**Добрый день, 16 группа!**

Продолжаем общаться дистанционно.

Сегодня мы рассмотрим условия и циклы в алгоритмических конструкциях

Задать вопросы, а также прислать ответы вы можете

1. на адрес электронной почты: [ddrmx@ya.ru](mailto:ddrmx@ya.ru)
2. через соцсеть <https://vk.com/ddrmx>

С уважением, Максим Андреевич.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ ПО ТЕМЕ:

Условия, циклы. (1 ЧАС)

***Краткие теоретические сведения.***

Если в программе возникает необходимость неоднократно выполнить некоторые операторы, то используются *операторы повтора (цикла).* В языке Паскаль различают три вида операторов цикла: *while, for*. Они используются для организации циклов различных типов. Выражение, управляющее повторениями, должно иметь булевский тип.

Если число повторений оператора (составного оператора) заранее неизвестно, а задано лишь условие его повторения (или окончания), используются оператор *while*. Оператор *for* используется, если число повторений заранее известно.

**Оператор цикла for**

В случаях, когда число повторений может быть заранее известно, для организации циклической обработки информации применяется оператор повтора *for.* Часто этот оператор повтора называют *оператором цикла с параметром,* так как число повторений задается переменной, называемой *параметром цикла*, или управляющей переменной. Оператор повтора *for* состоит из *заголовка* и *тела цикла*.

Он может быть представлен в двух форматах:

**for <параметр цикла> := <S1> +1 <S2> do <оператор>;**

**for <параметр цикла> := <S1> -1 <S2> do <оператор>;**

где *Sl* и *S2* — выражения, определяющие соответственно начальное и конечное значения параметра цикла;

for ... do — заголовок цикла;

<оператор> — тело цикла.

Значение управляющей переменной изменяется на +1 или –1

Тело цикла может быть простым или составным оператором. Оператор *for* обеспечивает выполнение тела цикла до тех пор, пока не будут перебраны все значения параметра цикла от начального до конечного.

**Оператор цикла while**

Оператор *while (пока)* часто называют *оператором цикла с предусловием* за то, что проверка условия выполнения тела цикла производится в самом начале оператора.

Формат записи:

**while <условие продолжения повторений>**

**<тело цикла>;**

*Условие* - логическое выражение, *тело цикла* - простой или составной оператор.

Перед каждым выполнением тела цикла вычисляется значение выражения условия. Если результат равен *True,* тело цикла выполняется и снова вычисляется выражение условия. Если результат равен *False*, происходят выход из цикла и переход к первому после *while* оператору.

Цикл*while* может не выполниться ни разу, если при первой проверке условие оказалось ложным (*False).*

Домашнее задание:

### Содержание отчета

1. Операторы цикла. Общая характеристика.
2. Оператор цикла for. Форматы записи, описание работы цикла, ограничения использования параметра цикла.
3. Оператор цикла while. Формат записи, описание работы цикла.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ ПО ТЕМЕ:

Разработка алгоритма. (1 ЧАС)

**Пример 1.**

Вывести квадраты первых десяти натуральных чисел.

**Решение**

Используемые переменные:i –натуральные числа, x– их квадраты

Program pr1;

Var i, x: integer;

begin

for i:=1 to 10 do begin *{перебираем натуральные числа от 1 до 10}*

x:=sqr(i); *{возводим очередное число в квадрат}*

write(x, ' '); *{выводим полученное значение}*

end;

end.

**Результат**

1 4 9 16 25 36 49 64 81 100

**Пример 2.**

Найти сумму 1 + 1/3 + 1/5 +...(N слагаемых).

**Решение**

Program pr2;

Var I, N: integer;

S: real;

begin

Write('N='); Readln(N); *{вводим количество слагаемых}*

S:=0; *{обнуляем сумму}*

*{выполняем цикл N раз, добавляя к сумме по одному слагаемому}*

For I:=1 to N do

S:=S+1/(2\*I-1);

Writeln('S=',S:5:2); *{выводим результат с двумя десятичными знаками}*

end.

**Результат:**

N=4

S= 1.68

**Пример 3.**

Дано натуральное число n. Вычислить 31+32+…...+3n

**Решение**

program pr3;

var i,n:integer;

s:real;

begin

write('введите n ');

readln(n);

s:=0;

for i:=1 to n do

s:=s+ exp(i\*ln(3));

writeln('сумма=',s:4:0);

readln;

end.

**Пример 4.**

Задана арифметическая прогрессия -21; - 16;… Определить номер первого положительного члена прогрессии.

a – очередной член прогрессии, n – его порядковый номер

**Решение**

Program Pr9;

var

a, n: integer;

begin

a:= -21; n:=1; *{задаем начальные значения}*

while a<=0 do begin *{пока очередной член прогрессии меньше либо ра*

a:=a+5; n:=n+1; *равен 0,* *вычисляем следующий член прогрессии и* end; *и его порядковый номер}*

writeln(‘n=’, n); *{выводим номер 1-го положительного члена*

*прогрессии}*

end.

**Результат:**

n=6

**Пример 5**. Первоначальный вклад составил S рублей. Через сколько лет сумма вклада более, чем в 2 раза превысит первоначальный вклад, если годовой процент составляет x%.

Например, S=1000 р, x=10%

1 год S=1000+1000\*10/100=1100

2 год S=1100+1100\*10/100=1210

**Решение**

Program pr10;

Var s, sum, x: real;

n: integer;

Begin

Write(‘S=’); Readln(S);

Write(‘x=’); Readln(x);

Sum:=2\*s; n:=0;

While S<=Sum do begin

s:=s+s\*x/100;

n:=n+1;

end;

write('через ', n, ‘ лет’);

end.