

# **ПРОГРАММИРОВАНИЕ ЦИКЛИЧЕСКИХ АЛГОРИТМОВ**

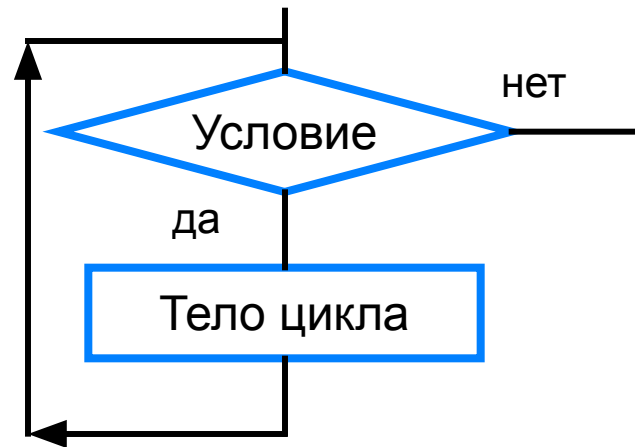
**НАЧАЛА ПРОГРАММИРОВАНИЯ**

# Ключевые слова

- **while** ( цикл-ПОКА)
- **repeat** (цикл-ДО)
- **for** (цикл с параметрами)



# Программирование циклов с заданным условием продолжения работы



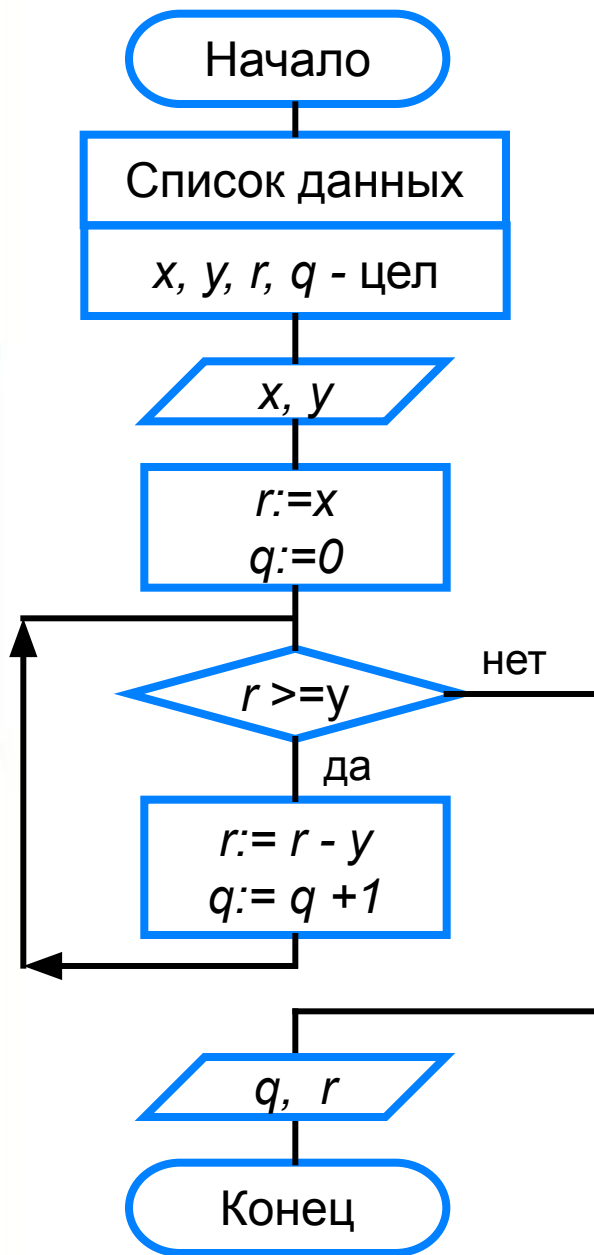
Общий вид оператора:

**while** <условие> **do** <оператор>

Здесь:

