Организация циклов в языке программирования Паскаль.

Циклом называется последовательность многократно повторяющихся действий.

- При решении алгоритмических задач может возникнуть необходимость повторить одни и те же действия несколько раз. Такие действия называют циклическими или циклами.
- Процесс называется циклическим, если вычисления в нём многократно повторяются по одним и тем же формулам при разных значениях входящих в них переменных.
- Алгоритмическая структура “Цикл” обеспечивает многократное выполнение некоторой последовательности действий, которая называется ***телом цикла***.

В Паскале различают циклы:

- 1) **с неопределенным числом повторений**; здесь различают цикл с пред-условием (цикл “Пока”) и цикл с пост-условием (цикл “До”).
- 2) **с определённым числом повторений** (цикл с параметром, или цикл со счетчиком, или цикл “Для”);

В Паскале существуют ***3 вида циклов***:

- цикл с пред-условием
- цикл с пост-условием
- цикл с параметром

ОТЛИЧИЕ - СПОСОБ ПРОВЕРКИ ОКОНЧАНИЯ ЦИКЛА.

Операторы цикла

1. Цикл с пред-условием

While <условие> **do**

Begin

<оператор 1>;

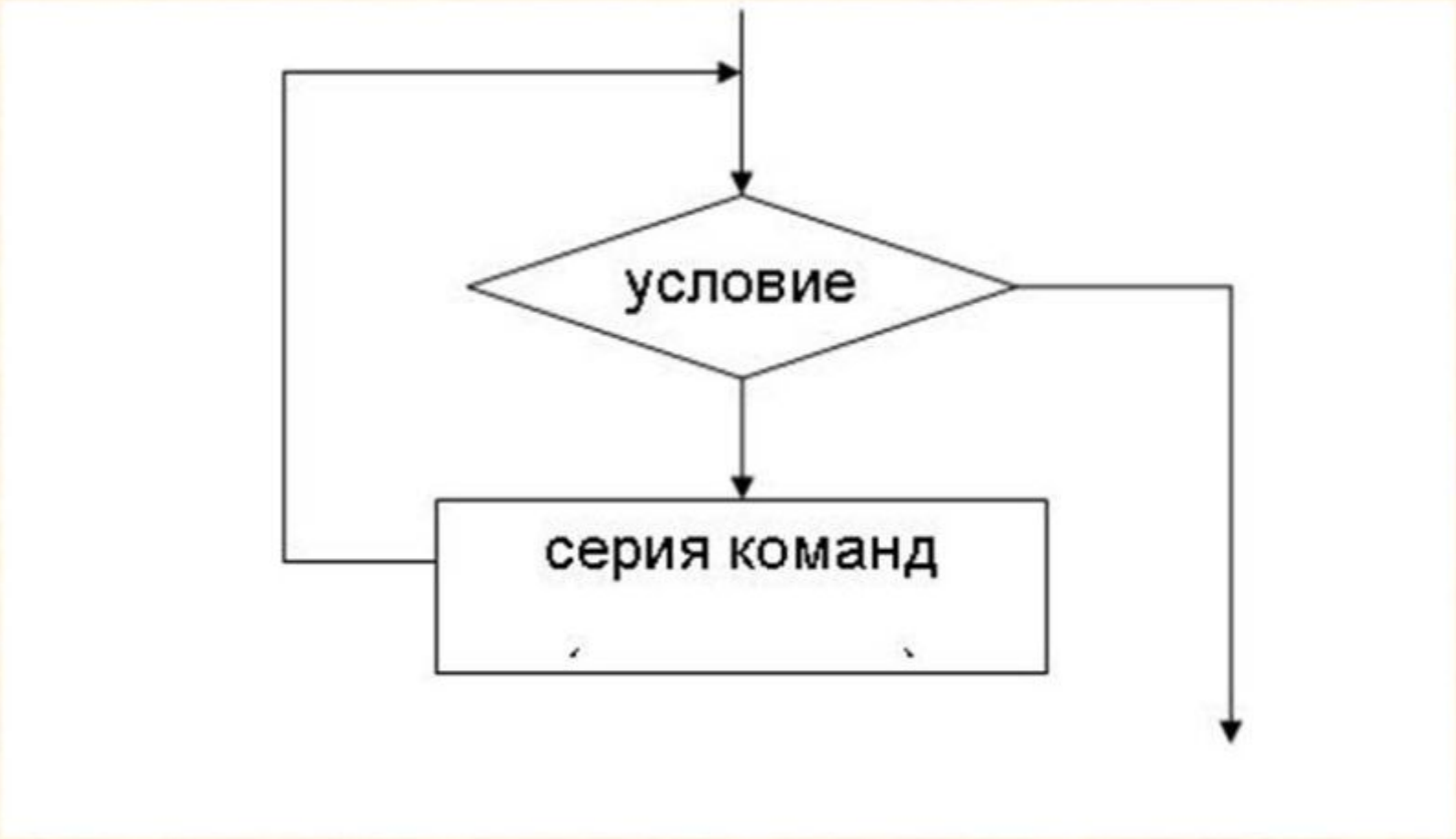
<оператор 2>;

