

# Программирование на C++



| Минцифры  
РОССИИ

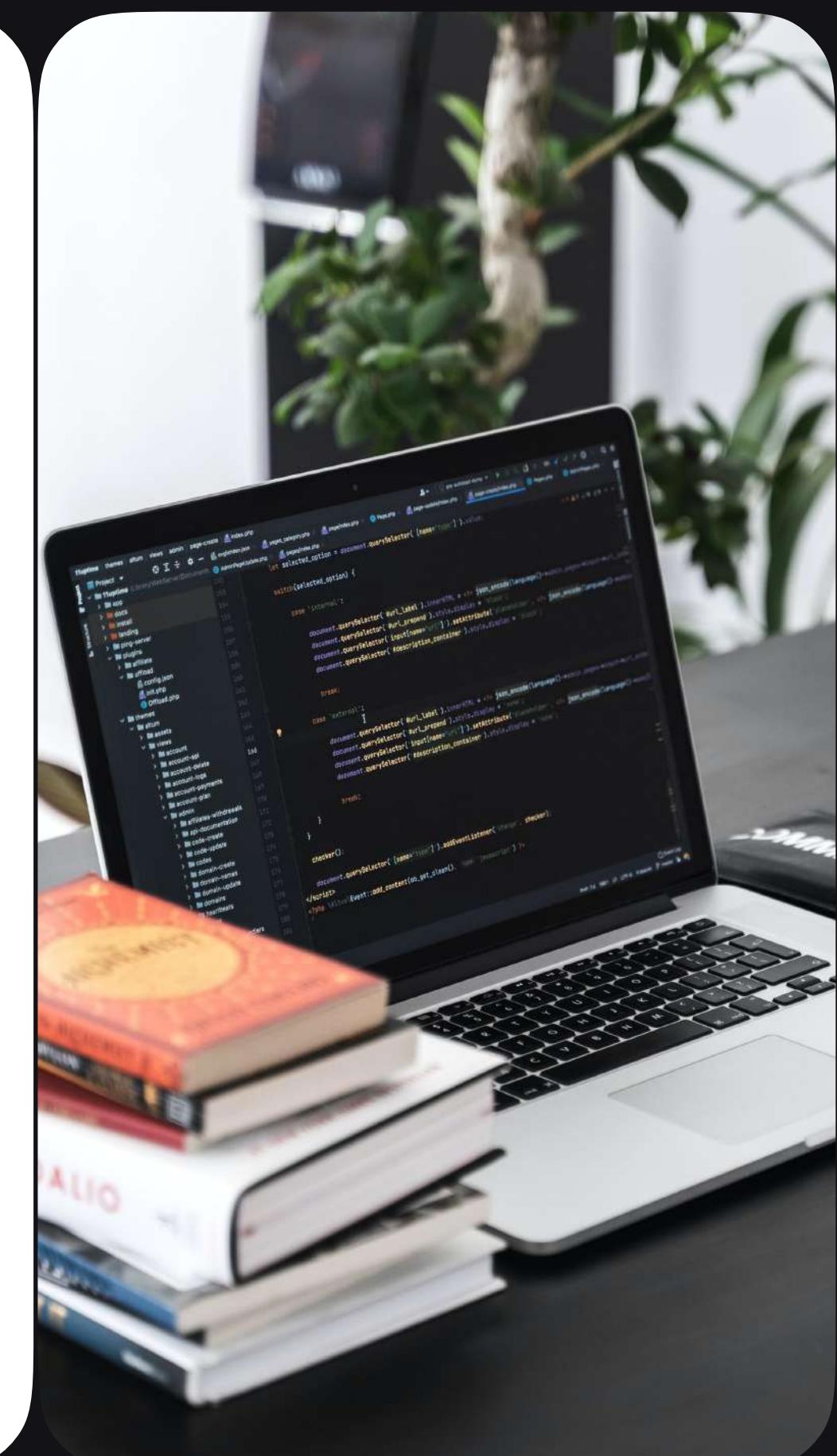
UCHI DOMA

**20.35**  
УНИВЕРСИТЕТ

Урок 8 Модуль 1

# Цикл с предусловием и постусловием

Полезные материалы



## Задача

Программа авторизации запрашивает логин и пароль до тех пор, пока не будет введена верная пара. Если введены неверные логин и пароль, программа выводит «Неверная пара логин/пароль» и запрашивает снова логин и пароль.