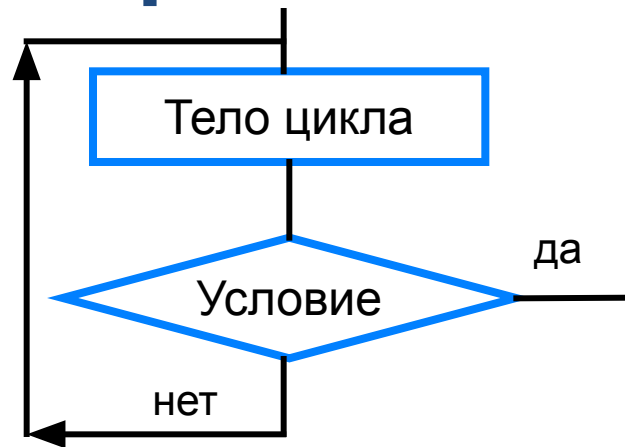
<условие> - логическое выражение;  
пока оно истинно, выполняется тело цикла;

<оператор> - простой или составной оператор,  
с помощью которого записано тело цикла.



```
program n_14;  
  var x, y, q, r: integer;  
begin  
  writeln ('Частное и остаток');  
  write ('Введите делимое X>>');  
  readln (x);  
  write ('Введите делитель y>>');  
  read (y);  
  r:=x;  
  q:=0;  
  while r>=y do  
  begin  
    r:=r-y;  
    q:=q+1  
  end;  
  writeln ('Частное q=', q);  
  writeln ('Остаток r=', r)  
end.
```

# Программирование циклов с заданным условием окончания работы



Общий вид оператора:

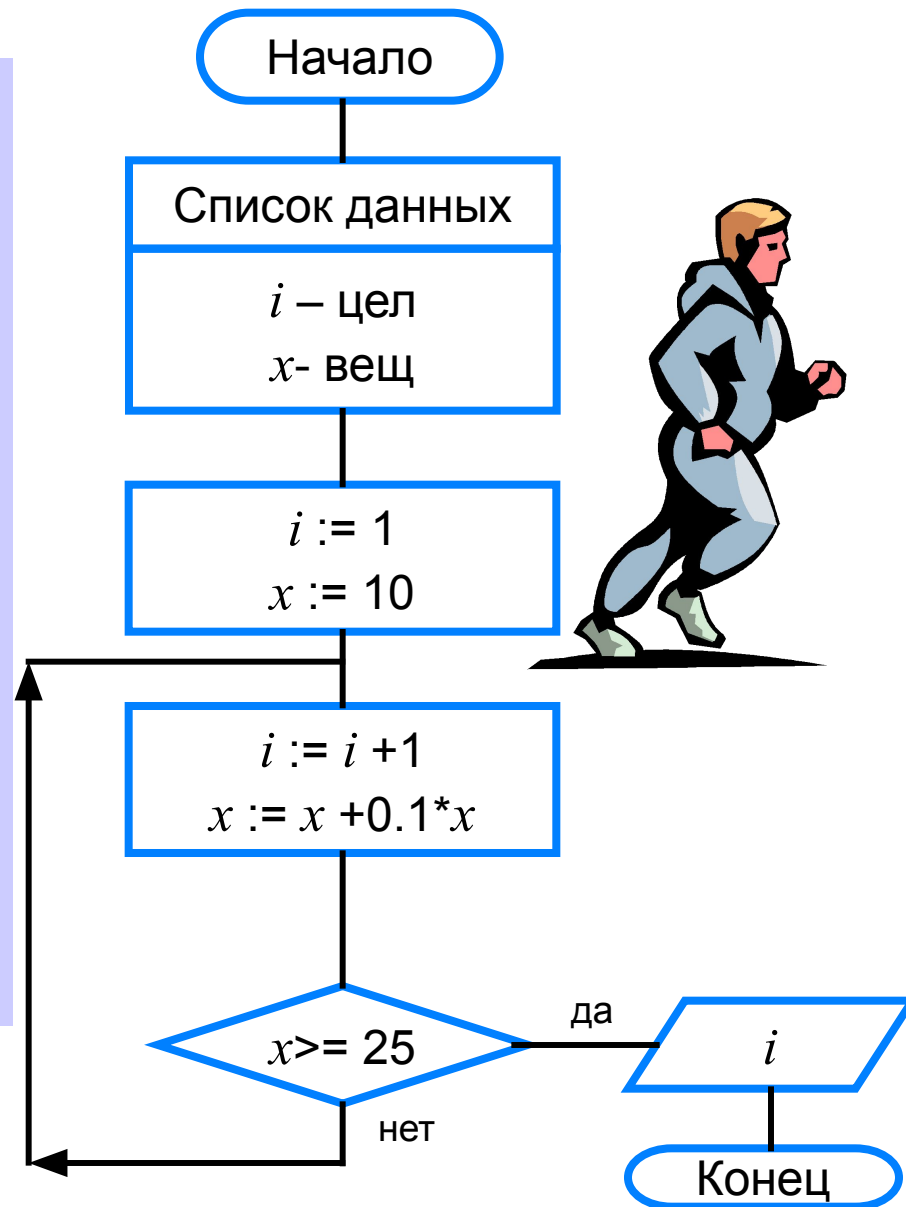
**repeat** <оператор1; оператор2; ...; > **until** <условие>

