

# *Операторы цикла*

**Циклы позволяют многократно выполнять одну или группу команд, причем в тексте программы нет необходимости записывать эти команды несколько раз.**

**В языке программирования PASCAL существует три вида циклов:**

- 1. Арифметический (перечисляемый) цикл FOR**
- 2. Логический цикл с предусловием WHILE**
- 3. Логический цикл с послеусловием REPEAT**

# Арифметический цикл **FOR**

```
For i := n1 to n2 do оператор;
```

**i** – параметр цикла (счетчик),  
переменная целого типа;  
**n1** и **n2** – начальное и конечное  
значения счетчика.

**Особенностью арифметического цикла является то, что число повторений операторов цикла должно быть известно заранее. Решение о выполнении или невыполнении в очередной раз тела цикла принимается до начала его прохождения, поэтому может случиться так, что тело цикла не будет выполнено ни разу.**



# Порядок выполнения цикла FOR

- 1) вычисляются значения выражений  $n1$ ,  $n2$ ;
- 2) параметру цикла присваивается значение  $n1$ ;
- 3) если полученное значение счетчика больше  $n2$ , то выполнение цикла заканчивается;
- 4) выполняется тело цикла;
- 5) значение параметра цикла увеличивается на 1, осуществляется переход к пункту 3.

Количество проходов цикла с заголовком

```
for i := n1 to n2 do
```

можно вычислить по формуле  $n2 - n1 + 1$

**Количество выполнений тела цикла определяется до первого выполнения тела цикла, поэтому, если в выражении  $n$  используются переменные, которые изменяются в теле цикла, это никак не повлияет на количество выполнений тела цикла. Например, в следующей последовательности операторов:**

```
n := 10;  
For i := 1 to n do  
  Begin  
    Write (i);  
    n := 100;  
  End;
```

**тело цикла будет выполняться 10 раз, хотя уже после первого выполнения тела цикла конечное значение параметра цикла станет равным 100.**

**Сколько раз будет выполнено тело цикла с данным заголовком?**

- 1) `for i := -10 to -4 do`
- 2) `for i := 6 to 2 do`
- 3) `for i := 3 to 3 do`

Цикл под номером 1 будет выполняться для счетчика, последовательно принимающего значения  $-10, -9, -8, -7, -6, -5, -4$ , то есть 7 раз.

Цикл под номером 2 не будет выполняться ни разу, так как начальное значение счетчика больше конечного.

Цикл под номером 3 будет выполняться 1 раз для счетчика, равного 3.



**Значение счетчика цикла может использоваться в выражениях, входящих в операторы тела цикла, но изменение значения счетчика цикла этими операторами недопустимо.**

**Попробуйте ответить, какие фрагменты программы записаны без ошибок.**

- 1) `FOR I := 2 TO 20 Do  
  WriteLn (I);`
- 2) `FOR I := 2 TO 20 Do  
  I := I + 1;`
- 3) `FOR I:= 2 TO 20 Do  
  ReadLn (I);`
- 4) `FOR I := 2 TO 20 Do  
  A := I * I;`

**Правильно записаны фрагменты под номерами 1 и 4. Во фрагментах 2 и 3 в операторах тела цикла производится изменение значения счетчика цикла.**

В Паскале имеются два способа для организации регулярного цикла: прямой и обратный.

Прямой пересчет идет от известного меньшего числа до известного большего, на каждом шаге прибавляется единица (например, от 20 до 25: 20, 21, 22, 23, 24, 25).

Обратный пересчет – от большего к меньшему. И на каждом шаге вычитается единица.

