

Pascal

Циклы

Счетный оператор FOR

Учитель информатики:
Корогод В.А

Что такое цикл?

В языке программирования имеются три различных оператора с помощью которых можно запрограммировать повторяющиеся действия.

Многократные повторения одних тех действий можно выполнить с помощью конструкции, которые в программировании называются циклами.

Повторяемые действия называют телом цикла.

Операторы цикла в Pascal

В Паскале существует несколько операторов цикла.

FOR – счетный оператор (арифметический цикл).

WHILE – оператор с предусловием (цикл-пока).

REPEAT – оператор с постусловием (цикл-до).



Рис. 3.8. Структура циклов, реализованных в Borland Pascal:

а – цикл-пока; б – цикл-до; в – счетный цикл

Счетный оператор FOR

Первый оператор цикла с которым мы познакомимся это оператор FOR, который повторяет тело цикла заданное число раз.

Оператор FOR имеет две формы:

1. Первая форма - с изменением значений параметра цикла по **возрастанию**.
2. Вторая форма – с изменением значений параметра цикла по **убыванию**.

Первая форма оператора

for переменная:=начальное значение **to** конечное значение **do**
оператор

Текст от слова **for** до слова **do** включительно называется
заголовком цикла, а оператор после **do** - *телом цикла*.

Переменная после слова **for** называется *параметром цикла*.

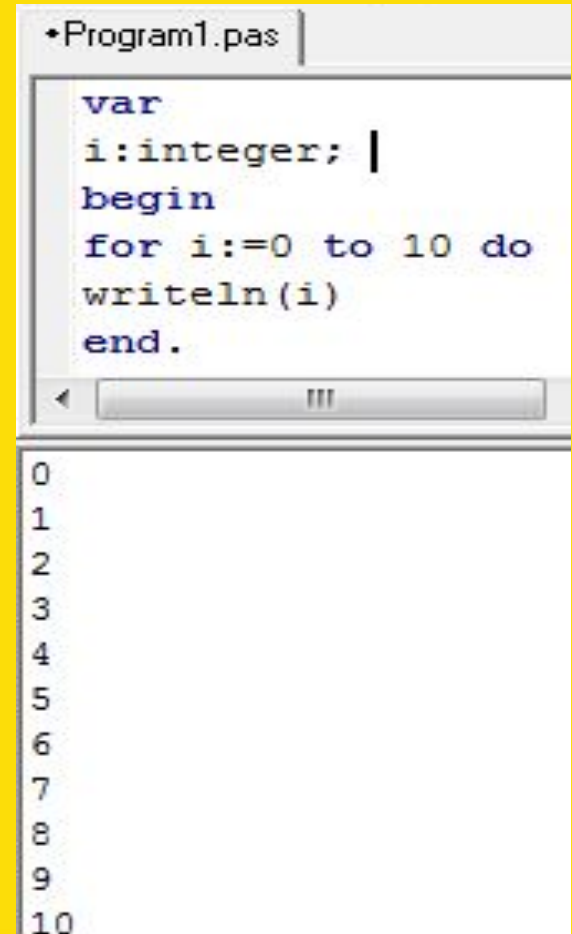
Естественным языком запись цикла можно перефразировать так:
Для переменной начиная с «**начального значения**» до
«**конечного значения**» выполнять «**оператор**».

В ходе цикла данной формы значение переменной
увеличивается строго на единицу.

Пример 1

Выведем на экран в столбик числа от 0 до 10.

```
var  
i:integer; {задаем целочисленную  
переменную i}  
begin  
for i:=0 to 10 do {открываем цикл в котором  
переменная i изменяет своё значение  
от 0 до 10}  
writeln(i) {выводим на экран  
значение переменной i}  
end.
```



The screenshot shows a window titled "Program1.pas" containing the following Pascal code:

```
var  
i:integer; |  
begin  
for i:=0 to 10 do  
writeln(i)  
end.
```

Below the code editor, the output of the program is displayed in a list box, showing the numbers 0 through 10, each on a new line.

Пример 1

Сравним две программы с циклом и без цикла:

```
var  
i:integer;  
begin  
  for i:=0 to 10  
    writeln(i)  
  end.
```

```
begin  
  writeln(0);  
  writeln(1);  
  writeln(2);  
  writeln(3);  
  writeln(4);  
  writeln(5);  
  writeln(6);  
  writeln(7);  
  writeln(8);  
  writeln(9);  
  writeln(10)  
end.
```

Обе программы выводят один и тот же результат, но количество строк и символов в программе с циклом гораздо меньше.

Вторая форма оператора

for переменная:=начальное значение **downto** конечное значение **do**
оператор

Отличие второй формы в том, что переменная уменьшает свое значение на единицу.

Значение начального и конечного значений переменного необходимо выбирать осмысленно.

