

# Программирование на языке Паскаль

## § 58. Циклические алгоритмы

# Что такое цикл?

**Цикл** – это многократное выполнение одинаковых действий.

## Два вида циклов:

- цикл с **известным** числом шагов (сделать 10 раз)
- цикл с **неизвестным** числом шагов (делать, пока не надоест)

**Задача.** Вывести на экран 10 раз слово «Привет».



Можно ли решить известными методами?

# Повторения в программе

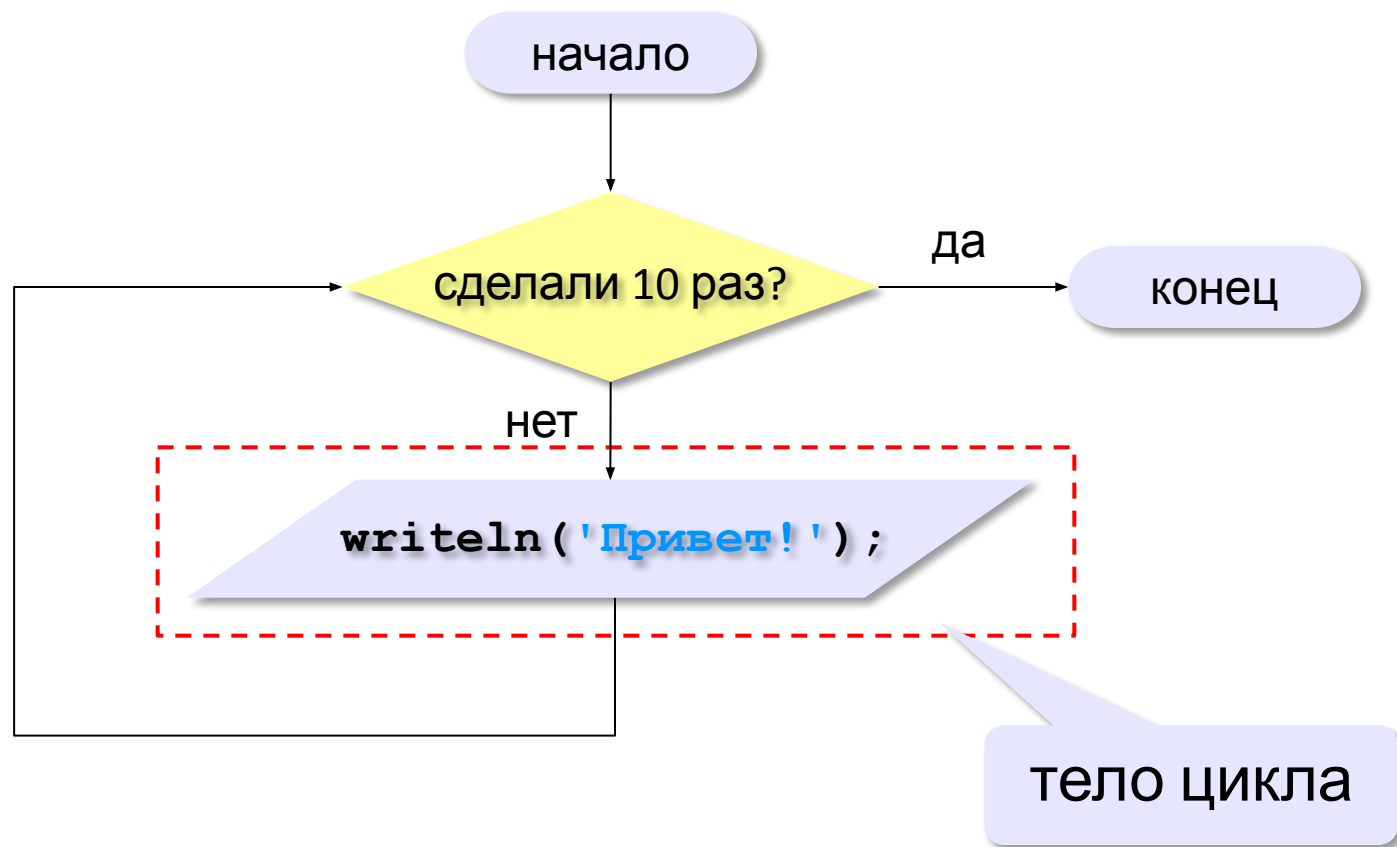
---

```
writeln ( ' Привет' );  
writeln ( ' Привет' );  
writeln ( ' Привет' );  
...  
writeln ( ' Привет' );
```



Что плохо?