Здесь:

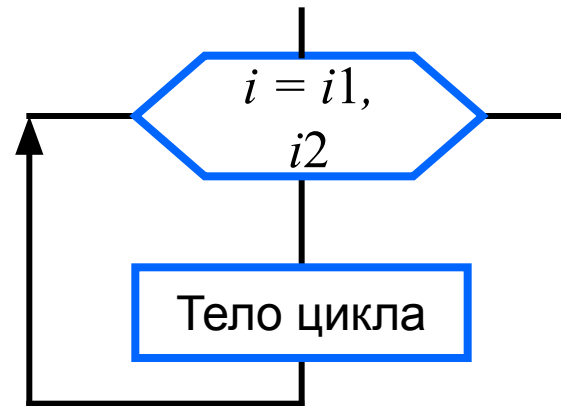
<оператор1>; <оператор2>; ... - операторы, образующие тело цикла;

<условие> - логическое выражение; если оно ложно, то выполняется тело цикла.

```
program n_15;  
  var i: integer; x: real;  
begin  
  writeln ('График тренировок');  
  i:=1;  
  x:=10;  
  repeat  
    i:=i+1;  
    x:=x+0.1*x;  
  until x>=25;  
  writeln ('Начиная с ', i, '-го дня  
спортсмен будет пробегать 25 км')  
end.
```



# Программирование циклов с заданным числом повторений



Общий вид оператора:

```
for <параметр>:=<начальное_значение>  
to <конечное_значение> do <оператор>
```