<оператор 3>;

. . .

End;

Блок-схема цикла While



Замечания:

- Если в теле цикла записано более одной команды, то тело цикла заключается в операторные скобки begin и end.
- Число повторений инструкции цикла while определяется ходом выполнения программы.
- Инструкция цикла while выполняется до тех пор, пока условие, записанное после слова while, истинно.
- После слова while надо записывать условие продолжения (выполнения) цикла.
- Для завершения цикла while в теле цикла обязательно должны присутствовать инструкции, влияющие на условие выполнения инструкций цикла.
- Цикл while – цикл с предусловием, т.е. инструкции тела цикла вообще могут быть не выполнены.
- В цикле while проверка условия выхода из цикла выполняется в начале цикла.

Пример: Найти сумму всех чётных чисел от 20 до 40.

Введём обозначения: 1) формула суммы S:=S+слагаемое, где начальное значение суммы S=0;

2) получение очередного чётного числа K:=K+2, где начальное чётное число K=20

Программа на Паскале:

```
Program Primer_1;
Var K,S: integer;
begin
S:=0;  K:=20;
   While K<=40 do
     begin
            S:=S+K;
            K:=K+2;
     end;
    writeln('S= ',S);
Readln;
end.
```

Операторы цикла

2. Цикл с пост-условием

Repeat

<оператор 1>;

<оператор 2>;

<оператор 3>;

. . .

Until <условие>;

Блок-схема цикла Repeat

Замечания:

- Зарезервированные слова repeat и until обрамляют тело цикла, поэтому дополнительные пары операторов begin и end уже не обязательны.
- Число повторений цикла repeat определяется ходом выполнения программы.
- Цикл repeat выполняется до тех пор, пока условие, стоящее после слова until, ложно.
- После слова until записывается условие завершения цикла (оно противоположно для условия продолжения цикла).
- Для завершения цикла repeat в теле цикла обязательно должны быть инструкции, выполнение которых влияет на условие завершения цикла.
- Цикл repeat – цикл с постусловием, т.е. тело цикла выполняется хотя бы один раз.

Пример: Найти сумму всех чётных чисел от 20 до 40.

Пример – тот же, но с циклом Repeat:

Фрагмент программы:

```
Program Primer_2;
Var K,S: integer;
begin
S:=0;  K:=20;
     Repeat
        S:=S+K;
        K:=K+2;
     Until K>40;
    writeln('S= ',S);
Readln;
end.
```

Операторы цикла

3. Цикл с параметром.

Синтаксическая запись цикла с параметром (цикл “Для”) имеет два варианта:

1) с возрастанием переменной цикла:

For <переменная>**:=**<начальное значение> **to** <конечное значение> **do**

Begin

<оператор 1>;
<оператор 2>;
<оператор 3>;

. . .

End;

2) с убыванием переменной цикла:

