

# **Операторы управления**

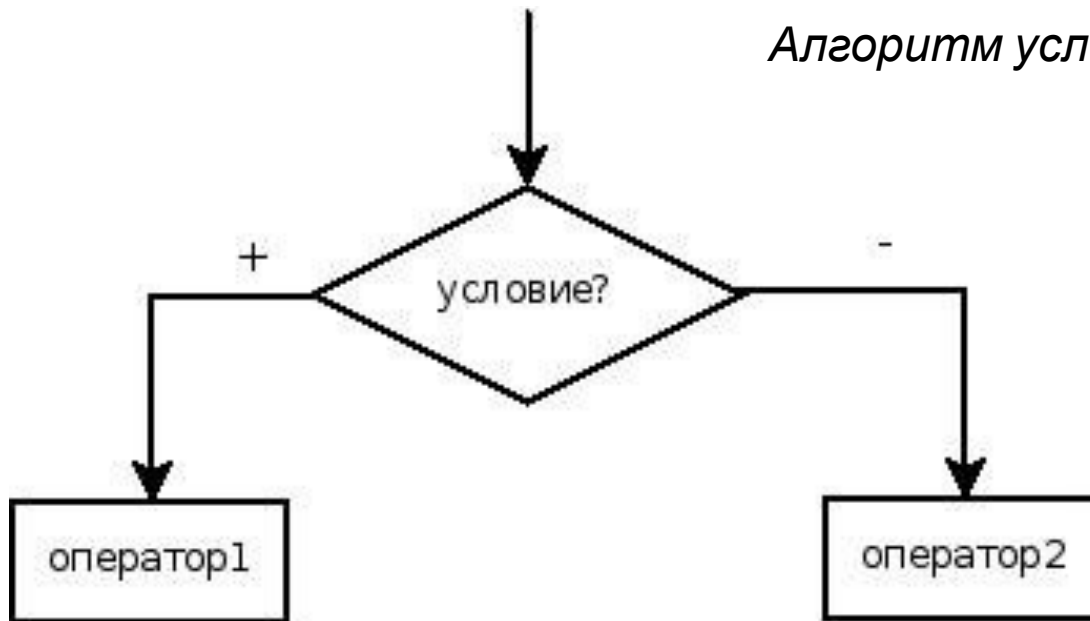
# Условный оператор

Структура:

**if (условие) оператор\_1; else оператор\_2;**

где условие - логическое выражение, переменная или константа

*Алгоритм условного оператора if... else*



# Составной оператор

начинается с символа «{» и заканчивающаяся  
символом «}»

```
{  
оператор_1;  
...  
оператор_n;  
}
```

