



Циклические алгоритмы

Лекция 8

*Иллюстративный материал к
лекциям по алгоритмизации и
программированию*

Автор Саблина Н.Г.

2016 г.





Содержание

Циклический алгоритм

Оператор цикла с предусловием

while

Оператор цикла с постусловием

do while

Цикл for

Примеры

Операторы передачи

управления

Итоги

Библиографический

список

Автор





Циклический алгоритм

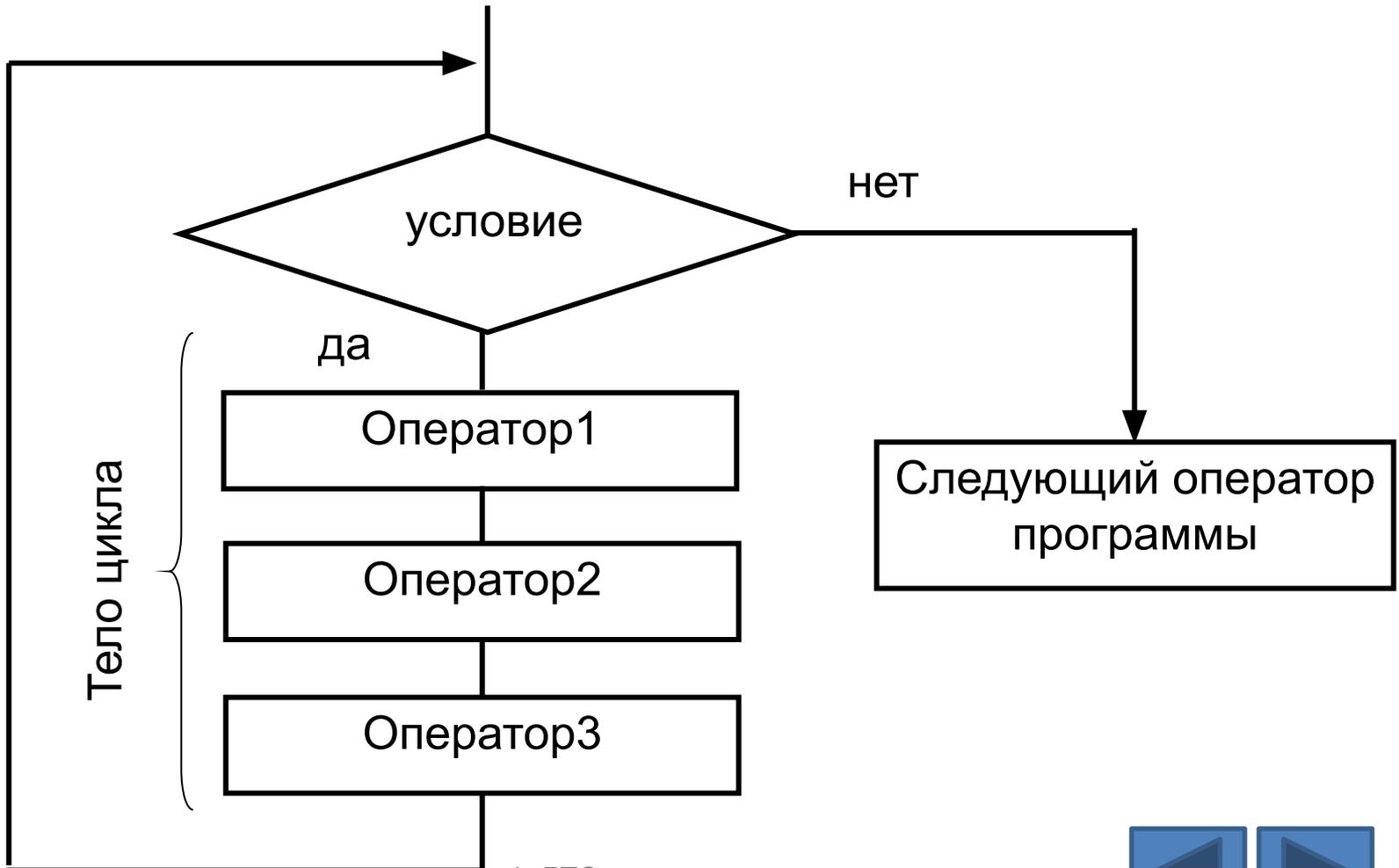
Используется для повторения некоторых действий несколько раз

Виды операторов цикла:

- - цикл с предусловием (**while**);
- - цикл с постусловием (**do - while**);
- - цикл с параметром (**for**).