For <переменная>**:=**<конечное значение> down**to** <начальное значение> **do**

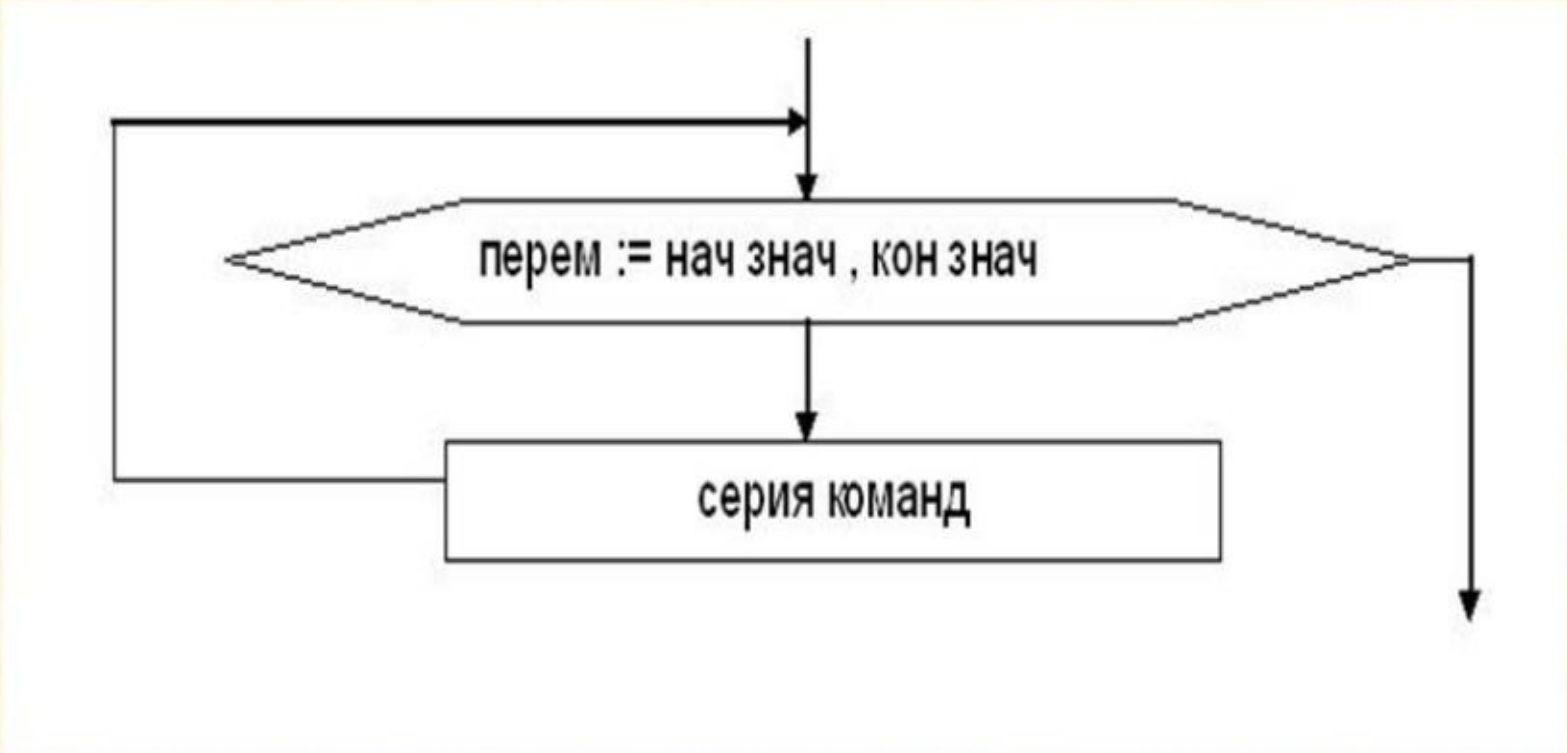
Begin

<оператор 1>;
<оператор 2>;
<оператор 3>;

. . .

End;

Блок-схема цикла For



Замечания:

