Когда известно число повторений удобно использовать инструкцию <u>FOR</u>. Например, таблица умножения или вычисление значений функции в нескольких различных отстоящих на равное расстояние друг от друга точках (построить таблицу значений функции).

Оператор цикла ПЕРЕСЧЕТ

В общем виде инструкция <u>FOR</u> записывается так:

for ***счетчик*****:=*****нач_значение*** **to** ***кон_значение*** **do**

begin

{

end;

где

*** счетчик - имя переменной-счетчика числа повторений операторов цикла.**

*** нач_значение - выражение, определяющее начальное значение переменной-счетчика циклов**

*** кон_значение - выражение, определяющее конечное значение переменной счетчика циклов**

*Примечание: Если между **begin** и **end** находится только один оператор, то слова **begin** и **end** можно не писать.*

Примеры:

```
1.  For i:= 1 to 10 do
      write('-');
2. For i:= 1 to 4 do
    begin
        x:=x+5;
        y:=2*x*x-10;
        writeln(x,'  ',y)
    end;
```

Обычно в качестве выражений, определяющих значения начального и конечного состояния счетчика циклов, используют переменные или константы. В этом случае последовательность операторов, находящаяся между *begin* и *end*, будет выполнена (кон_значение - нач_значение + 1) раз.

Переменная-счетчик должна быть целого типа (integer).

Следующая программа вычисляет сумму пяти элементов вводимых с клавиатуры:

```
Program Summa;
var Sum, element:real;
     i:integer;
begin
   Sum:=0;
   for i:=1 to 5 do
      begin
         writeln(‘введите число ’);
         readln(element);
         Sum:=Sum+element
      end;
  write(‘Сумма элементов = ’, Sum);
  readln
end.
```