# Задача

Программа авторизации запрашивает логин и пароль до тех пор, пока не будет введена верная пара. Если введены неверные логин и пароль, программа выводит «Неверная пара логин/пароль» и запрашивает снова логин и пароль.

```
1 #include <stdio.h>
2 int main ( )
3 {
4     int login,password;
5     printf("Введите логин");
6     scanf("%d",&login);
7     printf("Введите пароль:");
8     scanf("%d",&password);
9     if (login==123 && password==321) printf("Доступ разрешен\n");
10    else printf("Неверная пара логин/пароль\n");
11    return 0;
12 }
```

# Задача

Нужно повторять алгоритм

```
1 #include <stdio.h>
2 int main ( )
3 {
4     int login,password;
5     printf("Введите логин");
6     scanf("%d",&login);
7     printf("Введите пароль:");
8     scanf("%d",&password);
9     if (login==123 && password==321) printf("Доступ разрешен\n");
10    else printf("Неверная пара логин/пароль\n");
11
12 }
```

# Цели урока



изучить оператор цикла  
(повторения)



отработать на практике  
составление алгоритмов  
с циклом на Си



# Цикл

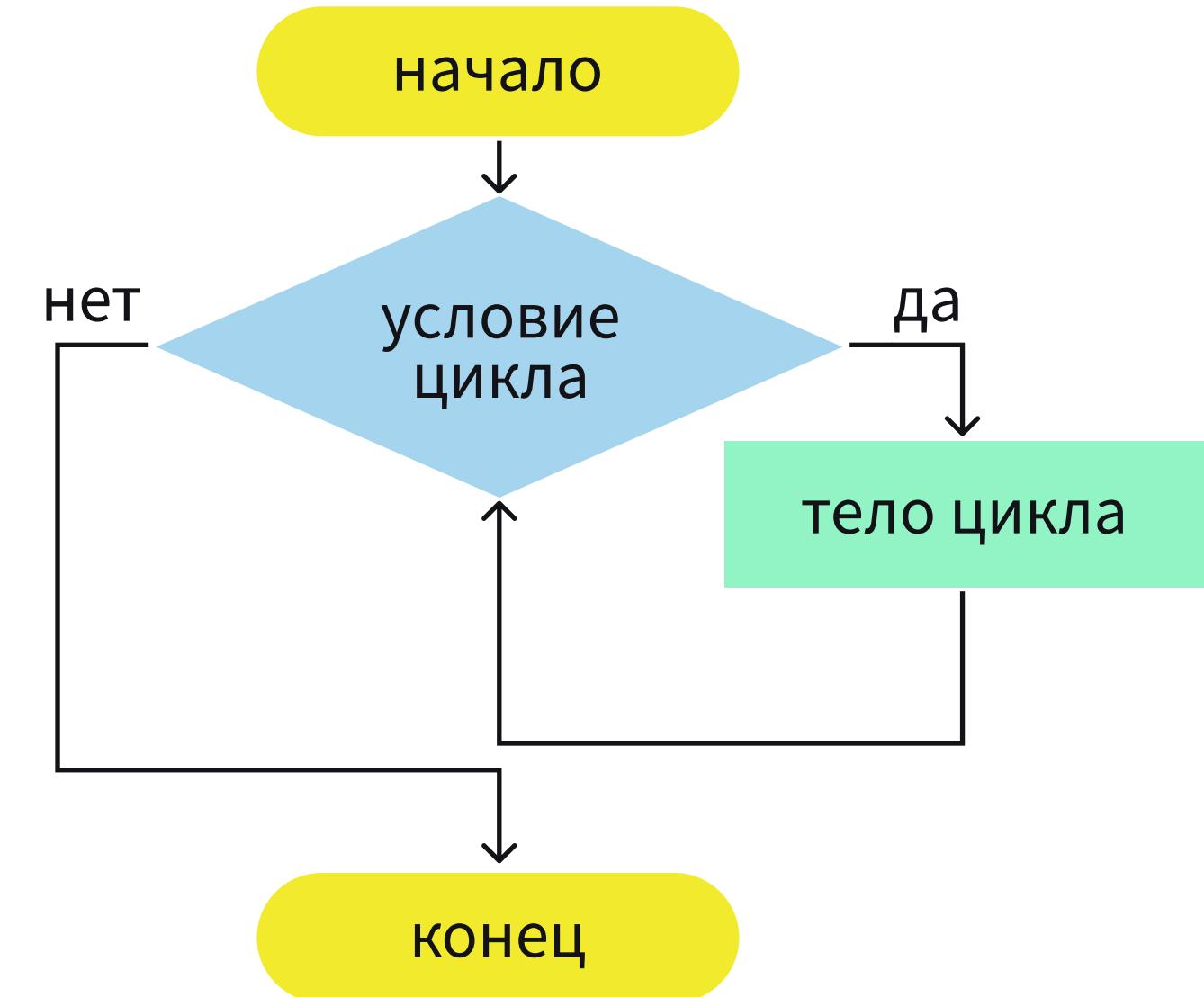
Это команда для повторения алгоритма несколько раз