Оператор цикла с предусловием (while)

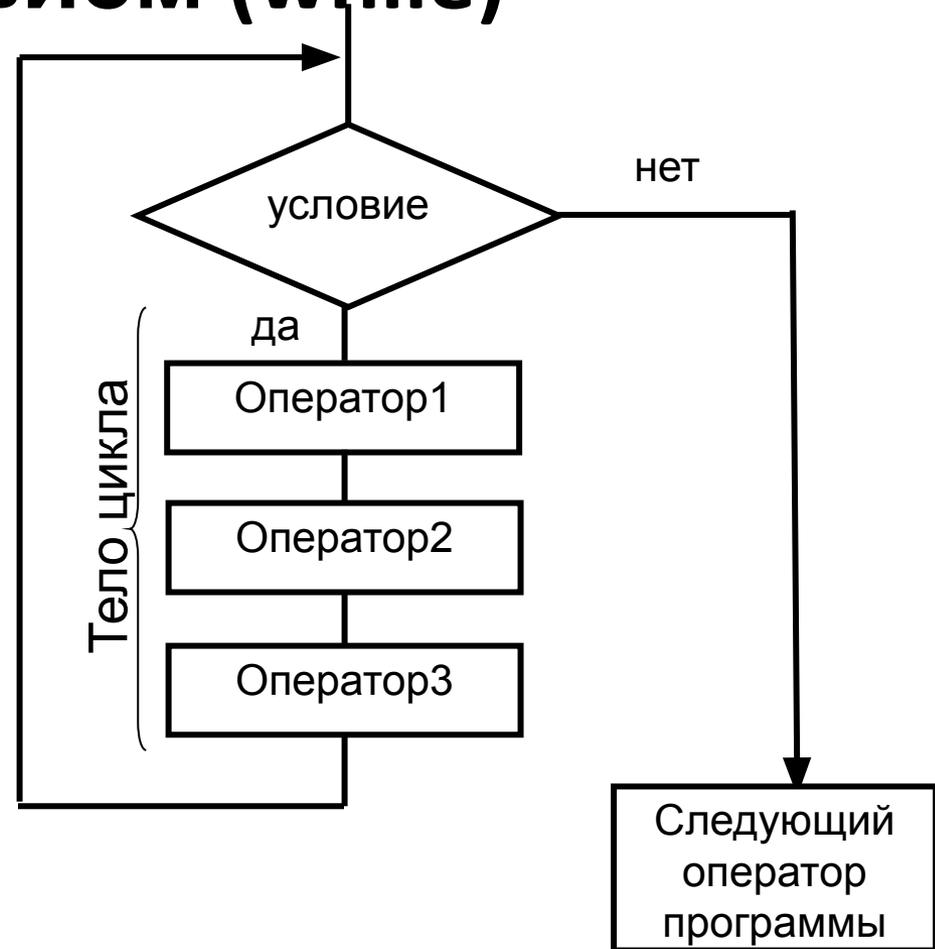


Синтаксис оператора цикла с предусловием (while)

while (условие)

```
{  
  оператор1;  
  оператор2;  
  оператор3;  
}  
Следующий оператор программы;
```

