

**Реализация циклических  
алгоритмов в языке Си Шарп.  
Примеры.**

## Цикл с параметром («для»)

Цикл с параметром в Си# имеет формат:

```
for (<НП-инициализация>; <У продолжения>; <ИП>)  
{  
    <ТЦ>  
}
```

## Оператор **for** выполняется следующим образом:

- Сначала вычисляется начальное значение параметра цикла.
- Затем, пока условие получает значение `true`, выполняется тело цикла и затем происходит новое вычисление параметра цикла.
- Когда условие получает значение `false` управление передается вне цикла.

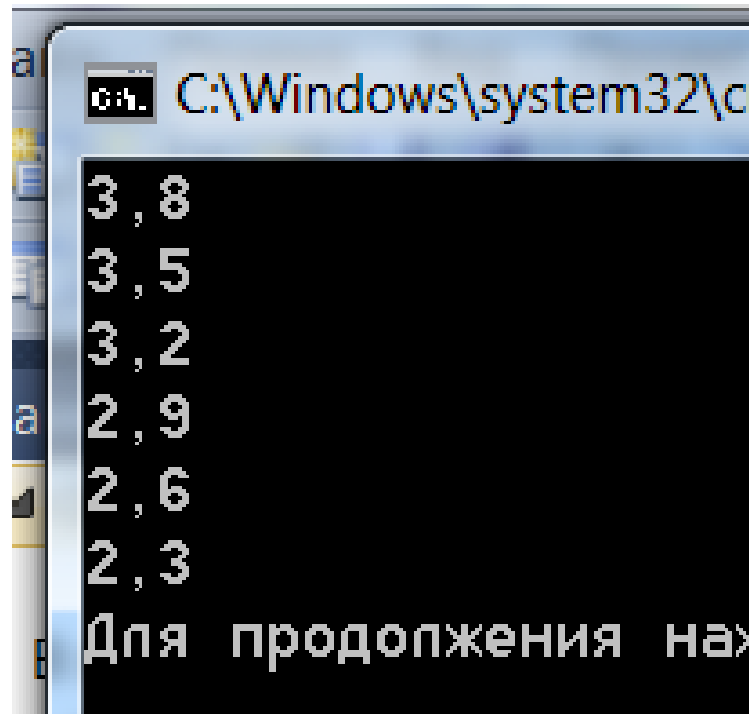
## Пример 1.

```
int i;  
for (i = 1; i <= 5; i++)  
Console.WriteLine(i); //тело цикла -  
всего один оператор и скобки {} не  
нужны  
/*напечатается  
1  
2  
3  
4  
5  
*/
```

## Пример 2

```
for (double j = 3.2 + 0.6; j > 2.1; j = j - 0.3)  
    Console.WriteLine(j);
```

Результат:



```
C:\Windows\system32\cmd.exe  
3,8  
3,5  
3,2  
2,9  
2,6  
2,3  
Для продолжения нажми
```

Инструкция **break** прекращает выполнение цикла и управление передаётся за цикл.

Инструкция **continue** выполняет пропуск операторов в теле цикла, идущих ниже, осуществляется вычисление следующего параметра цикла и дальнейшее повторение цикла.

Пример 3:

```
double i;  
for (i = 3.8; i > 2.1; i = i - 0.3)  
{  
Console.WriteLine(i);  
if (i == 3.5) break;  
}  
Console.WriteLine(i);
```

**/\* напечатается**

**3.8**

**3.5**

**3.5 \*/**

Пример 4

```
double i;
```

```
for ( i = 2; i < 7.2; i += 0.7)
{
    if (i < 4.5) continue;
    Console.WriteLine(i);
}
```

```
Console.WriteLine(i);
```

```
/* напечатается
```

```
4.8
```

```
5.5
```

```
6.2
```

```
6.9
```

```
7.6      */
```



# Циклы с предусловием («пока») и постусловием («до»)

*Цикл «пока»* в языке Си Шарп имеет  
следующий *формат*:

**While** (<*условие продолжения*>)

{

<Тело\_Цикла>

}

Пример:

```
int i = 0;
```

```
while (i < 5)
```

```
{
```

```
    Console.WriteLine(i);
```

```
    i++;
```

```
}
```

```
/* напечатается
```

```
0
```

```
1
```

```
2
```

```
3
```

```
4
```

```
*/
```

Цикл может выполняться «вечно» -  
«зацикливание»:

```
while (true)
{
    Console.WriteLine("Вечный цикл");
}
```

**Пример1.** Вычислить сумму  $S=1+2+3+4+\dots$  . Вычисления прекратить, как только значение  $S$  станет больше заданного числа  $x$  ( $x$  вводится с клавиатуры). Дополнительно определить, сколько слагаемых просуммировано.

