



Тема урока:
**Циклы с
параметром**

Операторы цикла в Pascal

Циклом называется такая алгоритмическая структура, в которой серия команд (тело цикла) выполняется многократно.

При решении некоторых задач вычислительный процесс имеет циклический характер. Это означает, что часть операторов многократно выполняется при разных значениях переменных. В Pascal имеется три вида операторов цикла:

- 1) цикл с параметром или цикл типа **for**,
- 2) цикл с предусловием или цикл типа **while**,
- 3) цикл с постусловием или цикл типа **repeat ... until**.

Оператор цикла с параметром

Оператор цикла с параметром используется в тех случаях, когда заранее известно, сколько раз должна повториться циклическая часть программы.



Формат оператора:

```
for параметр:= i1 to i2 do <оператор>
```

Здесь:

for (для), **to** (до), **do** (выполнить) - служебные слова;

<параметр> - переменная целого типа;

i1 – начальное значение, **i2** – конечное значение параметра - это выражения или числа целого типа;

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

Оператор цикла с параметром

```
for i:= <нач_значение> to <кон_значение> do  
  begin  
    операторы циклической части программы  
  end
```

Здесь **i** - параметр цикла;

Выполнение оператора:


Циклическая часть программы выполняется при каждом значении параметра цикла **i** от его начального значения до конечного значения, с увеличением параметра с каждым повторением цикла на 1.

Если значение параметра должно уменьшаться с каждым повторением цикла на -1, то нач. значение должно быть больше конечного и *формат оператора* должен быть следующим:

```
for i:= <нач_значение> downto <кон_значение> do  
  begin  
    операторы циклической части программы  
  end
```