# Варианты оператора if

```
if (условие)
{
оператор_1;
оператор_2;
...
}
else
{
оператор_1;
оператор_2;
...
}
```

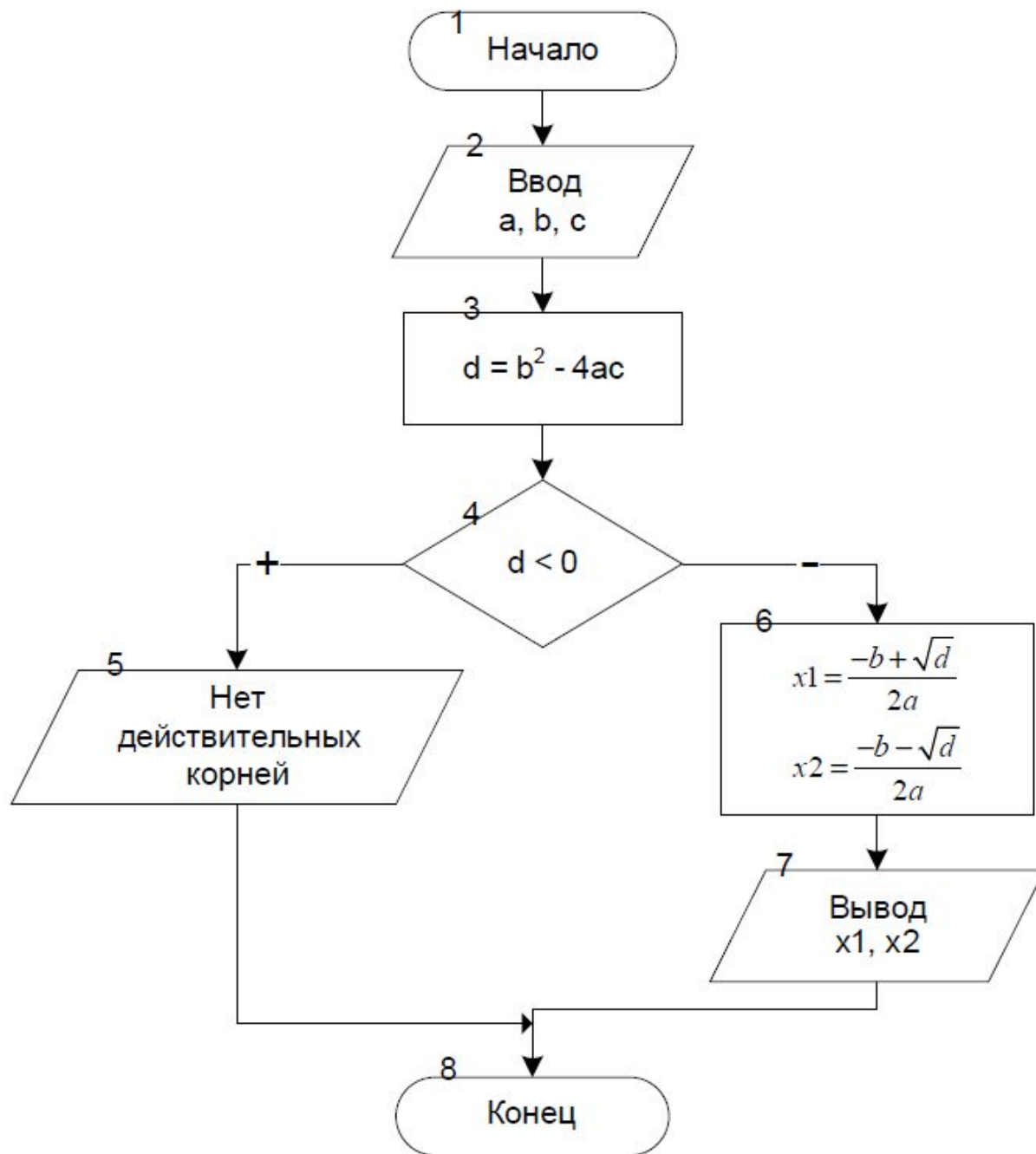
```
if (условие)
оператор;
или так
if (условие)
{
оператор_1;
оператор_2;
...
}
```

**ЗАДАЧА 3.1.** Написать программу решения квадратного уравнения  $ax^2 + bx + c = 0$ .

***Исходные данные:*** a, b и c.

***Результаты работы программы:*** x1 и x2 корни квадратного уравнения или сообщение о том, что корней нет.

***Вспомогательные переменные:***  
вещественная переменная d.



```

#include <iostream>
#include <math.h>
using namespace std;
int main()
{float a,b,c,d,x1,x2;
    //Ввод значений коэффициентов квадратного уравнения
    cout<<"a=";cin>>a;
    cout<<"b=";cin>>b;
    cout<<"c=";cin>>c;
    //Вычисление дискриминанта
    d=b*b-4*a*c;
    //Если дискриминант отрицателен,
    if (d<0)
        //то вывод сообщения, что корней нет,
        cout<<"Real roots are not present";
    Else  //{//иначе вычисление корней x1, x2
    x1=(-b+sqrt(d))/2/a;
    x2=(-b-sqrt(d))/(2*a);
        //и вывод их значений на экран
        cout<<"X1="<<x1<<"\t X2="<<x2<<"\n";
    }
    return 0;
}

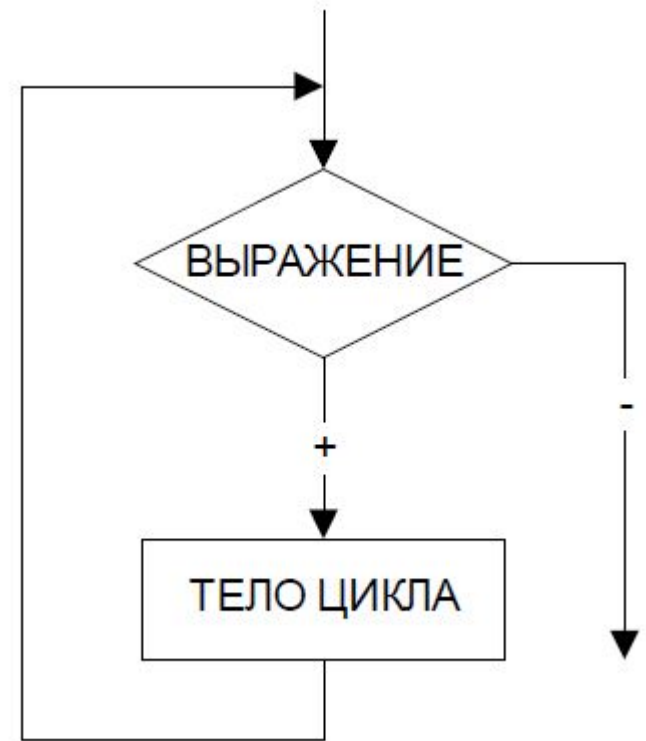
```

