**Программы с циклической структурой.**

## 1. Операторы цикла

 Операторы цикла используются для вычислений, повторяющихся многократно. В паскале имеется три вида циклов: цикл с предусловием **while**, цикл с постусловием **repeat** и цикл с параметром **for**. Каждый из них состоит из определенной последовательности операторов.

 Блок, ради выполнения которого и организуется цикл, называется *телом цикла*. Один проход цикла называется *итерацией*.

 *Начальные установки* служат для того, чтобы до входа в цикл задать значения переменных, которые в нем используются.

 Проверка *условия* продолжения цикла выполняется на каждой итерации либо до тела цикла, либо после тела цикла (цикл *с постусловием*). Разница между ними состоит в том, что тело цикла с постусловием всегда выполняется хотя бы один раз, после чего проверяется, надо ли его выполнять еще раз. Проверка необходимости выполнения цикла с предусловием делается до тела цикла, поэтому возможно, что он не выполнится ни разу.

 *Параметром цикла* называется переменная, которая используется при проверке условия цикла и принудительно изменяется на каждой итерации, причем, как правило, на одну и ту же величину(цикл *с параметром*). Если параметр цикла целочисленный, он называется *счетчиком цикла.* Количество повторений такого цикла можно определить заранее. Параметр есть не у всякого цикла. В так называемом *итеративном* цикле условие продолжения содержит переменные, значения которых изменяются в цикле по рекуррентным формулам.

 Цикл завершается, если условие его продолжения не выполняется. Возможно принудительное завершение как текущей итерации, так и цикла в целом. Для этого служат процедуры break, continue и оператор goto. Передавать управление извне внутрь цикла не рекомендуется, потому что при этом могут не выполниться начальные установки.

### 1.1 Цикл с предусловием *while*

Формат оператора прост:

 *while выражение do оператор*

 Выражение должно быть логического типа. Например, это может быть операция отношения или просто логическая переменная. Если результат вычисления выражения равен true, выполняется расположенный после служебного слова do простой или составной оператор2. Эти действия повторяются до того момента, пока результатом выражения не станет значение false. После окончания цикла управление передается па следующий за ним оператор.

 Если в теле цикла требуется выполнить более одного оператора, необходимо заключить их в блок с помощью ключевых слов *begin* и *end*.

*Пример.* При помощи цикла вывести слово «привет» 5 раз.

**Текст программы**

program pr;

var x:string;

i:byte;

begin

i:=1;x:='привет'; {Начальные параметры}

while i<=5 do begin{Заголовок цикла}

writeln(x); {Вывод текста}

inc(i); {увеличение на единицу}

end;

end.

### 1.2 Цикл с постусловием *repeat*

 Тело цикла с постусловием заключено между служебными словами repeat и until, поэтому заключать его в блок не требуется.

repeat

тело цикла

until выражение

 В отличие от цикла while этот цикл будет выполняться, пока логическое выражение после слова until *ложно.* Как только результат выражения станет истинным, произойдет выход из цикла. Вычисление выражения выполняется в конце каждой итерации цикла.

 Этот вид цикла применяется в тех случаях, когда тело цикла необходимо обязательно выполнить хотя бы один раз: например, если в цикле вводятся данные и выполняется их проверка. Если же такой необходимости нет, предпочтительнее пользоваться циклом с предусловием.

*Пример.* При помощи цикла вывести слово «привет» 5 раз.

**Текст программы**

program pr;

var x:string;

i:byte;

begin

i:=0;x:='привет'; {Начальные параметры}

repeat

writeln(x); {Вывод текста}

inc(i); {увеличение на единицу}

until i=5

end.

### 1.3 Цикл с параметром *for*

 Этот оператор применится, если требуется выполнить тело цикла заранее заданное количество раз. Параметр порядкового типа на каждом проходе цикла автоматически либо увеличивается, либо уменьшается на единицу.

for <параметр>:=<начальное значение> to <конечное значение> dо <оператор>

for <параметр>:=<конечное значение> downto <начальное значение> do <оператор>

 Выражения должны быть того же типа, что и переменная цикла1, оператор -- простым или составным.

*Пример 1.* Программа выводит на экран в столбик числа от 1 до 10.

var i:integer;

begin

for i:=1 to 10 do writeln (i);

end.

 Цикл будет выполнен 10 раз на каждом проходе счетчик цикла переменная i увеличивается на 1.

*Пример 2.* Программа выводит на экран в столбик числа от 10 до 1 и подсчитывает их сумму.

var i,sum:integer;

begin

sum:=0;

for i:=10 downto 1 do

begin

writeln(i);

sum:=sum+i;

end;

writeln('сумма чисел=',sum);

end.

В этом цикле переменная 1 автоматически уменьшается на 1.

*Пример 3.* Программа выводит на экран символы от 'a' до 'z'.

var c:char;

begin

for c:='a' to 'z' do write(c:3);

end.

 Здесь счетчик цикла *c* символьного тина поочередно принимает значение каждого символа от 'a' до 'z'*.*

 Если в теле цикла требуется выполнить более одного оператора, необходимо заключить их в блок с помощью ключевых слов begin и end.