

Технологии программирования

Доц. каф. «Медиаменеджмента и медиапроизводства» Евич
Л.Н.

Лекция 4. Циклы в C++.

Цикл — многократное повторение одного и того же участка программы.

Цикл **for**

Цикл **for** как правило используется в тех случаях, когда можно точно определить необходимое количество повторов.

```
for (/*инициализирующее выражение */ ; /* условное выражение */;  
/* модифицирующее выражение */ )  
{  
    /*один оператор или блок операторов*/;  
}
```



Лекция 4. Циклы в C++.

```
for (/*инициализирующее выражение */ ; /* условное выражение */ ;  
/* модифицирующее выражение */ )  
{  
    /*один оператор или блок операторов*/;  
}
```

При работе цикла **for** первым выполняется инициализирующее выражение, в котором обычно устанавливается счетчик цикла, то есть переменной задается начальное, например, ***i=1***. Это происходит только один раз перед запуском цикла. Затем анализируется условное выражение, которое также называется условием прекращения цикла. Пока оно равно **true**, цикл не прекращается. Например, ***i<10***. Каждый раз после всех строк тела цикла выполняется модифицирующее выражение, в котором происходит изменение счетчика цикла. Например, ***i++***. Как только проверка условного выражения даст результат **false**, все строки тела цикла и модифицирующее выражение будут пропущены и управление будет передано первому выражению, следующему за телом цикла.



Лекция 4. Циклы в C++.

Пример

```
#include <iostream>;
#include <stdio.h>;
using namespace std;
void main() {
    int sum = 0, i;
    for (i=1; i<10; i++) sum+=i;
    cout << sum << endl;
    getchar();
}
```



Лекция 4. Циклы в C++.

В C++ допускается объявление переменных прямо в строке инициализации цикла for. В этом случае, предыдущий пример программы примет вид

```
#include <iostream>;
#include <stdio.h>;
using namespace std;
void main() {
    int sum = 0;
    for (int i=1; i<10; i++) sum+=i;
    cout << "sum=" << sum << endl;
    getchar();
}
```



Лекция 4. Циклы в C++.

Обращение к переменной `i`, объявленной в цикле, вне цикла приведет к ошибке (область видимости переменной ограничивается циклом).

Пример

```
#include <iostream>;
#include <stdio.h>;
using namespace std;
void main() {
    int sum = 0;
    for (int i=1; i<10; i++) sum+=i;
    cout << "sum=" << sum << "i=" << i << endl; //в этой
                                                //строке ошибка
    getchar();
}
```



Лекция 4. Циклы в C++.

Вложенные циклы for

Синтаксис вложенных циклов **for**

//Внешний цикл

```
for (/*инициализирующее выражение */ ; /* условное выражение */;  
     /* модифицирующее выражение */ )  
{  
    /*один оператор или блок операторов*/;  
    // Внутренний цикл  
    for (/*инициализирующее выражение */ ; /* условное выражение */;  
         /* модифицирующее выражение */ )  
    {  
        /*один оператор или блок операторов*/;  
    }  
}
```



Лекция 4. Циклы в C++.

Вложенные циклы for

Пример

```
#include <iostream>;
#include <stdio.h>;
using namespace std;
void main() {
    int sum = 0;
    for (int i=1; i<4; i++) {
        cout << "i=" << i << endl;
        for (int j=1; j<5; j++)
            cout << "i=" << i << "   j=" << j << endl;
    }
    getchar();
}
```

```
i=1
i=1   j=1
i=1   j=2
i=1   j=3
i=1   j=4
i=2
i=2   j=1
i=2   j=2
i=2   j=3
i=2   j=4
i=3
i=3   j=1
i=3   j=2
i=3   j=3
i=3   j=4
```