# Оператор цикла с предусловием

**while** (выражение) оператор;

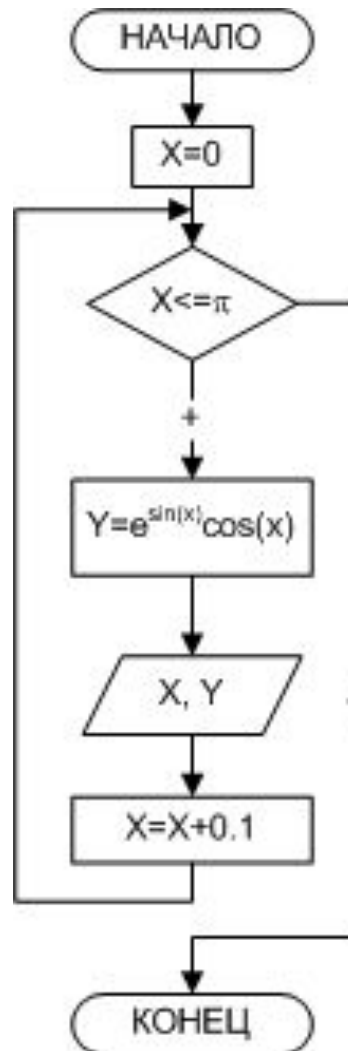
Или

```
while условие  
{  
  оператор 1;  
  оператор 2;  
  ...  
  оператор n;  
}
```





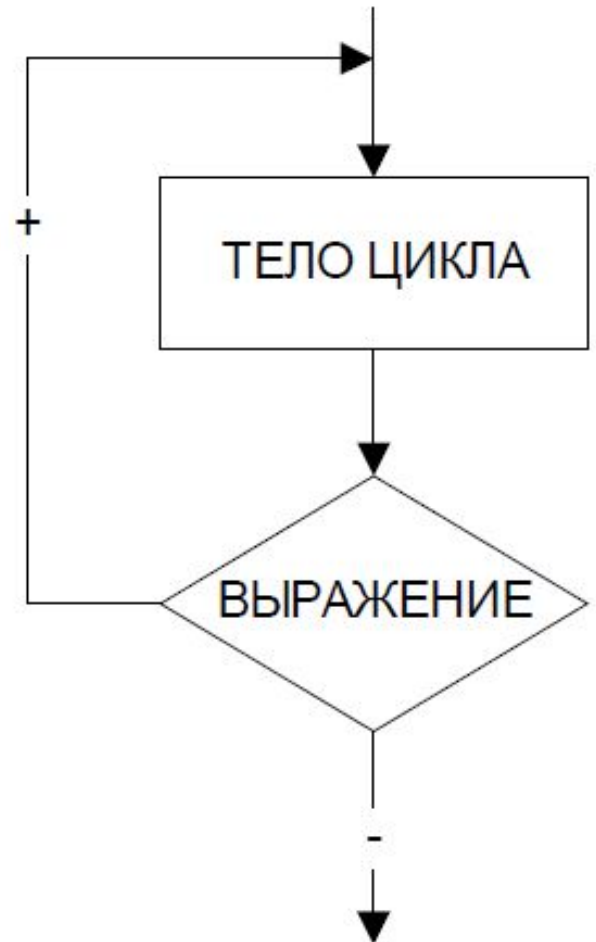
Вывести на экран таблицу значений функции  $y = e(\sin(x)) * \cos(x)$  на отрезке  $[0; \pi]$  с шагом 0.1.



```
#include <stdio.h>
#include <math.h>
#define PI 3.14159
int main()
{
float x, y;
x=0;
//Цикл с предусловием
while (x<=PI)
//Пока параметр цикла не превышает конечное
значение, выполнять тело
// цикла
{ y=exp(sin(x))*cos(x);
printf("x=%f \t y=%f \n",x,y);
x+=0.1; } //Конец цикла
}
```