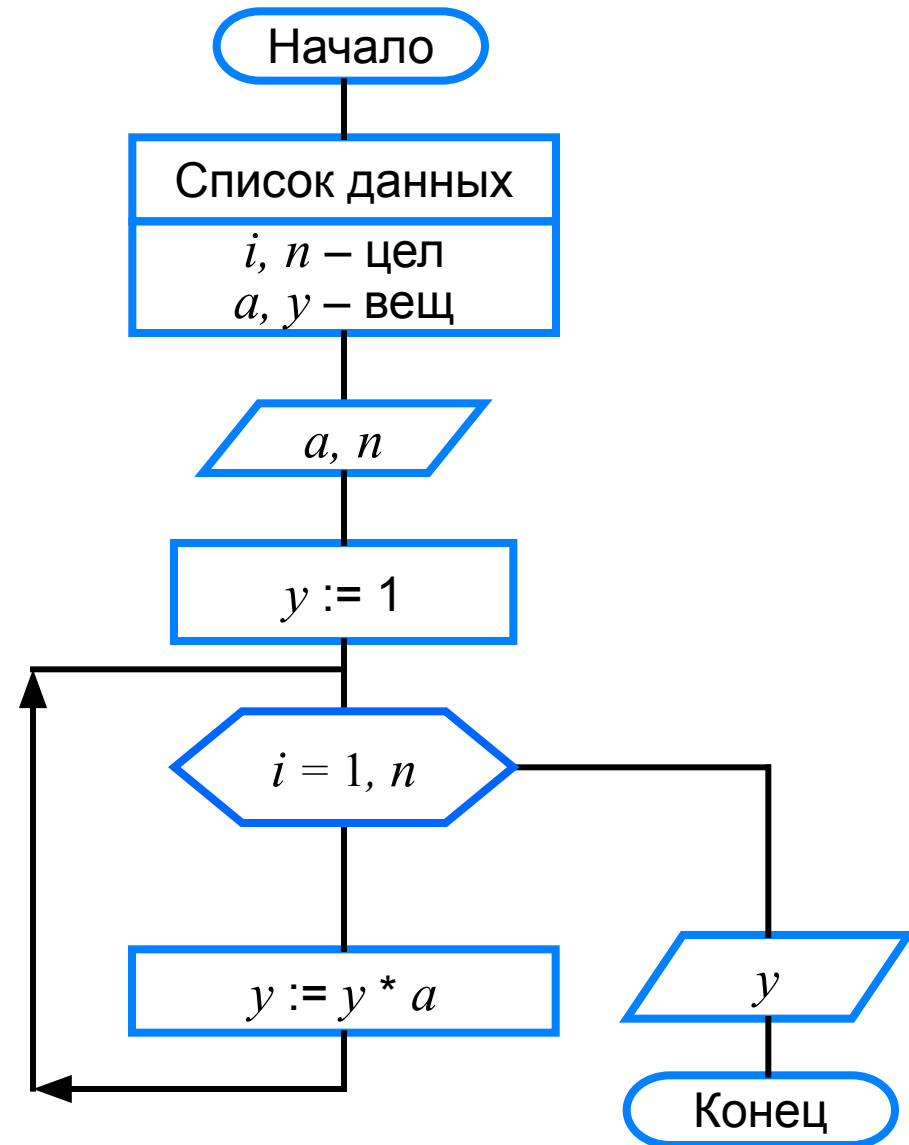
Здесь:

<параметр> - переменная целого типа;

После каждого выполнения тела цикла происходит увеличение на единицу параметра цикла; условие выхода из цикла - превышение параметром конечного значения.

<оператор> - простой или составной оператор - тело цикла.

```
program n_16;  
  var i,n:integer;a,y:real;  
begin  
  writeln ('Возведение в степень');  
  write ('Введите основание a>>');  
  readln (a);  
  write ('Введите показатель n>>');  
  readln (n);  
  y:=1;  
  for i:=1 to n do y:=y*a;  
  writeln ('y=', y)  
end.
```





# Различные варианты программирования циклического алгоритма

Для решения одной и той же задачи могут быть созданы разные программы.

Организуем ввод целых чисел и подсчёт количества введённых положительных и отрицательных чисел. Ввод должен осуществляться до тех пор, пока не будет введён ноль.

В задаче в явном виде задано условие окончания работы.

