

Структура программы на языке C/C++

программа на языке C

```
#include <stdio.h> //.....
int main( )        //.....
{
    int a,b,c;
    printf("Введите два целых числа \n");
    scanf("%d",&a);
    scanf("%d",&b);
    c=a/b;
    printf("c=%d \n", c);
    return 0;
}
```

программа на языке C++

```
#include <iostream>
using namespace std;
int main( )        //.....
```

```

{
    int a,b,c;
    cout<<"Введите два целых числа "<<<endl;
    cin>>a>>b;
    //cin>>b;
    c=a/b;
    cout<<"c="<< c;
    return 0;
}

```

```
# include <iomanip>
```

```

cout<<setw(6)<<setprecision(3)<<"x="<<x<<"
y="<<y<<endl;

```

ЭВМ $x = 4.200y = 8.400$

Замечание 1.

```
int a,b,c;
```

Если $a=5$, $b=2$ и $c=a/b$, то $c=2$

необходимо написать так:

```
int a,b;
```

```
float c;  
c=(float)a/b;  
Получим c=2.5
```

Замечание 2.

```
float d;  
d = 3/2;    // ответ d=1  
d = 3.0/2   // ответ d=1.5  
d = 3./2    // ответ d=1.5
```

ОСНОВНЫЕ ТИПЫ ДАННЫХ

Тип данных – характеристика объекта, которая определяет 2 его основные характеристика:

1. Значения, которые может принимать объект

2. Действия, которые можно производить с объектом

название	обозначение	спецификация
целые	int	%d
вещественные	float double	%f %E %lf
СИМВОЛЬНЫЕ	char	%c

строка	char s[10] string(только C++)	%s
логический	bool(только C++)	

Арифметические операции

+ − *

/ (a/b)

% (11%3 =2)

++ инкремент

-- декремент

a = a + 1 ;

a++ ;

++a ;

Приоритет операций

1. ++ --

2. −

3. * / %

4. + −

Операции отношения

$<$, $<=$, $>$, $>=$, $==$, $!=$

Пример $a < b$, $d == c$

Операция присваивания

$a = 2 ;$

$a = b = c = d = 2 ;$

Составные операции

$+=$, $-=$, $*=$, $/=$, $\%=$

Примеры

$s+=5;$

$s=s+5;$

$k*=2;$

$k=k*2;$

Логические операции

$\&\&$ $-$ И

$||$ $-$ ИЛИ

! – НЕ

Математические функции

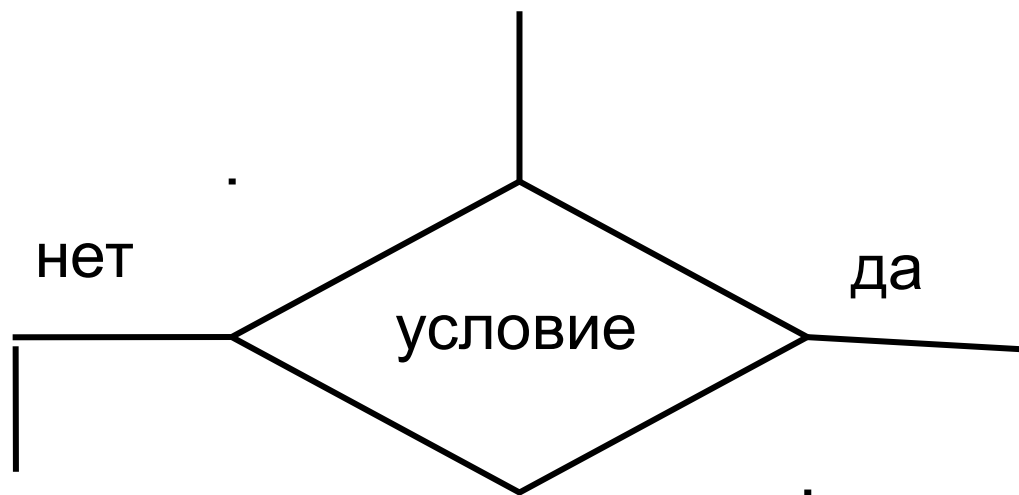
```
#include<math.h>
```

sin(x), cos(x), tan(x), sqrt(x), exp(x), log(x),
log10(x), abs(i), fabs(x),

pow(a,x) – a^x

корень четвертой степени из m – pow(m,1./4)

Ветвления



Краткая форма

```
if (условие) оператор;
```

Полная форма

```
if (условие) оператор1 ;  
    else оператор2 ;
```

Пример 1.