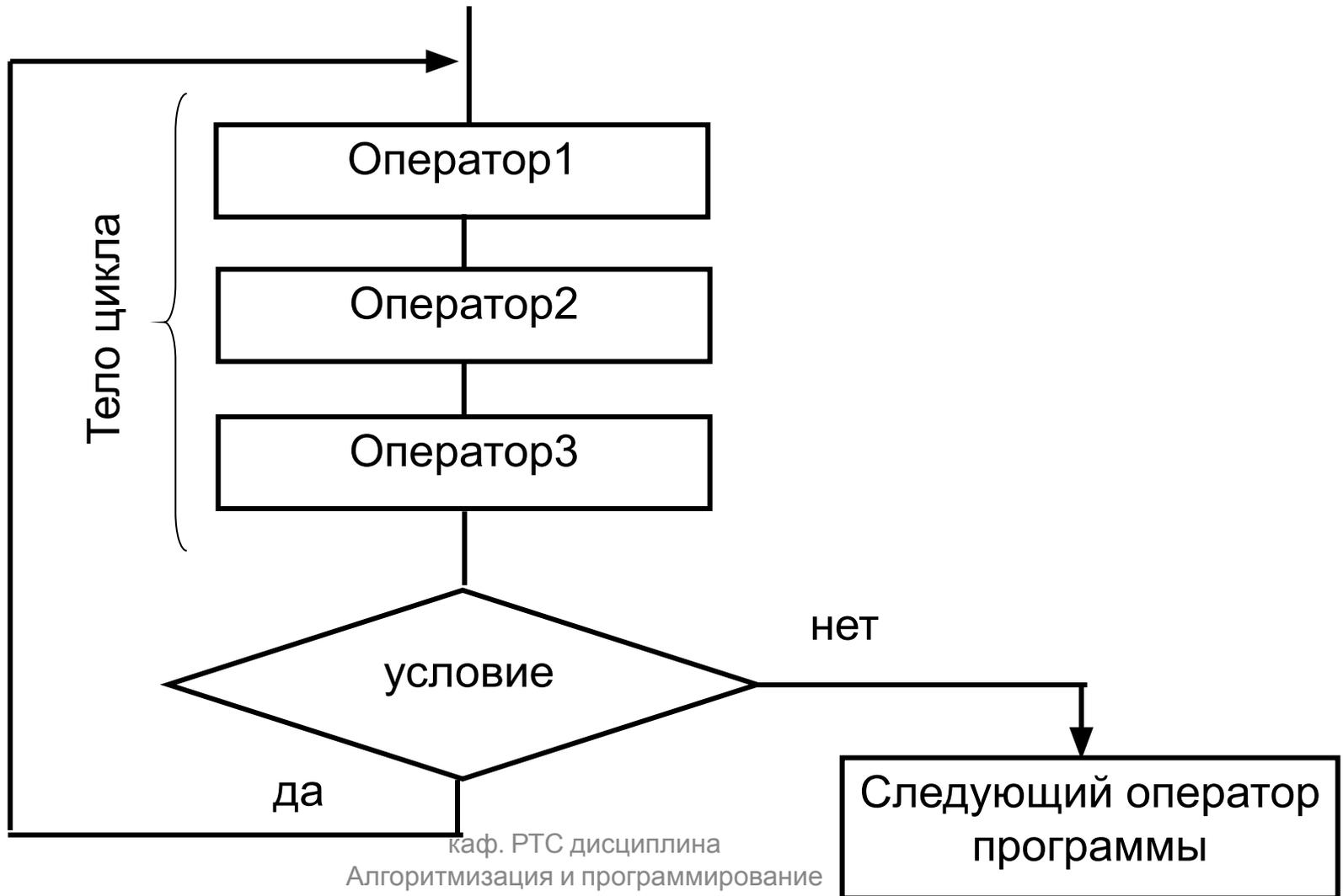
Тело цикла



Особенности оператора цикла с предусловием (while)

- Число повторений заранее неизвестно, определяется условием
- Возможен случай, когда тело цикла не выполнится ни разу