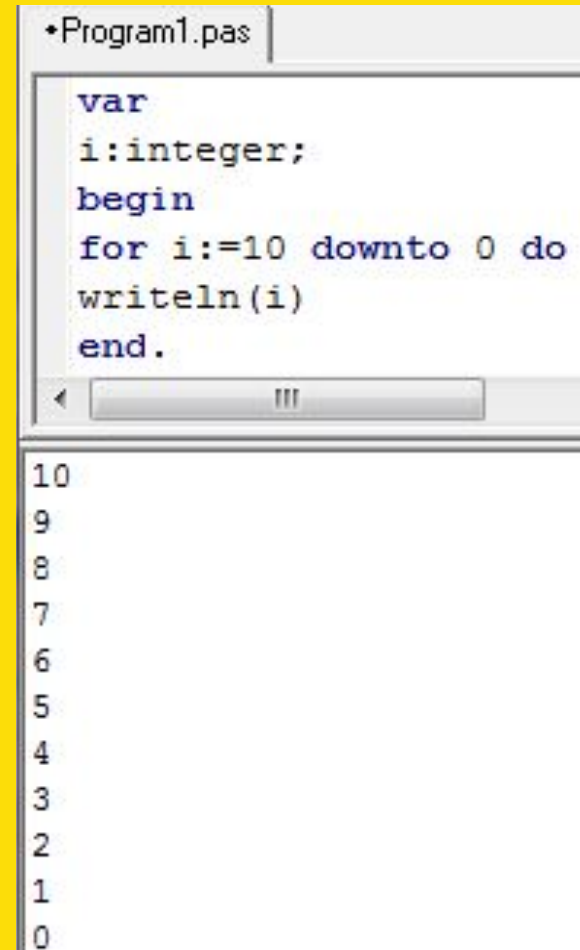
Так как в случае с первой формой записи оператора FOR начальное значение переменной должно быть **меньше либо равно** конечного значения переменной, иначе тело цикла не выполнится ни разу.

А в случае со второй формой записи оператора FOR начальное значение переменной должно быть **больше либо равно** конечного значения.

Пример 2

Выведем на экран в столбик числа от 10 до 0.

```
var  
i:integer; {задаем целочисленную  
переменную i}  
begin  
for i:=10 downto 0 do {открываем цикл в  
котором переменная i изменяет своё значение  
с 10 до 0}  
writeln(i) {выводим на экран  
значение переменной i}  
end.
```



The screenshot shows a window titled "Program1.pas" containing the following Pascal code:

```
var  
i:integer;  
begin  
for i:=10 downto 0 do  
writeln(i)  
end.
```

Below the code editor, the output of the program is displayed in a column, showing the numbers 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, and 0 from top to bottom.

Задача №1.

Написать программу возведения числа A в целую степень N ($N > 0$).

var

a, n, i, s : integer; {переменные: a -число, n -степень числа, i -счетчик, s -конечное значение}

begin

readln(a, n); {введем число и степень числа}

$s := 1$; {начальное значение переменной s равно 1}

for $i := 1$ to n do {цикл от 1 до значения переменной n }

$s := s * a$; {значение переменной s увеличивается за каждый проход цикла в a раз}

writeln(s) {выводим результат}

end.

Анализ задачи №1

Алгоритм решения задачи №1:

Для решения данной задачи необходимо, чтобы переменная s (начальное значения которой равно 1) умножалась на величину переменной a n -ое количество раз, и каждый раз принимало значение промежуточного результата.

Рассмотрим следующую таблицу для значений переменных $a=2$ и $n=4$:

Шаг цикла (значение переменной i)	Значение переменной s (выполнение $s:=s*a$)	Фрагмент программы
1	$S:=1*2=2$	for $i:=1$ to n do $s:=s*a$;
2	$S:=2*2=4$	
3	$S:=4*2=8$	
4	$S:=8*2=16$	

Когда переменная i присвоила значение равно 4 работа цикла закончилась (т.к. конечное значение переменной цикла равно 4). В итоге после четырех выполнений оператора ($s:=s*a$) значение переменной s стало равно значению a^n ($2^4=16$).

Задача №2.

Написать программу вычисления факториала заданного целого числа .

Факториал числа N вычисляется по следующей формуле:

$$N! = 1 * 2 * 3 * \dots * N$$

```
var
n,i,s:integer; {переменные:n-число, i-счетчик, s-конечное
значение}
begin
readln(n); {введем число}
s:=1; {начальное значение переменной s равно 1}
for i:=1 to n do {цикл от 1 до значения переменной n}
s:=s*i; {значение переменной s увеличивается за каждый проход
цикла в i раз, значение переменной i также увеличивается на
единицу}
writeln(s) {выводим результат}
end.
```

Анализ задачи №2

Алгоритм решения задачи №2:

Для решения данной задачи необходимо, чтобы переменная s (начальное значения которой равно 1) умножалась на величину переменной i n -ое количество раз, и каждый раз принимало значение промежуточного результата.

Рассмотрим следующую таблицу для значения переменной $n=4$:

Шаг цикла (значение переменной i)	Значение переменной s (выполнение $s:=s*i$)	Фрагмент программы
1	$S:=1*1=1$	for $i:=1$ to n do $s:=s*i$;
2	$S:=1*2=2$	
3	$S:=2*3=6$	
4	$S:=6*4=24$	

Когда переменная i присвоила значение равно 4 работа цикла закончилась (т.к. конечное значение переменной цикла равно 4). В итоге после четырех выполнений оператора ($s:=s*i$) значение переменной s стало равно значению $n!$ ($4!=24$).

Подведем итог

При выполнении оператора **FOR** сначала вычисляется выражение “начальное значение ” и осуществляется присваивание параметру цикла. **После чего циклически повторяется:**

- Проверка условия “параметр цикла” \leq “последнее значение”, если условие не выполнено оператор **FOR** завершает свою работу
- Если условие выполнено происходит выполнение тела цикла
- Далее наращивание переменной “параметр цикла” на единицу

Отметим два обстоятельства:

- Условие проверяется перед выполнением оператора “тело цикла”.
- Если условие не выполняется в самом начале, то используемый оператор не будет выполнен ни разу.
- Шаг наращивания параметра строго постоянен и равен 1.

Что мы узнали:

- Что такое цикл.
- Какие операторы цикла бывают в Pascal.
- Оператор счетного цикла FOR.
- Две формы оператора FOR.
- Как использовать оператор FOR.
- Решили несколько задач.