# Оператор цикла с постусловием

**do** оператор **while** (выражение);  
или  
**do**  
{  
оператор\_1;  
оператор\_2;  
...  
оператор\_n;  
}  
**while** (выражение);



Вывести таблицу значений функции  
 $y = e \sin(x) \cos(x)$  на отрезке  $[0; \pi]$  с шагом 0.1

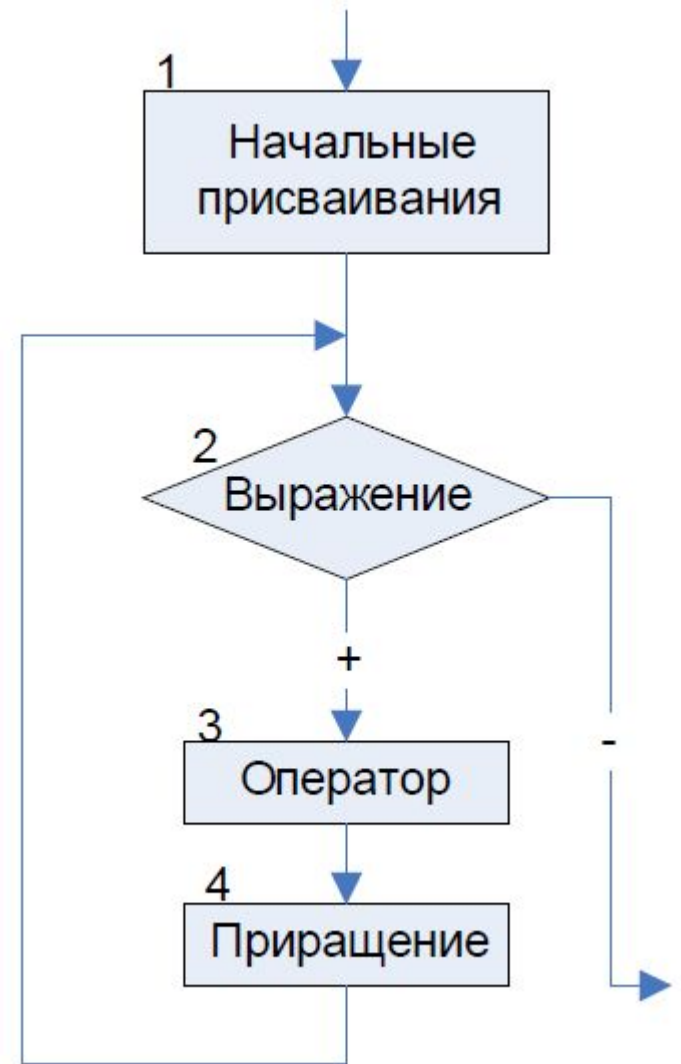
```
#include <stdio.h>
#include <math.h>
#define PI 3.14159
int main()
{
float x, y; //Описание переменных
x=0;
do //Цикл с постусловием
{//Выполнять тело цикла
y=exp(sin(x))*cos(x);
printf("x=%f \t y=%f \n",x,y);
x+=0.1; }
while(x<=PI);
return 0;}
```

# Оператор цикла с параметром

**for** (начальные\_присваивания;  
выражение; приращение)  
оператор;

или

**for** (начальные\_присваивания;  
выражение; приращение)  
{  
оператор1;  
оператор2;  
...  
}



# Решение задачи вывода таблицы значений функции $y = e \sin(x) \cos(x)$ на отрезке $[0;p]$ с шагом 0.1

```
#include <stdio.h>
#include <math.h>
#define PI 3.14159
int main()
{
    float x, y;
    //Параметру цикла присваивается начальное значение, если оно не
    // превышает конечное значение, то выполняются операторы тела
    // цикла, и значение параметра изменяется, в противном случае цикл
    заканчивается
    for (x=0;x<=PI;x+=0.1)
    {
        y=exp(sin(x))*cos(x);
        printf("x=%f \t y=%f \n",x,y);
    }
    return 0;
}
```