Цикл выполняется до тех пор, пока блок проверки условия возвращает истинное значение.



Тело цикла содержит последовательность операций, которая выполняется в случае истинного условия повторения цикла.



# Цикл с предусловием

Если условие выполняется (выражение, проверяющее условие, не равно нулю), то выполняется тело цикла, заключенный в фигурные скобки, затем условие проверяется снова.

```
1 while (условие)
2 {
3     тело цикла
4 }
```



# Пример



```
1 #include <stdio.h>
2 int main ()
3 {
4     int result=0;
5     while (result!=4)
6     {
7         printf("Сколько будет 2+2=?");
8         scanf("%d",&result);
9     }
10    printf("Верно!");
11    return 0;
12 }
```

Спрашивать «Сколько будет  $2+2=?$ » до тех пор пока не будет введён верный ответ, когда пользователь введёт 4 — программа должна вывести «Верно!» и завершить работу.

# Пример



```
1 #include <stdio.h>
2 int main ( )
3 {
4     int login,password;
5     printf("Введите логин:");
6     scanf("%d",&login);
7     printf("Введите пароль:");
8     scanf("%d",&password);
9     while (login!=123 || password!=321)
10    {
11        printf("Неверная пара логин/пароль\n");
12        printf("Введите логин:");
13        scanf("%d",&login);
14        printf("Введите пароль:");
15        scanf("%d",&password);
16    }
17    printf("Доступ разрешен\n");
18    return 0;
19 }
```

Программа авторизации запрашивает логин и пароль до тех пор, пока не будет введена верная пара. Если введены неверные логин и пароль, программа выводит «Неверная пара логин/пароль» и запрашивает снова логин и пароль.

# Пример

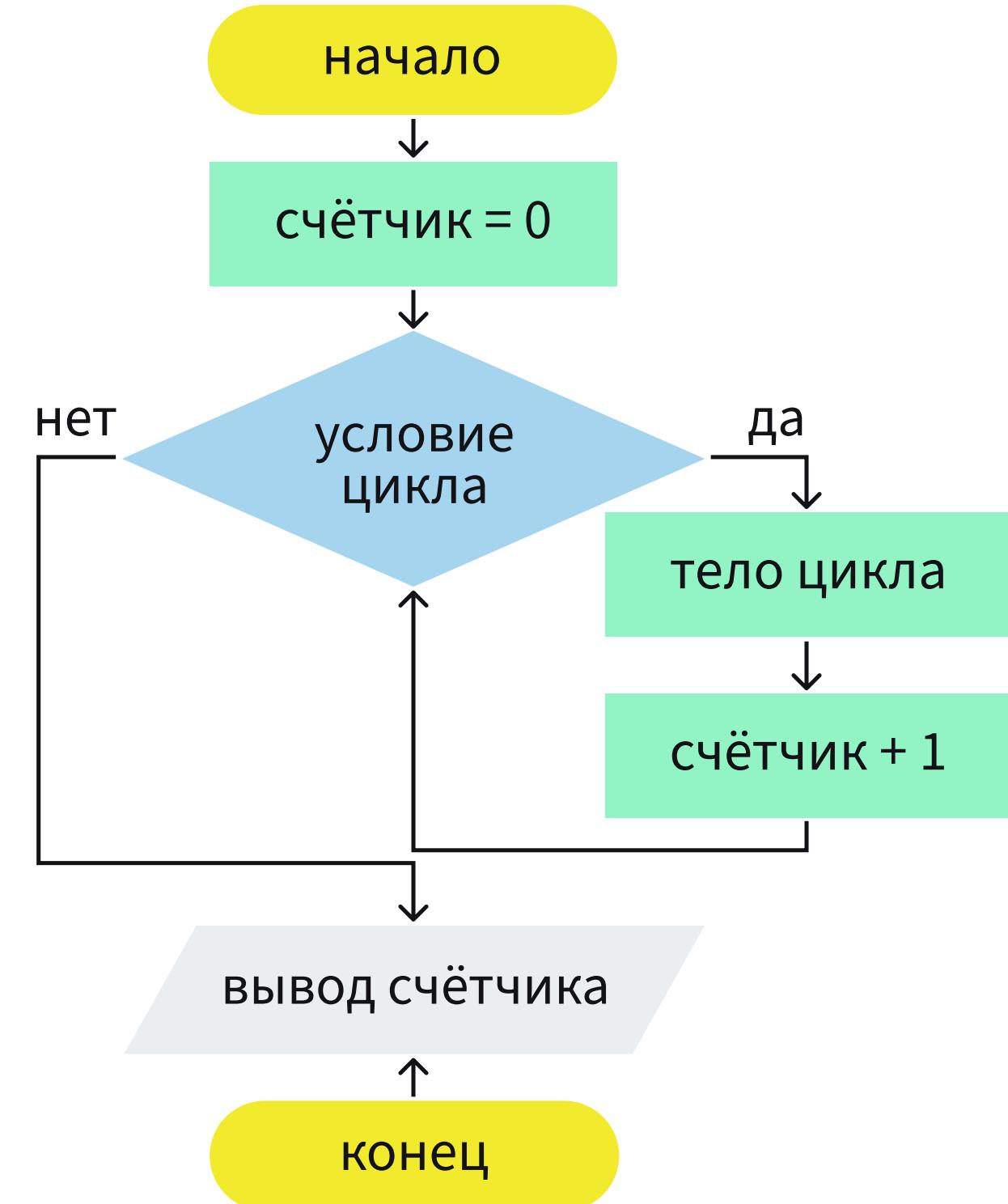


```
1 #include <stdio.h>
2 int main ()
3 {
4     int a=4, sum=0;
5     while (sum<100)
6     {
7         ...
8         scanf("%d",&a);
9         sum+=a;
10    }
11 }
```

