Структура программы на Паскаль

Заголовок программы

Program <имя>**;**

Описание переменных

Var <имя переменной>,<имя переменной>:тип переменной;

Раздел операторов

begin

< оператор>;

…

< оператор>;

end.

Повторение:

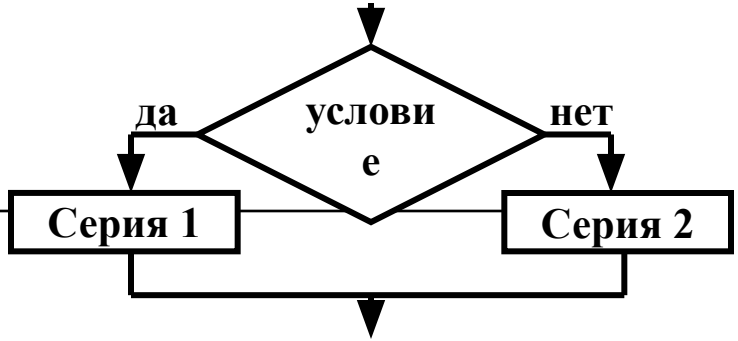
Операторы	Назначение
1. Write(список вывода);	1. Ввод данных
2. WriteLn(список вывода);	2. Присваивание
3. Read(список ввода);	3. Ввод данных и перевод строки
4. ReadLn(список ввода);	4. Вывод данных и перевод строки
5. : =	5. Вывод данных

Найти ошибки в программе

Programm Div;
Var a,b,c,d,m: integer
begin;
readln(a,b,c,d); *{Ввод}*
m=a*d; *{Числитель}*
n:=bc; *{Знаменатель}*
write(m,n); *{Вывод}*
end;
	Program Div;
	Var a,b,c,d,n,m: integer;
	begin
	readln(a,b,c,d); *{Ввод}*
	m:=a*d; *{Числитель}*
	n:=b*c; *{Знаменатель}*
	write(m,n); *{Вывод}*
	end.

Алгоритмы с ветвлением

Условный оператор

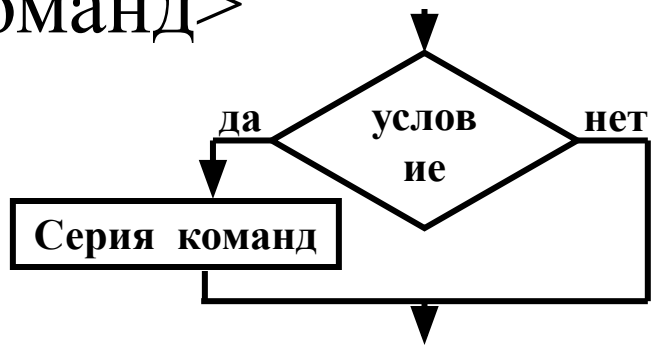


Полное ветвление:

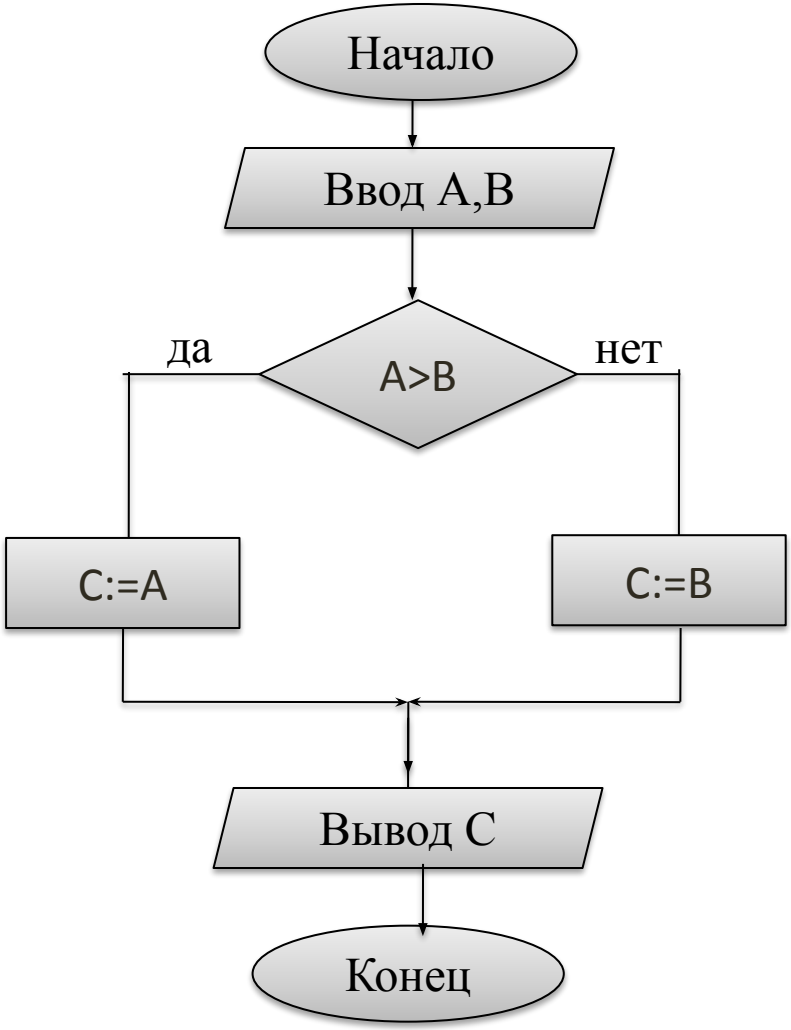
if <условие> **then** < серия команд 1 > **else**
<серия команд 2>