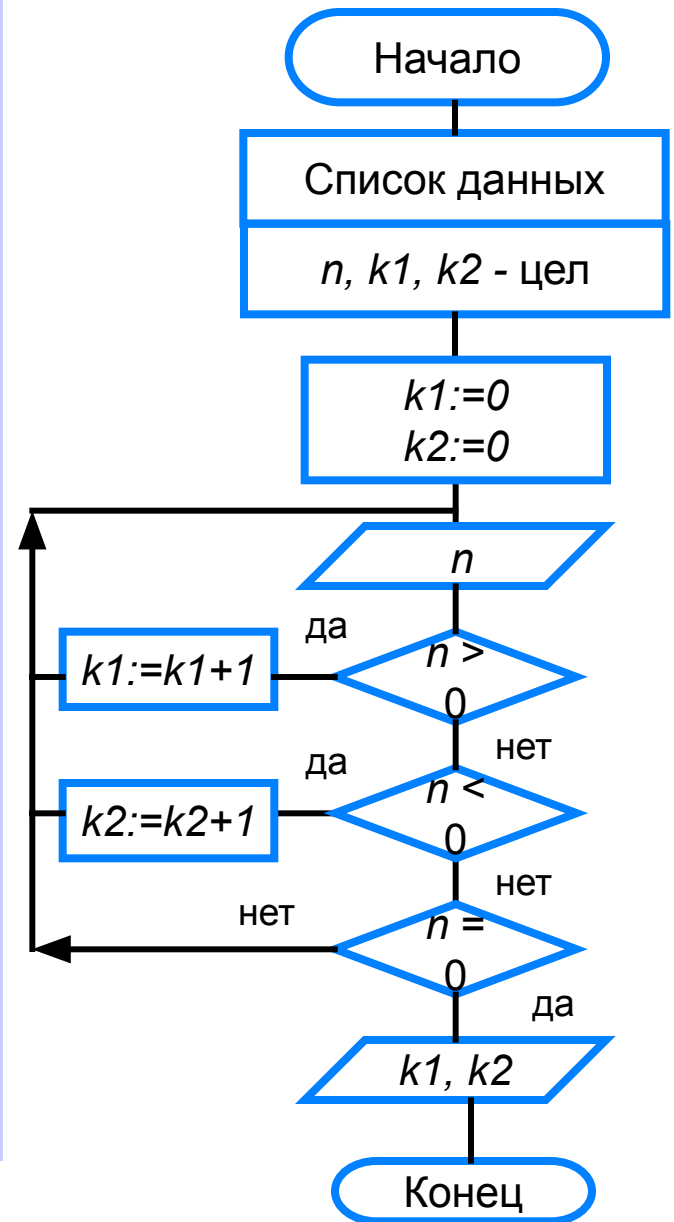


Воспользуемся оператором **repeat**.

```

program n_17;
  var n, k1, k2: integer;
begin
  k1:=0;
  k2:=0;
  repeat
    write ('Введите целое число>>');
    readln (n);
    if n>0 then k1:=k1+1;
    if n<0 then k2:=k2+1;
  until n=0;
  writeln ('Введено:');
  writeln ('положительных чисел – ', k1);
  writeln ('отрицательных чисел – ', k2)
end.

```

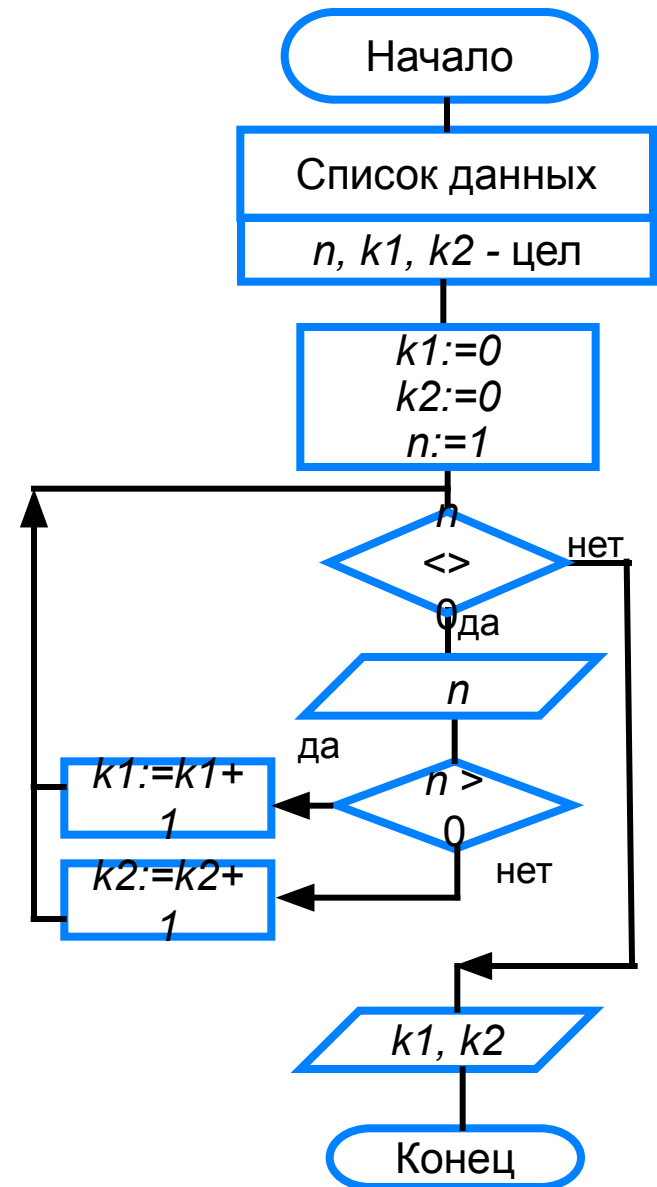


Ввод осуществляется до тех пор, пока не будет введен ноль.

