простой оператор;
- составной оператор.

Структура составного оператора:

```
begin
<оператор 1>; <оператор 2>; ...<оператор N>;
end;
```