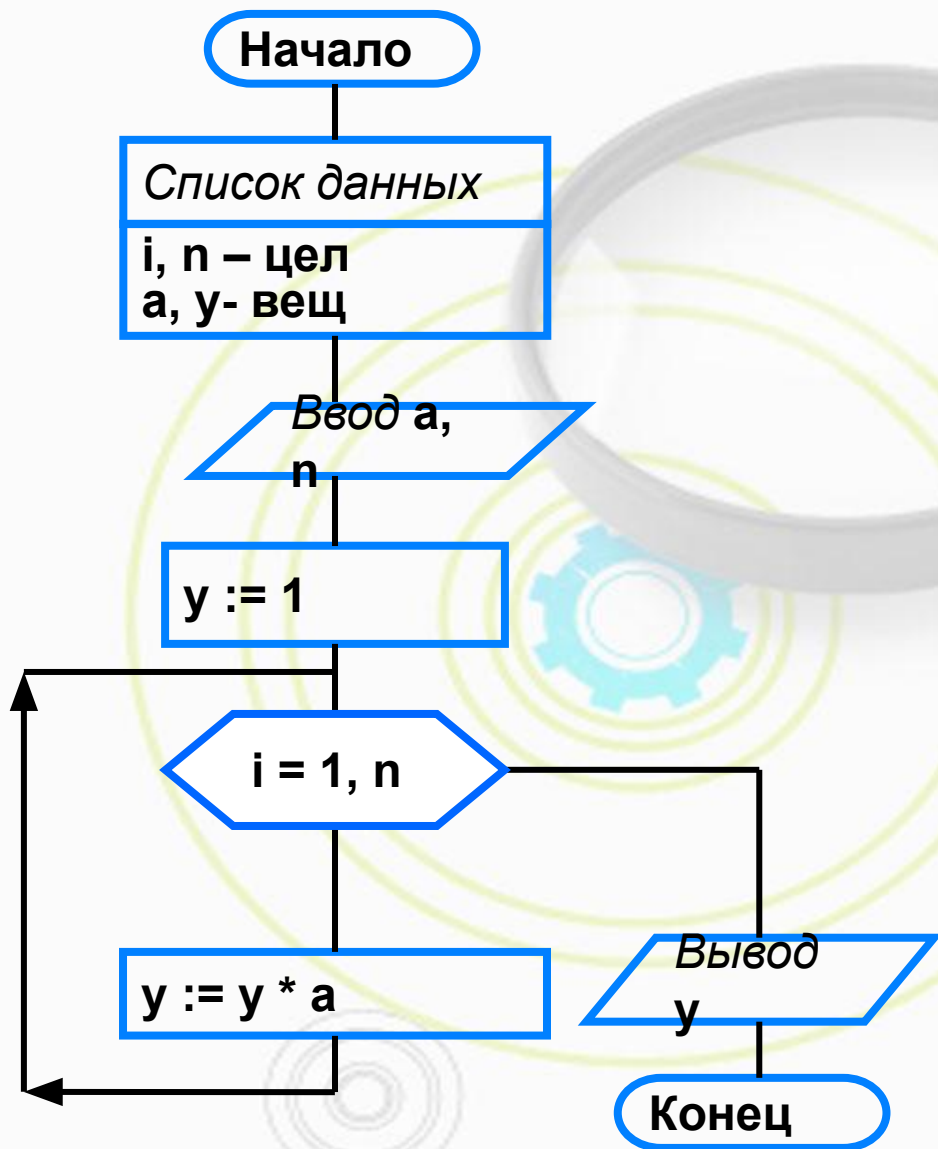

Это необходимо помнить!

- *Переменные оператора цикла должны быть определены до входа в цикл;*
- *Повторяющиеся вычисления записываются всего один раз внутри цикла;*
- *Вход в цикл возможен только через его начало;*
- *Нельзя самим изменять значение параметра (счетчика) внутри цикла;*
- *Передавать управление внутрь цикла нельзя!*
- *Передавать управление из цикла – можно по оператору перехода **goto**.*



***Составление
и выполнение программ
циклической структуры***

Программа, которая возводит в указанную степень основание.



```
program STEPEN;  
  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.
```

Задача 1. Составьте программу вычисления суммы элементов ряда с 1 по 20:

$$S = 5 + 12 + 19 + \dots + (7 * i - 2)$$

где i - порядковый номер элемента,

$(7 * i - 2)$ - формула для вычисления элемента ряда.

Программа:

```
program Summa;  
  var i:integer; S:real;  
begin  
  writeln ('Вычисление суммы ряда');  
  S:=0;  
    for i:=1 to 20 do S:=S+ (7 * i - 2);  
  writeln ('S=', S)  
end.
```


Задача2. Составьте программу вычисления произведения 9-ти элементов ряда:

$$P = 3 * 5 * 7 * \dots * (2 * i + 1)$$

где i - порядковый номер элемента,

$(2 * i + 1)$ - формула для вычисления элемента.

Программа:

```
program Prouzvedenie;  
  var P, i: integer;  
begin  
  writeln ('Вычисление произведения');  
  P:=1;  
  for i:=1 to 9 do P:=P* (2 * i +1);  
  writeln ('P=', P)  
end.
```

Задача 3.

Составьте программу вычисления значений функции $Y = x^2 - 3x - 7$, при изменении аргумента x от 1 до 15 с шагом 1. Вывод значений аргумента и функции организуйте в виде двух столбцов:

$x = \dots$ $y = \dots$

$x = \dots$ $y = \dots$

```
program Tabulirovanie
```

```
  var Y, X: integer;
```

```
  Begin
```

```
    writeln ('Вычисление значений функции');
```

```
    for X:=1 to 15 do
```

```
      Begin
```

```
        Y:= X*X-3*X-7;
```

```
        writeln ('X=', X:6, 'Y=', Y)
```

```
      End
```

```
  End.
```

Задача 4. Составьте программу, которая вычислит и напечатает столбец таблицы умножения числа 8 на число K, которое изменяется от 1 до 10.

Вывести на печать в виде:

$$8 * 1 = 8$$

.....

$$8 * 10 = 80$$

```
program Stolbez
```

```
var K, P: integer;
```

```
Begin
```

```
writeln ('столбец таблицы умножения');
```

```
for K:=1 to 10 do
```

```
Begin
```

```
  P:= 8*K;
```

```
  writeln ('8*', K, '=', P)
```

```
End
```

```
End.
```

РЕФЛЕКСИЯ

Ребята, давайте каждый из вас выскажется одним предложением о том, как для вас прошел текущий урок, выбирая начало фразы из рефлексивного экрана на доске:

1. **Сегодня я узнал...**
2. **Было интересно...**
3. **Было трудно...**
4. **Я выполнял задания...**
5. **Я научился...**
6. **У меня получилось ...**

Задание на дом:

- 1). Выучить конспект урока.
- 2). Выполнить задания:

Задача 1.

Составьте программу вычисления произведения 10 элементов ряда: $P = 1 * 3 * 5 * \dots * (2 * i - 1)$
где i - порядковый номер элемента.

Задача 2.

Составьте программу вычисления функции $y = 2 * x - \text{SQRT}(x)$ при изменении аргумента x от 0 до 6
Вывод значений аргумента и функции организуйте в виде двух столбцов:

x = ... y = ...