Работа продолжается, пока  $n \neq 0$ .

Воспользуемся оператором **while**:

```
program n_18;  
  var n, k1, k2: integer;  
begin  
  k1:=0;  
  k2:=0;  
  n:=1;  
  while n<>0 do  
  begin  
    writeln ('Введите целое число>>');  
    read (n);  
    if n>0 then k1:=k1+1;  
    if n<0 then k2:=k2+1;  
  end;  
  writeln ('Введено:');  
  writeln ('положительных – ', k1);  
  writeln ('отрицательных – ', k2)  
end.
```



# Самое главное

В языке Паскаль имеются три вида операторов цикла:

*while* (цикл-ПОКА)

*repeat* (цикл-ДО)

*for* (цикл с параметром).

Если число повторений тела цикла известно, то лучше воспользоваться оператором *for*; в остальных случаях используются операторы *while* и *repeat*.



# Вопросы и задания

Дана последовательность операторов:

a:=1;

b:=1;

**repeat**

a:=a+1;

b:=b\*2;

**until** b>8;

s:=a+b

Проанализируйте следующий цикл:

**while** a<b **do**

c:=a=b;

В чём его особенность?

Сколько раз будет повторён цикл и какими будут значения переменных **a**, **b**, **s** после исполнения этой последовательности операторов?

Пример входных данных	Пример выходных данных
Введите n> 5	5! = 120
Введите n> 6	6! = 720

Напишите программу, которая генерирует 10 случайных чисел в диапазоне от 1 до 20, выводит эти числа на экран и вычисляет их среднее арифметическое.

Напишите программу, которая 10 раз выводит на экран ваши имя и фамилию.

Напишите программу вычисления наибольшего общего делителя двух целых чисел:

- а) используйте оператор **repeat**;
- б) используйте оператор **while**.

Какой из трёх рассмотренных операторов цикла является, по вашему мнению, основным, т. е. таким, что им можно заменить два других?

Обоснуйте свою точку зрения.

Напишите программу, которая вычисляет сумму:

- а) первых  $n$  натуральных чисел;
- б) квадратов первых  $n$  натуральных чисел;
- в) всех чётных чисел в диапазоне от 1 до  $n$ ;
- г) всех двузначных чисел.

Пример входных данных	Пример выходных данных
Введите $n > 5$	$5 * 2 = 10$ $5 * 3 = 15$ $5 * 4 = 20$ $5 * 5 = 25$ $5 * 6 = 30$ $5 * 7 = 35$ $5 * 8 = 40$ $5 * 9 = 45$ $5 * 10 = 50$



Дана последовательность операторов:

a:=1;

b:=2;

while a+b<8 do

begin

a:=a+1;

b:=b+2;

end;

s:=a+b

Сколько раз будет повторен цикл и какими будут значения переменных **a**, **b**, **s** после исполнения этой последовательности операторов?

Напишите программу, выводящую на экран изображение шахматной доски, где чёрные клетки изображаются звёздочками, а белые - пробелами. Рекомендуемый вид экрана после выполнения программы представлен ниже:

```
*      *      *      *
      *      *      *      *
*      *      *      *
      *      *      *      *
*      *      *      *
      *      *      *      *
*      *      *      *
      *      *      *      *
```

Сколько раз будет выполнен цикл?

а) **for** i:=0 **to** 15 **do** s:=s+1;

б) **for** i:=10 **to** 15 **do** s:=s+1;

в) **for** i:=-1 **to** 1 **do** s:=s+1;

г) **for** i:=10 **to** 10 **do** s:=s+1;

д) k:=5;

**for** i:=k-1 **to** k+1 **do** s:=s+1;

# Опорный конспект

В языке Паскаль имеются три вида операторов цикла:

