

24.04.20

Тема урока: Программирование циклов с заданным числом повторений

Сегодня мы продолжим изучать программирование циклов.

Посмотрите по ссылке урок:

https://www.youtube.com/watch?time_continue=20&v=9_pKcvr8uG4&feature=emb_logo



Пример 1. Дано целое положительное значение ***N***. Требуется вычислить сумму:

$$1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{N!}.$$

Ниже приводятся два варианта программы решения этой задачи. В первом варианте используется цикл с предусловием, во втором — цикл с постусловием.

```
Program Summa_1;
Var E,a: Real; N,i: Integer;
Begin
  Write('N='); ReadLn(N);
  E:=0; i:=0; a:=1;
  While i<=N Do
  Begin
    E:=E+a;
    i:=i+1;
    a:=a/i;
  End;
  WriteLn('E=', E);
End.

Program Summa_2;
Var E,a: Real; N,i: Integer;
Begin
  Write('N='); ReadLn(N);
  E:=0; i:=0; a:=1;
  Repeat
    E:=E+a;
    i:=i+1;
    a:=a/i;
  Until i>N;
  WriteLn('E=', E);
End.
```

Обратите внимание на то, как цикл с предусловием преобразуется в цикл с постусловием — условие цикла помещается после тела цикла и заменяется на противоположное.

Для программирования циклов с заданным числом повторений при постоянном шаге изменения параметра цикла в Паскале существует **цикл с параметром**. Вот как выглядит программа решения той же задачи с использованием цикла с параметром:

```
Program Summa_3;
Var E, a: Real; N, i: Integer;
Begin
  Write('N='); ReadLn(N);
  E:=1; a:=1;
  For i:=1 To N Do
  Begin
    a:=a/i;
    E:=E+a;
  End;
  WriteLn('E=', E);
End.
```

В программе используется оператор цикла For, для которого существуют два варианта:

1) **For** <параметр цикла>:=<выражение 1> **To** <выражение 2>

Do <оператор>

2) **For** <параметр цикла>:=<выражение 1> **Downto** <выражение 2>

Do <оператор>

Здесь <параметр цикла> — имя простой переменной порядкового типа. Выполнение оператора **For** в первом варианте (**To**) происходит по следующей схеме.

1. Вычисляются значения <выражения 1> и <выражения 2>. Это делается только один раз при входе в цикл.

2. Параметру цикла присваивается значение <выражения 1>.

3. Значение параметра цикла сравнивается со значением <выражения 2>. Если параметр цикла меньше или равен этому значению, то выполняется тело цикла (<оператор>), в противном случае выполнение цикла заканчивается.

4. Значение параметра цикла изменяется на следующее значение в его типе (для целых чисел — увеличивается на единицу); происходит возврат к пункту 3.

Оператор цикла **For** объединяет в себе действия, которые при использовании цикла **While** выполняют различные операторы: присваивание параметру начального значения, сравнение его с конечным значением, изменение значения параметра на следующее.

Во втором варианте оператора **For** слово **Downto** буквально можно перевести как «вниз до». В таком случае параметр цикла изменяется по убыванию, т. е. при каждом повторении цикла параметр изменяет свое значение на предыдущее (равносильно $i := \text{pred}(i)$).

Работая с оператором **For**, учитывайте следующие правила:

(Правила записать в тетрадь)

- параметр цикла не может иметь вещественного типа;
- в теле цикла нельзя изменять переменную-параметр цикла;
- при выходе из цикла значение переменной-параметра является неопределенным.

Рассмотрим пример программы, в которой в теле цикла будет присутствовать ветвление.



Пример 2. Составим программу проверки знаний учеником таблицы умножения. Компьютер задает ученику 10 вопросов на умножение чисел от 2 до 9. На каждое задание ученик вводит свой ответ, компьютер сообщает, верный ответ или нет.

На рисунке 3.16 приведена блок-схема такого алгоритма.

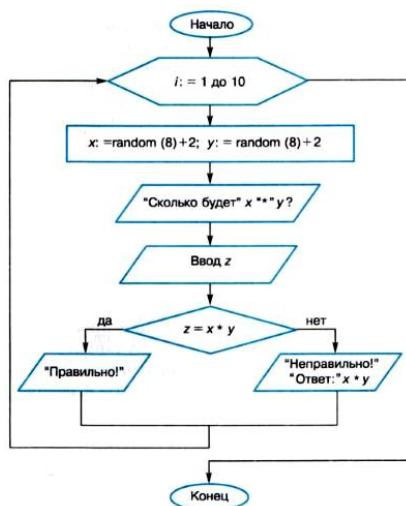


Рис. 3.16. Блок-схема алгоритма из примера 2

Обратите внимание на то, как отображается на блок-схеме цикл с параметром.

В этом алгоритме использована функция `random(x)`, результатом выполнения которой является случайное целое число из диапазона от 0 до $x - 1$. Следовательно, выражение `random(8)+2` принимает случайные значения от 2 до 9. Функция `random` называется датчиком случайных чисел.

На Паскале этот алгоритм программируется так:

```

Program TablMul;
Begin
  Var x,y,i,z: Integer;
  randomize;
  For i:=1 To 10 Do
    Begin
      x:=random(8)+2;
      y:=random(8)+2;
      WriteLn('Сколько будет ', x, '*', y, '?');
      Read(z);
      If z=x*y
      Then WriteLn('Правильно!')
      Else WriteLn('Неправильно! ', x, '*', y, '=', x*y);
    End;
  End.

```

А вот фрагмент интерфейса исполнения этой программы:

*Сколько будет 4*8?*
21
*Неправильно! 4*8=32*
*Сколько будет 6*9?*
54
Правильно!

В программе используется стандартная процедура `randomize`. Ее исполнение производит установку случайного начального состояния датчика случайных чисел. Благодаря этому при повторном выполнении программы будут получаться разные последовательности случайных чисел.

Запишите таблицу «Система основных понятий» в тетрадь.

Система основных понятий

Программирование циклов		
Операторы цикла		
Цикл-пока	Цикл с параметром	Цикл-до
While <логическое выражение> Do <оператор> <оператор> — тело цикла. Цикл повторяет выполнение, пока истинно <логическое выражение>	Параметр — переменная порядкового типа 1) For <параметр цикла>:= <выражение 1> To <выражение 2> Do <оператор> — по возрастанию параметра 2) For <параметр цикла>:= <выражение 1> Downto <выражение 2> Do <оператор> — по убыванию параметра	Repeat <оператор> Until <логическое выражение> <оператор> — тело цикла. Повторяется выполнение тела цикла до того, как <логическое выражение> станет истинным
Цикл с заданным числом повторений имеет управляющий параметр, изменяющийся с постоянным шагом в определенном диапазоне значений; реализуется всеми типами операторов цикла		

Конспект пришлите на проверку учителю.