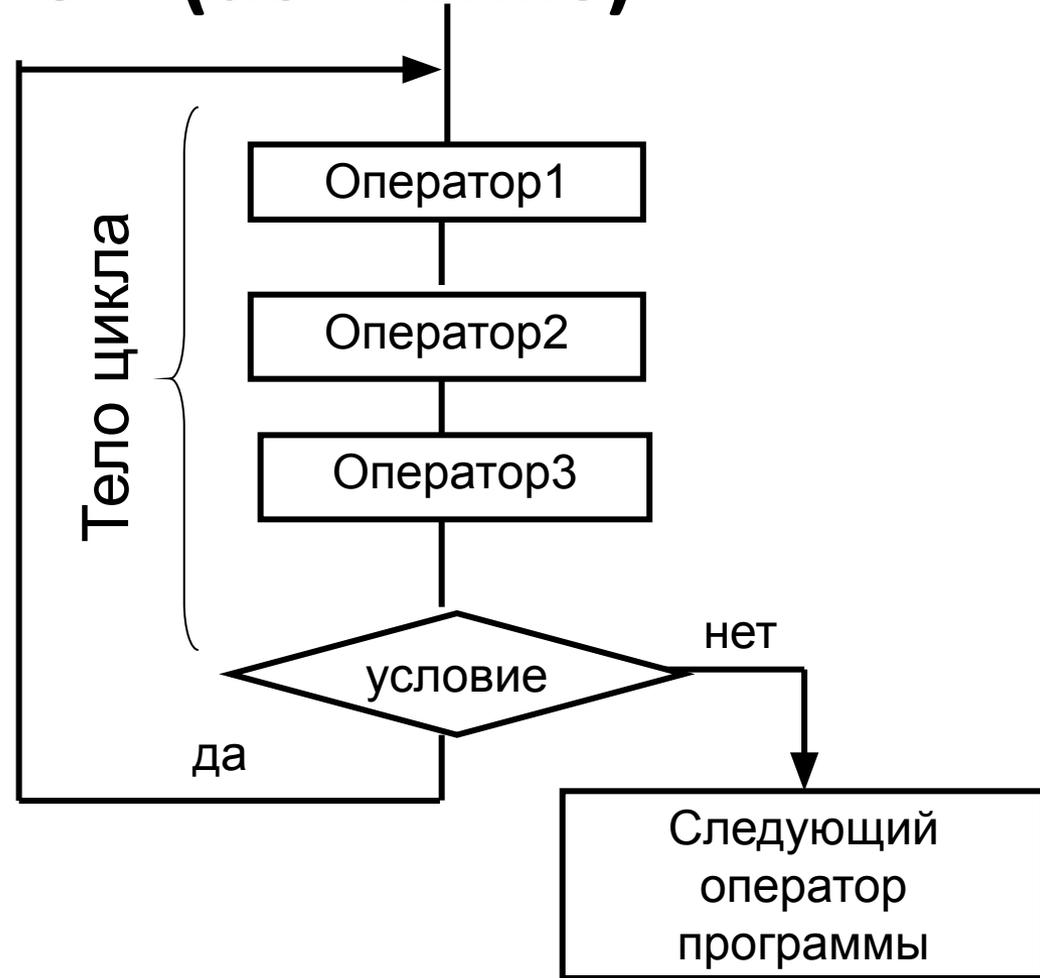
Оператор цикла с постусловием (do - while)



Синтаксис оператора цикла с постусловием (do - while)

```
do  
{  
  оператор1;  
  оператор2;  
  оператор3;  
}  
while (условие);
```

Тело цикла



Следующий оператор
программы;