Неполное ветвление:

if <условие> **then** < серия команд>



Задача 1. Даны значения двух величин: выбрать из них большее.

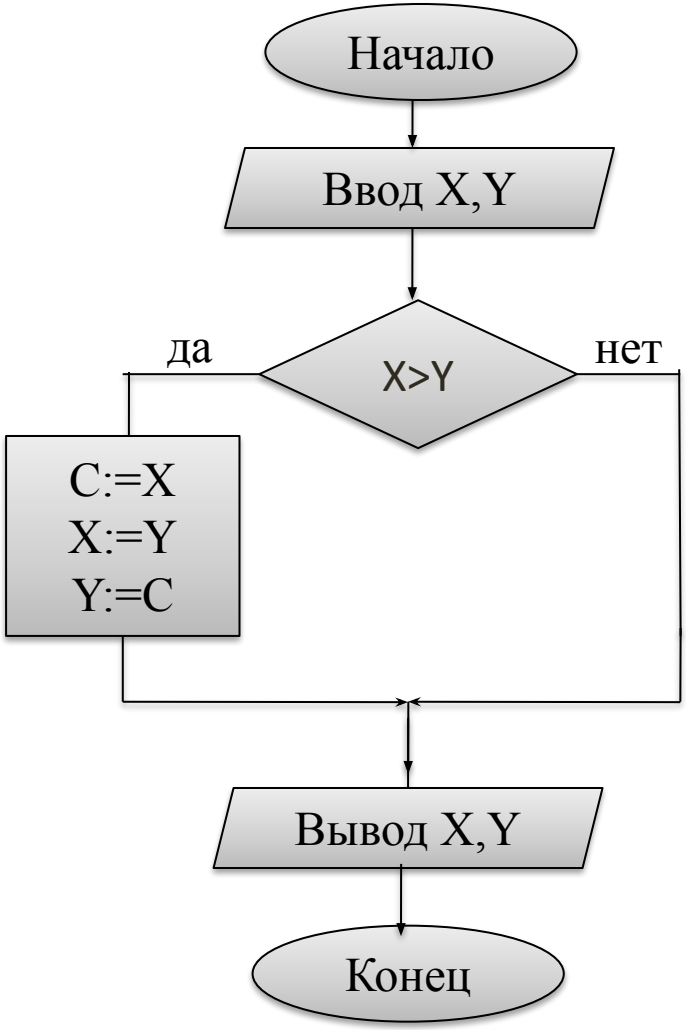


Программа:

```
Program Max2;
Var A,B,C: integer;
Begin
Write (‘Введи два числа’);
 Read (A,B);
 If A>B then C:=A  else C:=B;
 Write (‘Max = ’, C);
end.
```

Полное ветвление

Задача 2. Обменять местами (упорядочить) значения переменных X и Y по возрастанию.



Программа:

```
Program Sort;
Var X,Y,C: integer;
Begin
Write (‘Введи два числа’);
Read (X,Y);
 If X>Y then
     begin
          C:=X;
          X:=Y;
          Y:=C;
     end;
Write (X,Y);
end.
```

Begin-end скобки условного оператора

Неполное ветвление

Задача. Определите, какая задача решается следующим алгоритмом.

```
Program tri;
Var X: integer;
Begin
Write (‘Введи число’);  Read (X);
 If X=0 then write (‘Число равно нулю’) else  If X>0 then Write
 (‘Введено  положит. число’) else Write (‘Введено отрицат.
 число’) ;
 end.
```

Определите ответ при X=0