1) цикл с прямым отсчетом:

```
for i := n1 to n2 do оператор;
```

2) цикл с обратным отсчетом:

```
for i := n2 downto n1 do оператор;
```

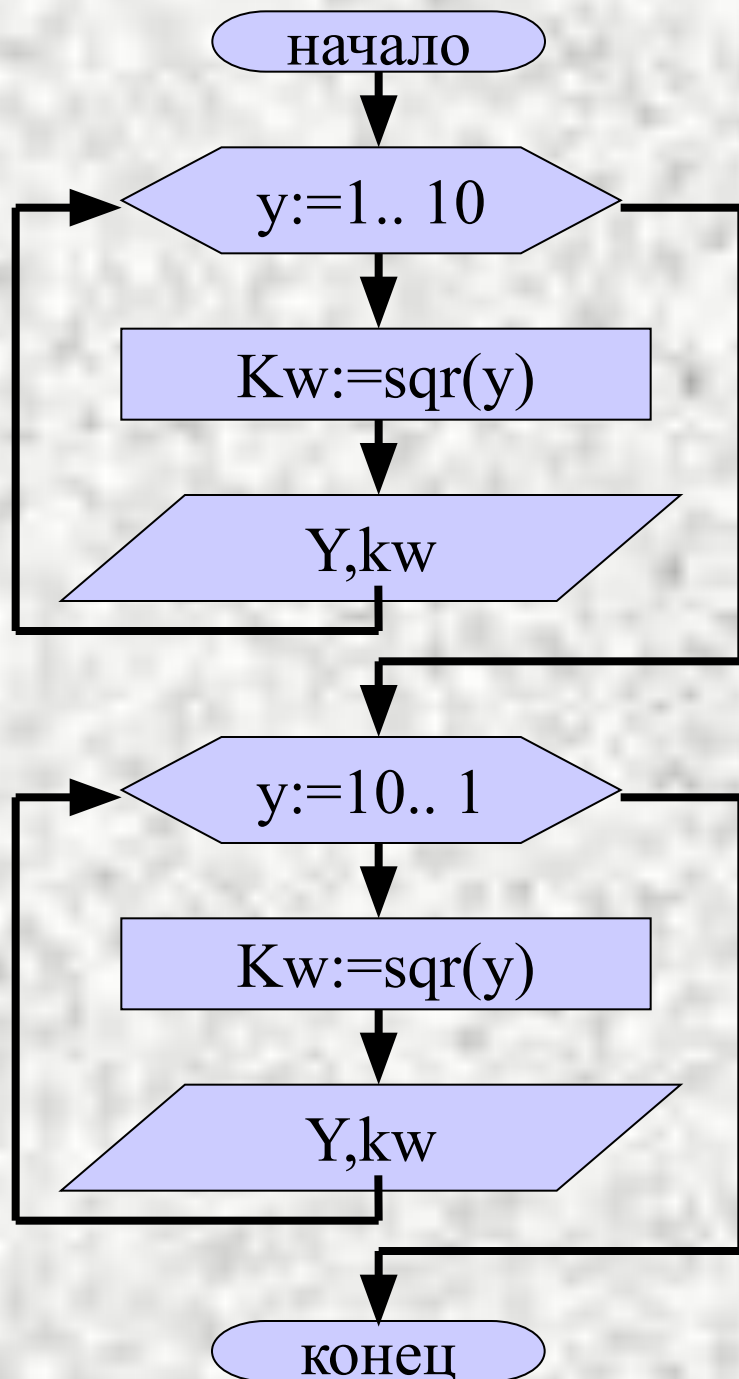


Использование оператора цикла *for*.

Вывести на экран значение  $y^2$

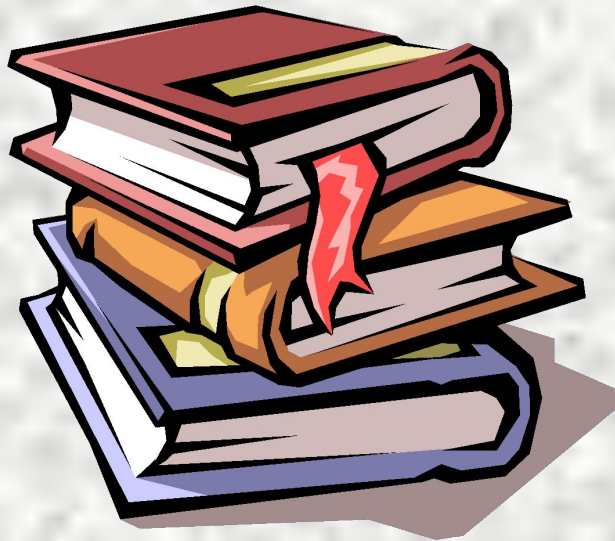
( $y=1,2,\dots,10$ )