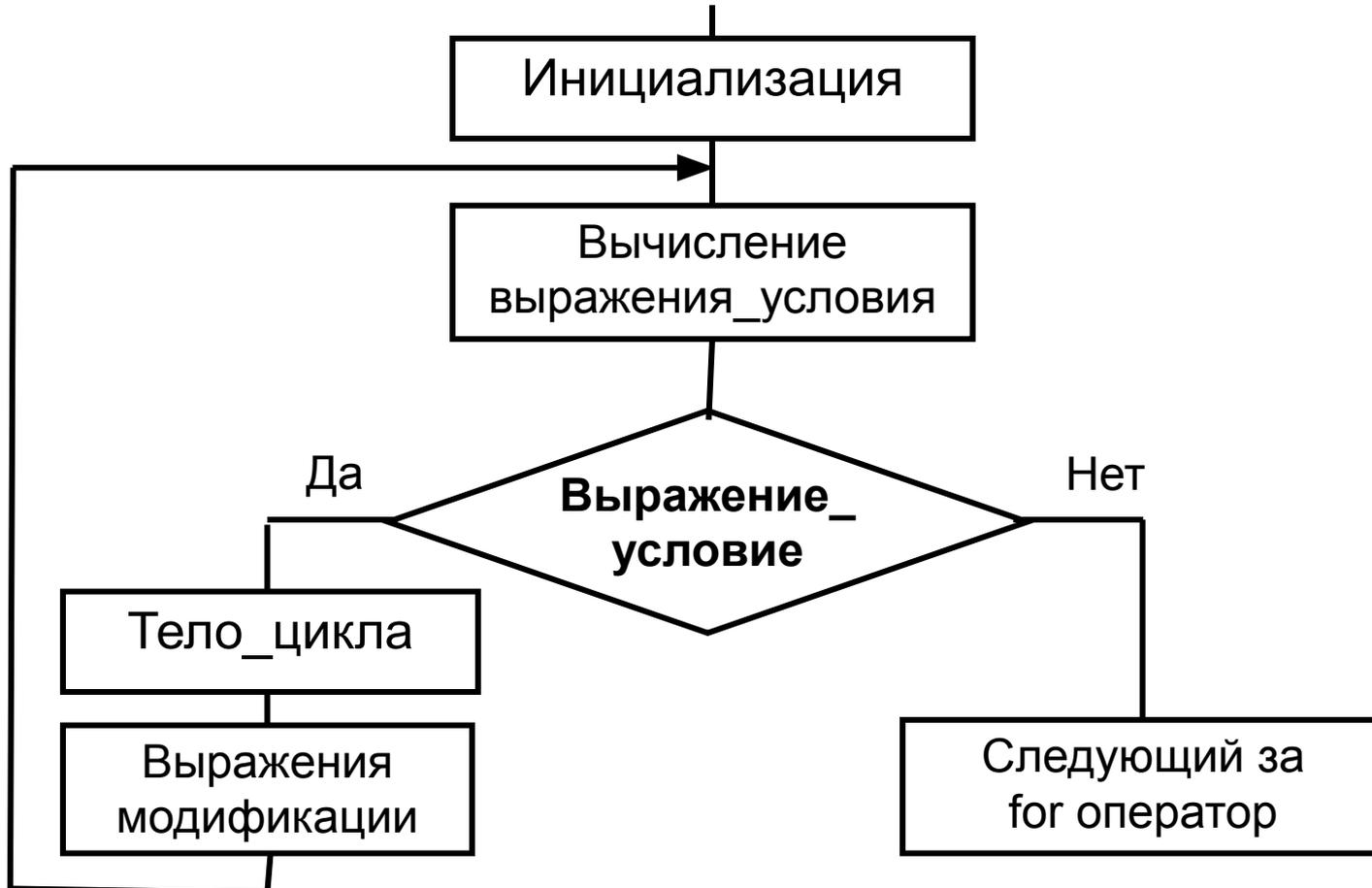
Особенности оператора цикла с постусловием (do -while)

- Число повторений заранее неизвестно, определяется условием
- Тело цикла обязательно выполнится хотя бы один раз





Обобщенна схема цикла for





Синтаксис оператора цикла for

for (инициализация; выражение_условие;
выражение_модификации) тело_цикла;