Программа запрашивает число, пока сумма введенных чисел меньше 100.

# Подсчёт количества попыток

Спрашивать «Сколько будет  $2+2=?$ » до тех пор пока не будет введён верный ответ, когда пользователь введёт 4 – программа должна вывести «Верно!», вывести количество попыток и завершить работу.

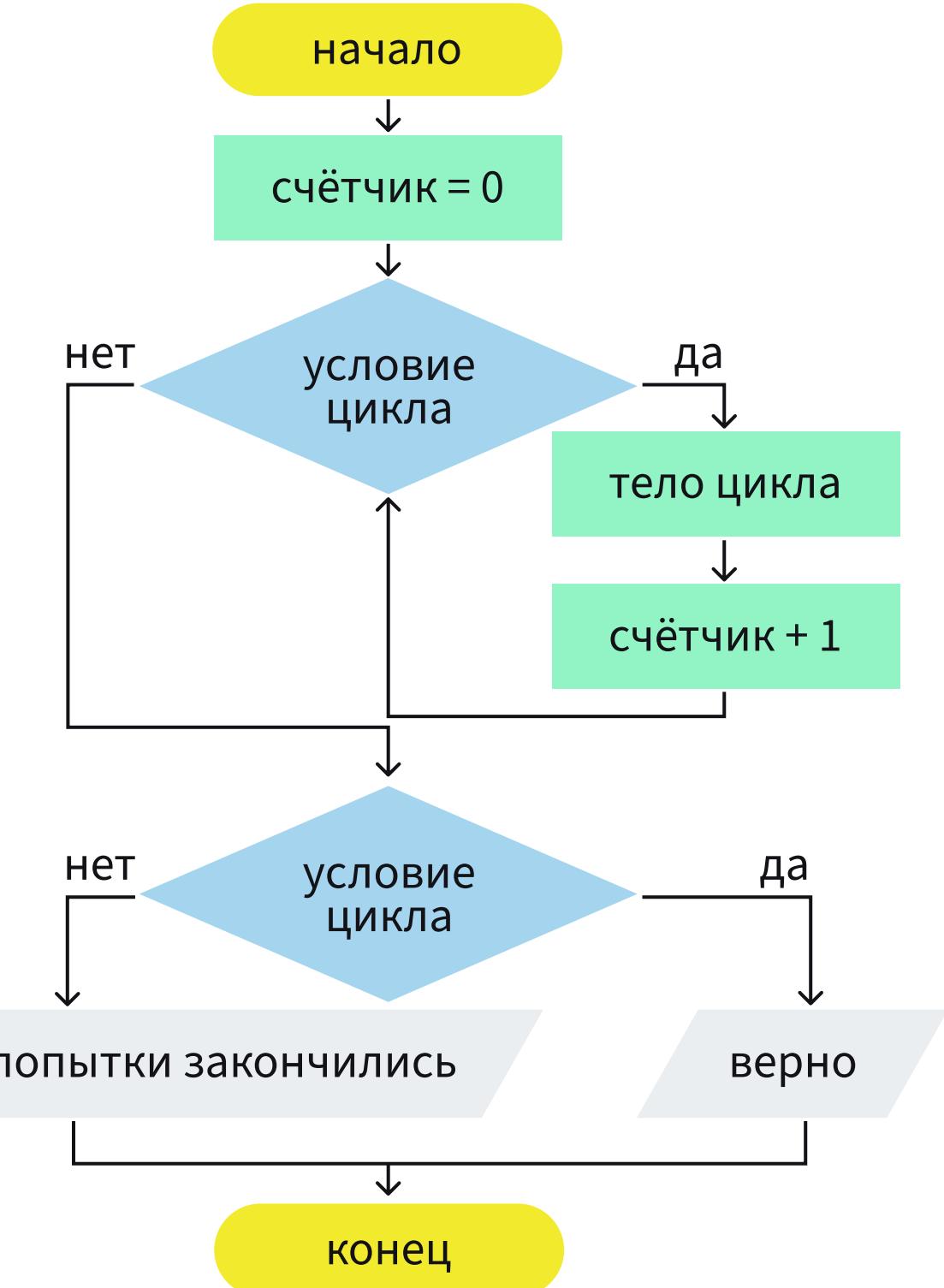


# Подсчёт количества попыток

```
1 #include <stdio.h>
2 int main ( )
3 {
4     int result=0, count=0;
5     while (result!=4)
6     {
7         printf("Сколько будет 2+2=?");
8         scanf("%d",&result);
9         count++;
10    }
11    printf("Верно!\n");
12    printf("Количество попыток:%d\n",count);
13    return 0;
14 }
```

# Ограничение количества попыток

Спрашивать «Сколько будет  $2+2=?$ » до тех пор пока не будет введён верный ответ, когда пользователь введёт 4 – программа должна вывести «Верно!», вывести количество попыток и завершить работу.



# Ограничение количества попыток

```
1 #include <stdio.h>
2 int main ( )
3 {
4     int result=0, count=0;
5     while (count<3 && result!=4)
6     {
7         printf("Сколько будет 2+2=?");
8         scanf("%d",&result);
9         count++;
10    }
11    if (result==4) printf("Верно!\n");
12    else printf("Попытки закончились\n");
13    return 0;
14 }
```