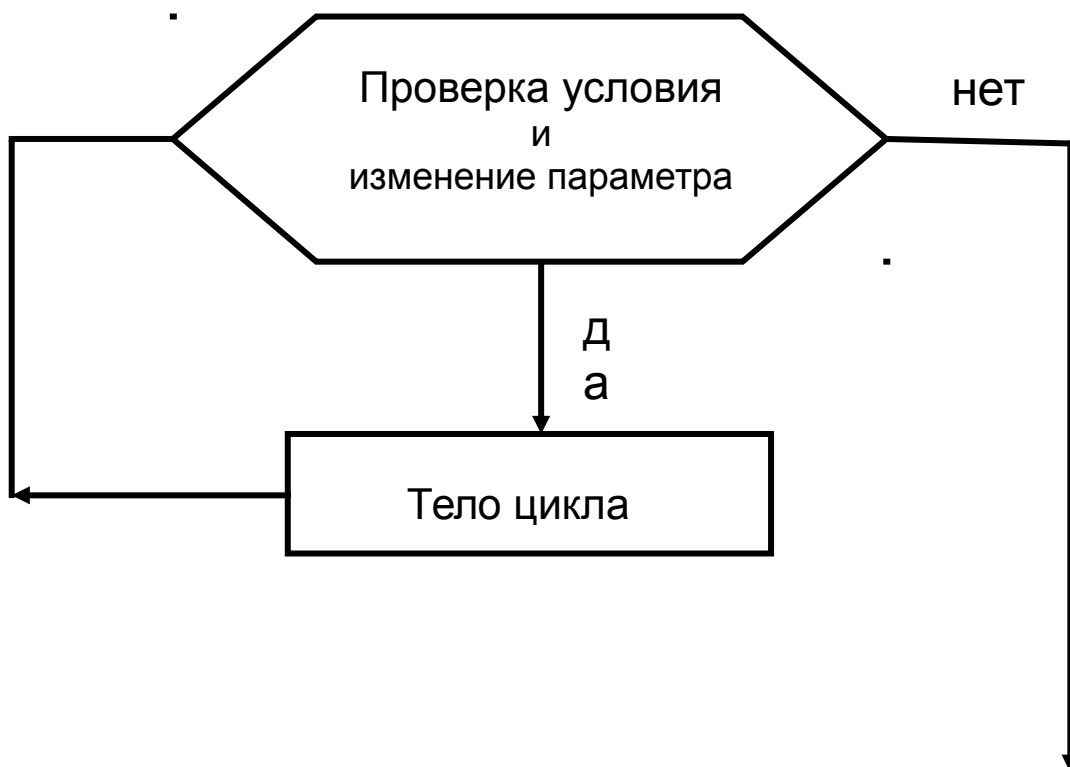
Найти max из x1 и x2

```
int main()  
{
```

```
float max, x1, x2;  
cout<<"Введи x1=";  
cin>>x1;  
cout<<"Введи x2=";  
cin>>x2;  
if (x1>=x2) max=x1;  
    else max=x2;  
cout<<" max="<<max;  
return 0;
```

Операторы цикла

for, while, do..while




```
for(часть1;часть2;часть3)
    {
        тело цикла ;
    }
```

Примеры

```
1. for( int i=1;i<=5;i++) //i=i+1
    cout<<i<<" ";
```

ЭВМ: 1 2 3 4 5

Замечание.

```
for(i=1;i<=5;i++) ;
    cout<<i<<endl;
```

Результат:

6

!!

Пример 1

```
int main()
{
    int N=5 ;
```

```

float x, y;
for(int i=1; i<=N; i++)
    {
        cout<<"Введи x";
        cin>>x ;
        y=x*x ;
        cout<<"x="<<x<<" y="<<y<<endl;
    }
return 0;}

```

Пример 2. Табулирование функции

Протабулировать функцию $y=x*x$ на интервале $[-1;7]$ в 5 равностоящих точках

```

int n=5 ;
float x, y, xn, xk, h;
h=(xk-xn)/(n-1);
cout<<"Введи xn";
cin>>xn ;
cout<<"Введи xk";
cin>>xk ;
for(int i=1; i<=n; i++)
    {

```

```
        x=xn+(i-1)*h;  
        y=x*x ;  
        cout<<"x= " <<x<<"  y=" <<y<<endl;  
    }  
return 0;}
```

На ЭВМ

x=-1	y=1
x=1	y=1
x=3	y=9
x=5	y=25
x=7	y=49
x=	y=

Оператор цикла while

```
while(условие)
```

```
{
```

```
    тело цикла
```

```
}
```

```
# include <iostream>
```

```
int main()
```

```
{  
  int j=0;  
  while(j<10)  
    {  
      cout<<"j="<<j<<endl;  
      j++;  
    }  
  return 0;  
}
```

Оператор цикла do - while

```
do  
{  
  тело цикла  
}  
while(условие);
```

```
# include <iostream>  
int main()
```

```
{  
  int j=0;  
  do  
    {  
      cout<<"j="<<j<<endl;  
      j++;  
    }  
  while(j<10);  
  return 0;  
}
```