Тело цикла имеет вид  **$S:=S+k$** .

До начала цикла следует выполнить начальное присваивание  **$k:=1$** , а перед каждым новым повторением цикла изменять ее по закону  **$k:=k+1$** .

Условие окончания цикла:  **$S>x$** , а соответственно условие продолжения -  **$S\leq x$**  (взаимно противоположное условие):

```
double x; string c; int s, k;
```

```
Console.Write("Введите x=");
```

```
c = Console.ReadLine();
```

```
x = Convert.ToDouble(c);
```

```
s = 0; //сумму всегда обнуляем перед циклом
```

```
k = 1; //первое слагаемое
```

```
while (s <= x)
```

```
{
```

```
    s = s+k;
```

```
    k++;
```

```
}
```

```
Console.WriteLine(" сумма равна {0}, число  
слагаемых {1} ", s, k-1);
```

Введите  $x=5$

сумма равна 6, число слагаемых 3

Введите  $x=80$

сумма равна 91, число слагаемых 13

**Пример 2.** Составить программу, подсчитывающую сумму цифр некоторого натурального числа  $N$ .

Количество цифр в числе неизвестно, поэтому в теле цикла мы будем выполнять следующие действия:

- 1. выделять** из числа крайнюю правую цифру при помощи операции  $c = N \% 10$  ;
- 2. добавлять** её к сумме:  $S=S+c$ ;
- 3. «отбрасывать»** эту цифру при помощи оператора  $N=N / 10$ .

Продолжать этот процесс будем до тех пор, пока число  $N$  не станет равным 0.

```
int N,c,sum; string s;
    Console.Write("Введите целое положительное число N=");
    s = Console.ReadLine();
    N = Convert.ToInt32(s);

    sum = 0;
    while (N != 0)
    {
        c = N % 10;
        sum = sum + c;
        N = N / 10;
    } ;

    Console.WriteLine("Сумма цифр равна {0} ", sum);
```



Введите целое положительное число  $N=345$

Сумма цифр равна 12

Для продолжения нажмите любую клавишу . . .

Введите целое положительное число  $N=7235$

Сумма цифр равна 17

Для продолжения нажмите любую клавишу . . .

*Цикл «до»* в языке Си Шарп имеет следующий *формат*:

**do**

{

    <Тело\_цикла>

}

**while** (<условие продолжения>;

*Тело цикла повторяется до тех пор, пока условие истинно. Как только оно станет ложным – цикл закончится.*

**В отличии от цикла «пока» в этом цикле тело цикла выполняется хотя бы один раз всегда.**

Пример программы, которая не завершит работу, пока из клавиатуры не введут число 5:

```
int number;
do
{
    Console.WriteLine("Введите число 5");
    number = Convert.ToInt32(Console.ReadLine());
}
while (number != 5);
```

**Пример:** Дано целое число  $N > 0$ . С помощью операций деления нацело и взятия остатка от деления определить, имеются ли в записи числа  $N$  нечетные цифры.

```
int N; string s; bool f;
    Console.WriteLine("Введите целое положительное число N=");
    s = Console.ReadLine();
    N = Convert.ToInt32(s);
    int n = N; //сохраним исходное значение

    f = false;
    do
    {
        if (N % 2 == 1) f = true;
        N = N / 10;
    }
    while (N != 0 && !f);

    if (f) Console.WriteLine("Нечётные цифры в числе {0} есть", n);
    else Console.WriteLine("Нечётных цифр в числе {0} нет", n);
        Console.ReadKey();
```

Введите целое положительное число  $N=468$   
Нечётных цифр в числе 468 нет

Введите целое положительное число  $N=68374$   
Нечётные цифры в числе 68374 есть  
Для продолжения нажмите любую клавишу . . .