## Особенность цикла с предусловием

**While** — цикл с предусловием, поэтому возможно, что тело цикла не будет выполнено ни разу если в момент первой проверки проверяемое условие окажется ложным.

# Особенность цикла с предусловием

```
1 #include <stdio.h>
2 int main ()
3 {
4     int result=4;
5     while (result!=4)
6     {
7         printf("Сколько будет 2+2=?");
8         scanf("%d",&result)
9     }
10    return 0;
11 }
```

# Цикл с постусловием

Цикл **do...while** — это цикл с постусловием, где истинность условия проверяется после выполнения **тела цикла**, заключенного в фигурные скобки.



Тело цикла выполняется до тех пор, пока выражение, проверяющее условие, не станет ложным.



Тело цикла с постусловием выполнится хотя бы один раз.

```
1 do {  
2     тело цикла  
3 } while (условие)
```

# Пример



```
1 #include <stdio.h>
2 int main () {
3     int num; // объявляем целую переменную для числа
4     do {
5         printf("Введите число от 0 до 10: "); // приглашение пользователю
6         scanf("%d", &num); // ввод числа
7     } while ((num < 0) || (num > 10)); // повторяем цикл пока num<0 или num>10
8     printf("Вы ввели число %d", num); // выводим введенное значение num - от 0 до 10
9     return 0;
10 }
```

# Особенности цикла с постусловием

Цикл `do...while` применяется когда:

- ❖ должна быть выполнена хотя бы одна итерация
- ❖ инициализация объектов, участвующих в проверке условия, происходит внутри тела цикла