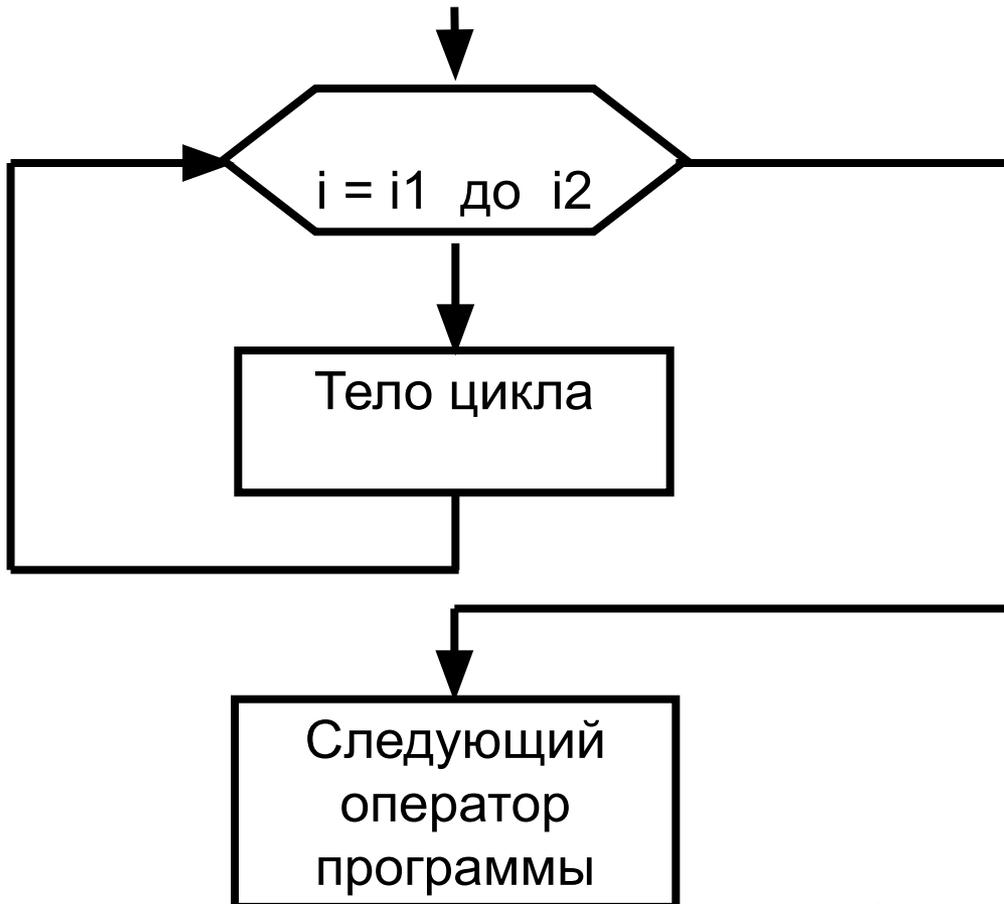
или в общем виде

for (выражение1; выражение2; выражение 3)
тело_цикла;



Цикл с параметром (1)

используется, когда число повторений заранее известно



```
for (i = i1 ; i <= i2; i++)
```

```
{
```

```
операторы тела цикла
```

```
}
```

Следующий оператор
программы ;



Примеры цикла с параметром

- a) `for (i=0; i<10; i++) printf ("%d \n", i);`
- b) `for (i=9; i>=0; i--) printf ("%d\n", i);`
- c) `for (ch='A'; ch<='Z'; ch++) printf ("%c ", ch);`
- d) `for (ch='0'; ch!='N';) scanf ("%c", &ch);`
- e) `for (int j=1, s=0; j<=k; j++) s+=j*j;`
- f) `for (int j=1,s=0; j<=k; s+=j*j++);`
- g) `for (int j=0,s=0; j<=k;) s+=++j*j;`





Бесконечные циклы

- `for (;;) printf ("Бесконечный цикл\n");`
- `for (i=1;1;i++) printf(" Бесконечный цикл\n");`
- `for (i=10;i>6;i++) printf ("Бесконечный цикл\n");`
- `while (1) printf ("Бесконечный цикл\n");`