в возрастающем и убывающем  
порядке

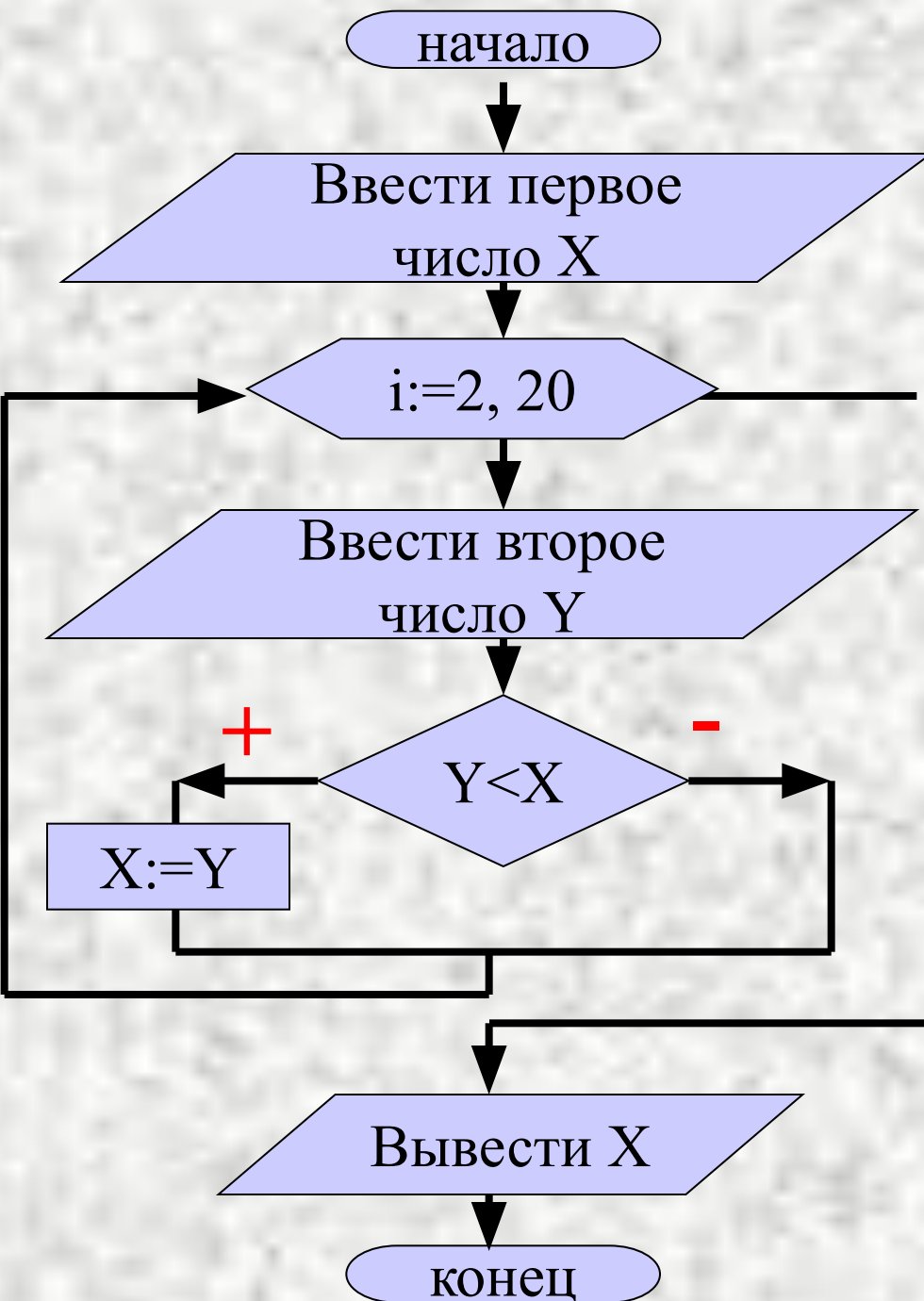


**Program KWADRAT;**  
**Var kw, y: integer;**  
**Begin**  
**For y:=1 to 10 do**  
    **Begin**  
        **kw:= sqr(y);**  
        **Writeln (y:3,kw:5);**  
    **End;**  
**Writeln;**  
**For y:=10 downto 1 do**  
    **Begin**  
        **kw:=sqr(y);**  
        **Writeln (y:3,kw:5);**  
    **End;**  
**End.**

**Выбрать наименьшее значение из 20 чисел,  
вводимых с клавиатуры:**







```
program min;
var i:integer;
    min, number : real;
begin
  writeln ('Enter first');
  readln (min);
  for i:=2 to 20 do
  begin
    write ('Enter next');
    readln (number);
    if number < min then
      min := number;
    end;
  end;
  writeln ('min=', min :6:2);
end.
```

# **Вычисление суммы чисел**

**Вычислить  
сумму  
n чисел,  
вводимых  
с клавиатуры.**

```
program Summa;  
var i, n :integer;  
x, sum : real;  
begin  
  writeln ('Enter n');  
  readln (n);  
  sum:=0;  
  for i:=1 to n do  
    begin  
      write ('Enter x');  
      readln (x);  
      sum:=sum+ x;  
    end;  
  writeln ('sum=', sum:6:2);  
end.
```



**Среди всех двузначных чисел найти те, сумма цифр которых равна n ( $0 < n \leq 18$ ) и количество этих чисел.**

```
Program Chisla;  
var n, i, k, p1, p2:integer;  
begin  
k:=0;  
writeln ('Vvesti n');  
readln (n);  
for i:=10 to 99 do  
    begin  
        p1:=i div 10;  
        p2:= i mod 10;  
        if (p1+ p2) =n then  
            begin  
                writeln (i);  
                k:=k+1;  
            end;  
    end;  
writeln ('Koli4estvo 4isel ', k);  
end.
```

# **Задания для самостоятельного выполнения**

- 1. Написать программу, которая вычисляет среднее арифметическое вводимой с клавиатуры последовательности дробных чисел. Количество чисел должно задаваться во время работы программы.**
- 2. Написать программу, которая находит сумму целых положительных чисел, которые больше 20, меньше 100, кратны 3 и заканчиваются на 2, 4 или 8.**
- 3. Написать программу, которая выводит на экран таблицу стоимости порций мороженого.**

<b>Количество</b>	<b>Пломбир</b>	<b>Каштан</b>	<b>Крем-брюле</b>
<b>1</b>	<b>0.80</b>	<b>1.60</b>	<b>1.00</b>
<b>2</b>	<b>1.60</b>	<b>3.20</b>	<b>2.00</b>
<b>...</b>			
<b>10</b>	<b>8.00</b>	<b>16.00</b>	<b>10.00</b>