# Блок-схема цикла



# Как организовать цикл?

```
счётчик := 0  
пока счётчик < 10  
  writeln('привет');  
  увеличить счётчик на 1
```

```
счётчик := 10  
пока счётчик > 0  
  writeln('привет');  
  уменьшить счётчик на 1
```

результат операции  
автоматически  
сравнивается с нулём!



Какой способ удобнее для процессора?

## Цикл с условием

**Задача.** Определить **количество цифр** в десятичной записи целого положительного числа, записанного в переменную **n**.

```
счётчик := 0
пока n > 0
    отсечь последнюю цифру n
    увеличить счётчик на 1
```

n	счётчик
1234	0

**?** Как отсечь последнюю цифру?

```
n := n div 10
```

**?** Как увеличить счётчик на 1?

```
счётчик := счётчик + 1
```

# Цикл с условием

начальное значение  
счётчика

условие  
продолжения

заголовок  
цикла

```
count := 0;  
while n > 0 do begin  
    n := n div 10;  
    count := count + 1  
end;
```

тело цикла



Зачем begin-end?



Цикл с предусловием – проверка на входе в цикл!

# Цикл с условием

---

При известном количестве шагов:

```
k := 0;  
while k < 10 do begin  
    writeln('привет');  
    k := k + 1  
end;
```

Защивание:

```
k := 0;  
while k < 10 do  
    writeln('привет');
```



# Сколько раз выполняется цикл?

```
a := 4; b := 6;  
while a < b do a := a + 1;
```

2 раза  
a = 6

```
a := 4; b := 6;  
while a < b do a := a + b;
```

1 раз  
a = 10

```
a := 4; b := 6;  
while a > b do a := a + 1;
```

0 раз  
a = 4

```
a := 4; b := 6;  
while a < b do b := a - b;
```

1 раз  
b = -2

```
a := 4; b := 6;  
while a < b do a := a - 1;
```

**защелкивание**

# Цикл с постусловием

заголовок  
цикла

```
repeat
```

```
  write ( 'Введите n > 0: ' );  
  read (n)
```

тело цикла

```
until n > 0 ;
```

условие  
окончания

- при входе в цикл условие **не проверяется**
- цикл всегда выполняется **хотя бы один раз**
- в последней строке указывают **условие окончания** цикла, а не условие его продолжения

# Задачи

---

**«А»:** Напишите программу, которая получает два целых числа  $A$  и  $B$  ( $0 < A < B$ ) и выводит квадраты всех натуральных чисел в интервале от  $A$  до  $B$ .

**Пример:**

Введите два целых числа:

10 12

$10 * 10 = 100$

$11 * 11 = 121$

$12 * 12 = 144$

**«В»:** Напишите программу, которая получает два целых числа и находит их произведение, не используя операцию умножения. Учтите, что числа могут быть отрицательными.

**Пример:**

Введите два числа:

10 -15

$10 * (-15) = -150$

# Задачи

---

**«С»:** Ввести натуральное число  $N$  и вычислить сумму всех чисел Фибоначчи, меньших  $N$ . Предусмотрите защиту от ввода отрицательного числа  $N$ .

**Пример:**

Введите число  $N$ :

10000

Сумма 17709

## Задачи-2

---

**«А»:** Ввести натуральное число и найти сумму его цифр.

**Пример:**

Введите натуральное число:

12345

Сумма цифр 15.

**«В»:** Ввести натуральное число и определить, верно ли, что в его записи есть две одинаковые цифры, стоящие рядом.

**Пример:**

Введите натуральное число:

12342

Нет.

**Пример:**

Введите натуральное число:

12245

Да.

## Задачи-2

---

**«С»:** Ввести натуральное число и определить, верно ли, что в его записи есть две одинаковые цифры (не обязательно стоящие рядом).

**Пример:**

Введите натуральное число:

**12342**

Да.

**Пример:**

Введите натуральное число:

**12345**

Нет.