Вложенные циклы

```
#include <stdio.h>

main()
{ int i, j;
  for ( i=1; i<10; i++)      //внешний цикл
  {for ( j=1; j<5; j++)      //вложенный цикл
    printf ( "%d*%d = %2d ", i, j, i*j );
    printf ( "\n");
  }
}
```



Пример работы программы

The screenshot shows a Turbo C++ IDE window titled "с:\. Turbo C++ IDE". The main area displays the output of a program, which is a multiplication table. The text is as follows:

```

Таблица умножения

1*1 = 1    1*2 = 2    1*3 = 3    1*4 = 4
2*1 = 2    2*2 = 4    2*3 = 6    2*4 = 8
3*1 = 3    3*2 = 6    3*3 = 9    3*4 = 12
4*1 = 4    4*2 = 8    4*3 = 12   4*4 = 16
5*1 = 5    5*2 = 10   5*3 = 15   5*4 = 20
6*1 = 6    6*2 = 12   6*3 = 18   6*4 = 24
7*1 = 7    7*2 = 14   7*3 = 21   7*4 = 28
8*1 = 8    8*2 = 16   8*3 = 24   8*4 = 32
9*1 = 9    9*2 = 18   9*3 = 27   9*4 = 36
    
```



Операторы передачи управления

К операторам передачи управления относятся:

- **goto** - оператор безусловного перехода
- **break** – оператор выхода из цикла или из оператора выбора
- **continue** – оператор перехода к следующей итерации (шагу) цикла
- **return** – оператор возврата из функции (подробно рассмотрен в главе, посвященной функциям)





Оператор *GOTO*

Оператор безусловного перехода имеет вид goto метка. Одно из полезных применений оператора goto – это выход из вложенных циклов:

```
for ()
```

```
{while ()
```

```
{ for()
```

```
{...
```

```
goto exit;
```

```
... } }
```

```
exit: printf(“Быстрый выход из вложенных циклов”);
```



Применение оператора `break`

- окончание `case` в операторе `switch`.
- немедленное окончание цикла, не связанное с проверкой обычного условия окончания цикла: происходит немедленный выход из цикла и переход к выполнению оператора, следующего за оператором цикла.

Оператор CONTINUE

- оператор `continue` в теле цикла передает управление на начало следующей итерации цикла. В циклах **while** и **do-while** - на проверку условия, в цикле **for** - на приращение.
- Этот оператор необходим, если нужно закончить текущую итерацию цикла и не выполнять оставшиеся операторы, а сразу перейти к следующей итерации цикла.

Пример «Кубы чисел»

```
#include <stdio.h>

main()
{
    int i;
    for (i=0;i<1000;i++)
        { printf ("%d - %d \n", i, i*i*i);
          if (i*i*i >= 10000) break;}
}
```



Пример «Числа, кратные семи»

```
#include <stdio.h>

main()
{
    int i;
    for(i=1;i<1000;i++)
    {
        if (i%7) continue;
        printf ("%8d", i );
    }
}
```





Итоги

Рассмотренные вопросы:

- Линейный алгоритм
- Разветвляющийся алгоритм
- Циклический алгоритм
 - Цикл с параметром
 - Цикл с предусловием
 - Цикл с постусловием



Библиографический список

- Подбельский В.В. Язык СИ++. Учебное пособие. М.: Финансы и статистика, 2003. – 560 с.
- Павловская Т.А. С/С++. Программирование на языке высокого уровня: учебник для студентов вузов, обучающихся по направлению "Информатика и вычисл. техника" СПб.: Питер, 2005. - 461 с.
- Березин Б.И. Начальный курс С и С++ / Б.И. Березин, С.Б. Березин. - М.: ДИАЛОГ-МИФИ, 2001. - 288 с
- Каширин И.Ю., Новичков В.С. От С к С++. Учебное пособие для вузов. – М.: Горячая линия – Телеком, 2005. – 334 с.



Автор:
Саблина Наталья Григорьевна

Ст. преподаватель
каф. РТС УрФУ

