

# Программирование на языке Паскаль

## Тема 5. Циклы с условием

# Цикл с неизвестным числом шагов

---

**Пример:** Отпилить полено от бревна. Сколько раз надо сделать движения пилой?

**Задача:** Ввести целое число (<2000000) и определить число цифр в нем.

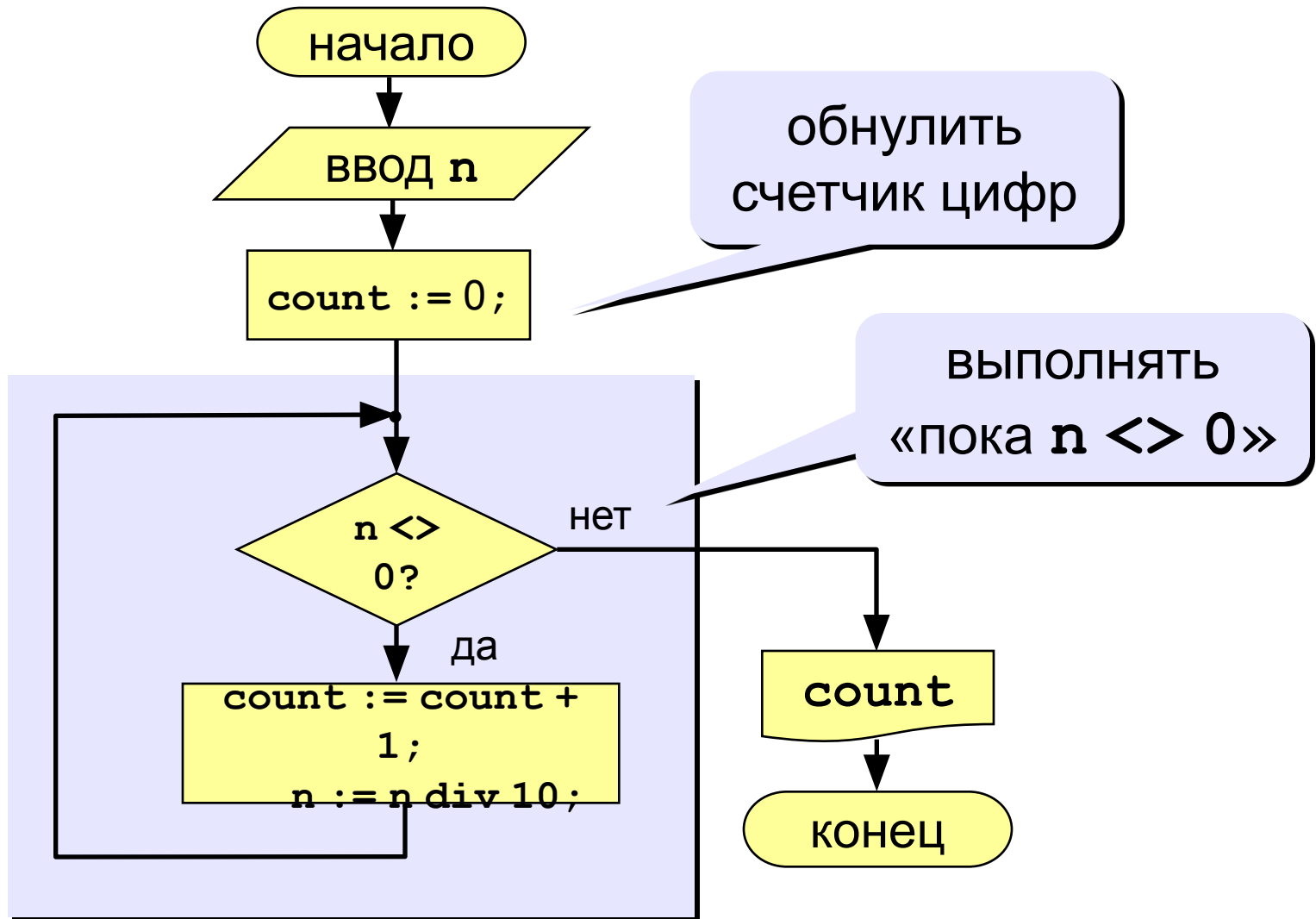
**Идея решения:** Отсекаем последовательно последнюю цифру, увеличиваем счетчик.

n	count
123	0
12	1
1	2
0	3

**Проблема:** Неизвестно, сколько шагов надо сделать.

**Решение:** Надо остановиться, когда  $n = 0$ , т.е. надо делать «пока  $n \neq 0$ ».

# Алгоритм



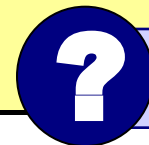
# Программа

```
program qq;  
var n, count, n1: integer;  
begin  
  writeln('Введите целое число');  
  read(n); n1 := n;  
  count := 0;  
  while n <> 0 do begin  
    count := count + 1;  
    n := n div 10;  
  end;  
  writeln('В числе ', n1, ' нашли ',  
          count, ' цифр');  
end.
```

ВЫПОЛНЯТЬ  
«ПОКА n <> 0»

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

```
writeln('В числе ', n1, ' нашли ',  
        count, ' цифр');
```



Что плохо?

# Цикл с условием

---

```
while <условие> do begin
    {тело цикла}
end;
```

## Особенности:

- МОЖНО ИСПОЛЬЗОВАТЬ СЛОЖНЫЕ УСЛОВИЯ:

```
while (a < b) and (b < c) do begin
    {тело цикла}
end;
```

- если в теле цикла только один оператор, слова **begin** и **end** можно не писать:

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

# Цикл с условием

---

## Особенности:

- условие пересчитывается **каждый раз** при входе в цикл
- если условие на входе в цикл ложно, цикл не выполняется ни разу

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

- если условие никогда не станет ложным, программа **зацикливается**

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

# 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;
```

**зацикливание**

# Замена for на while и наоборот

```
for i:=1 to 10 do begin
  {тело цикла}
end;
```

```
i := 1;
while i <= 10 do begin
  {тело цикла}
  i := i + 1;
end;
```

```
for i:=a downto b do
  begin
    {тело цикла}
  end;
```

```
i := a;
while i >= b do begin
  {тело цикла}
  i := i - 1;
end;
```

Замена цикла **for** на **while** возможна **всегда**.

Замена **while** на **for** возможна только тогда, когда можно заранее **рассчитать число шагов цикла**.



# Задания

---

**«1»:** Ввести целое число и найти сумму его цифр.

**Пример:**

Введите целое число:

**1234**

Сумма цифр числа 1234 равна 10.

# Задания

---

**«2»:** Написать программу в результате выполнения которой выяснится, входит ли цифра 2 в запись данного целого числа  $n$ .

**Пример:**

Введите целое число:

**5231**

Цифра 2 входит в запись данного числа.