Оператор **break**

Оператор **break** (разрыв) вызывает немедленный выход из циклов, организуемых с помощью операторов **for**, **while**, **do-while**, **switch**; управление передается на оператор, следующий за законченным.

```
#include <iostream>;
#include <stdio.h>;
using namespace std;
void main() {
    int sum = 0;
    for (int i=1; i<6; i++) {
        for (int j=1; j<5; j++) {
            if (i < j) break;
            cout << "i=" << i << "\t j=" << j << endl;
            cout << "i-j=" << i-j << endl;
        }
        cout << endl;
    }
    getch();
}
```

```
i=1      j=1
i-j=0

i=2      j=1
i-j=1
i=2      j=2
i-j=0

i=3      j=1
i-j=2
i=3      j=2
i-j=1
i=3      j=3
i-j=0

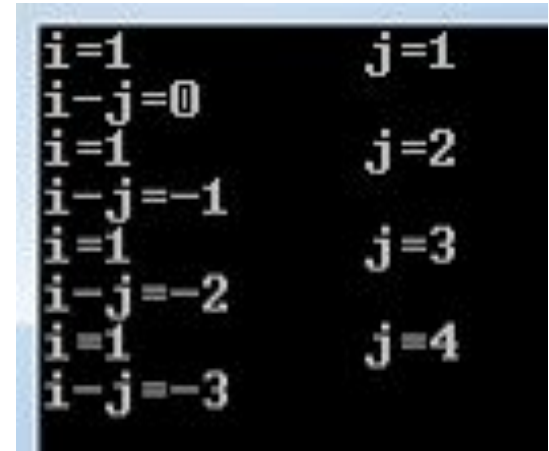
i=4      j=1
i-j=3
i=4      j=2
i-j=2
i=4      j=3
i-j=1
i=4      j=4
i-j=0

i=5      j=1
i-j=4
i=5      j=2
i-j=3
i=5      j=3
i-j=2
i=5      j=4
i-j=1
```

Оператор **break**

Пример 2

```
#include <iostream>;
#include <stdio.h>;
using namespace std;
void main() {
    int sum = 0, j;
    for (int i=1; i<6; i++) {
        for (j=1; j<5; j++) {
            cout << "i=" << i << "\t j=" << j << endl;
            cout << "i-j=" << i-j << endl;
        }
        if (i < j) break;
        cout << endl;
    }
    getch();
}
```



```
i=1      j=1
i-j=0
i=1      j=2
i-j=-1
i=1      j=3
i-j=-2
i=1      j=4
i-j=-3
```

}



Оператор continue

Оператор **continue** передает управление на следующую итерацию того цикла, в теле которого он находится.

Пример 1

```
#include <iostream>;
#include <stdio.h>;
using namespace std;
void main() {
    int sum = 0;
    for (int i=1; i<8; i++) {
        if (i % 2 == 0) continue;
        cout << "i=" << i << endl;
    }
    getchar();
}
```



```
i=1
i=3
i=5
i=7
_
```



Лекция 4. Цикл **for** в **C++**.

Задания для самостоятельной работы

26. Определите что будет выведено на экран в результате работы следующей программы.

```
#include <iostream>;
#include <stdio.h>;
using namespace std;
void main() {
    int a,b=0,i;
    for (i=1;i<=6;i++)
    {
        a=i+2;
        if (a>=5) b-=a;
        else b+=a;
    }
    cout<< "b="<<b<<endl;
    getch();
}
```

Наберите текст программы и проверьте правильность ответа.



Лекция 4. Цикл **for** в **C++**.

Задания для самостоятельной работы

27. Определите что будет выведено на экран в результате работы следующей программы.

```
#include <iostream>;
#include <stdio.h>;
using namespace std;
void main() {
    int a=0,i,j;
    for (i=12;i<=15;i++)
        for (j=4;j>=2;j--)
            if (i%j!=0) continue;
            else
                a+=i/j;
    cout<< "a="<<a<<endl;
    getchar();
}
```

Наберите текст программы и проверьте правильность ответа.



Лекция 4. Цикл `for` в `C++`.

Задания для самостоятельной работы

28. Дано целое число n ($n > -10$). Составьте программу, которая вычисляет сумму чисел от -10 до n .

29. Дано натуральное число n . Составьте программу, которая выводит на экран числа следующим образом:

```
1   2   3
4   5   6
...
n-2 n-1 n
```



Лекция 4. Цикл `for` в C++.

Задания для самостоятельной работы

30. Составьте программу, которая выводит на экран таблицу умножения в виде:

1*1=1

1*2=2

. . .

1*9=9

2*1=2

2*2=4

. . .

2*9=18

. . .

9*1=9

9*2=18

. . .

9*9=81



Лекция 4. Цикл `for` в `C++`.

Задания для самостоятельной работы

31. Дано натуральное число n . Составьте программу, которая вычисляет

сумму $\frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$

32. Даны натуральное число n и целые числа x_1, x_2, \dots, x_n . Составьте программу, которая находит среднее арифметическое тех из них, которые кратны 4-м.

33. Дано натуральное число n . Составьте программу, которая вычисляет значения элементов последовательности $a_0, a_1, a_2, \dots, a_n$, если

$$a_0 = -1, \quad a_i = ia_{i-1} + \frac{1}{i} \quad (i = 1, 2, \dots, n)$$

34. Дано натуральное число n . Составьте программу, которая вычисляет сумму

$$\frac{1}{a} + \frac{2}{a \cdot (a+1)} + \dots + \frac{n}{a \cdot (a+1) \cdot \dots \cdot (a+n)}$$