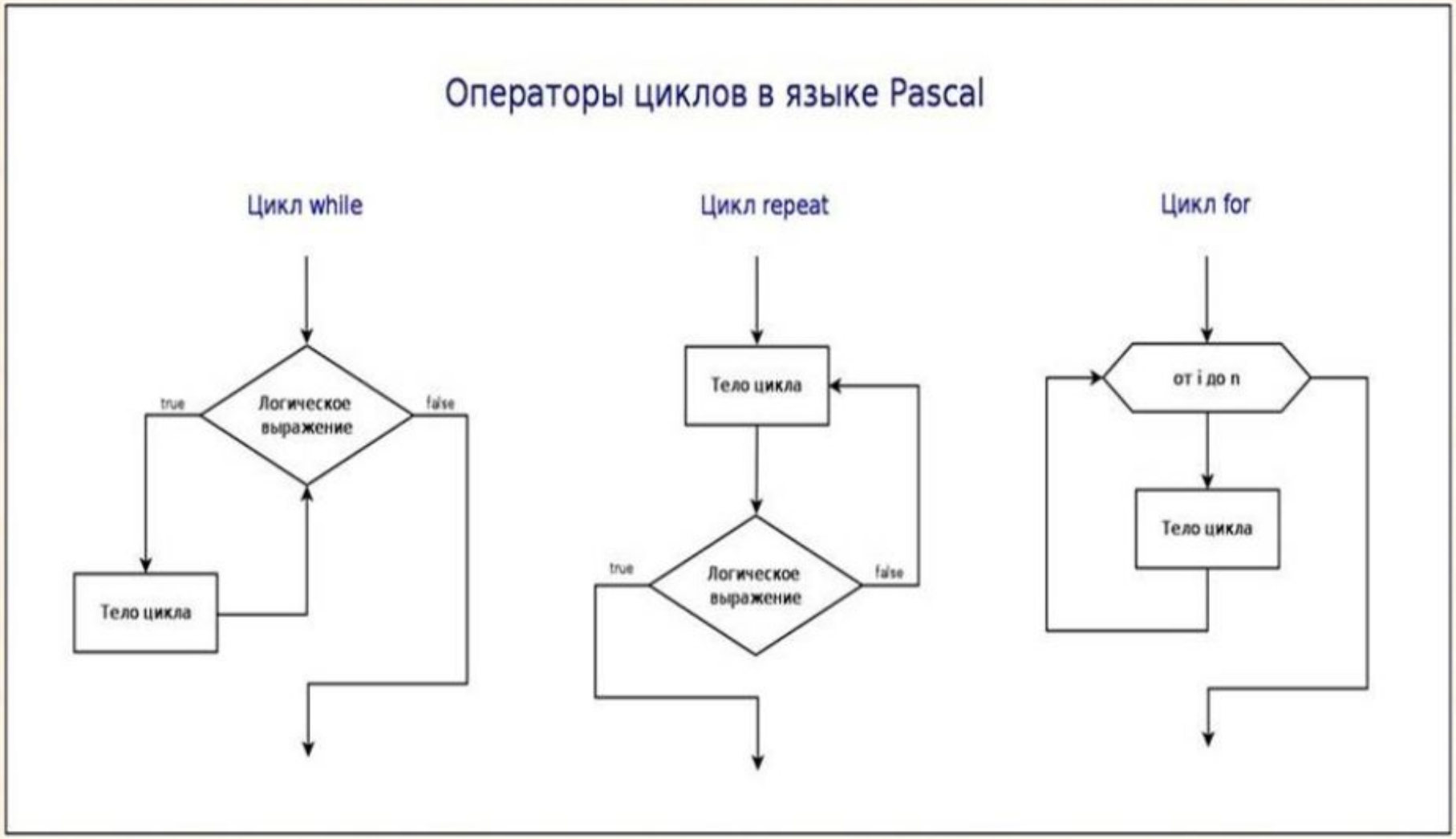
- Цикл For используется для организации циклов с фиксированным, определяемым во время разработки программы, числом повторений.
- Количество повторений цикла определяется начальным и конечным значениями переменной-счётчика (т.е. параметром).
- Переменная – счётчик должна быть целого типа.
- Идентификатор(переменная), начальное и конечное значение переменной должны быть одного и того же порядкового типа.
- Оба выражения (начальное и конечное значение переменной) вычисляются перед выполнением оператора цикла и не должны изменяться внутри оператора цикла.
- После окончания цикла значение параметра цикла не определено, т.е. нельзя считать, что значение параметра равно значению второго выражения.

Пример: Напечатать таблицу квадратов чисел от 20 до 30.

```
Program Kvadrat;
Var X,Y: integer;
Begin
    Write(‘  X ’, ‘   Y ‘); writeln;
    For X:=20 to 30 do
    Begin
            Y:=X*X; Writeln(X:4, Y:4);
     End;
     Readln;
End.
```

Примечание: на вывод значений X и Y отводится по 4 позиции.

Общая схема циклов в Паскале



Задания для практической части

Пример 1*:** ***Вычислить произведение первых N целых положительных чисел. Количество чисел вводится с клавиатуры.

Пример 2 : ***Напечатать таблицу стоимости порции сыра весом от 100 до 1000 граммов с шагом 50 граммов. Цена килограмма сыра задаётся с клавиатуры в процессе исполнения программы.***

Пример 3 : ***Вычислить сумму ряда чисел***

$$\frac{1}{2}+\frac{2}{3}+\frac{3}{4}+\ldots+\frac{599}{600}$$

***Пример 1*: Вычислить произведение первых N целых положительных чисел. Количество чисел вводится с клавиатуры.**

```
Program Proizvedenie;
Var N,X,P: integer;
Begin
    Writeln(‘ введите количество чисел N ‘); readln(N);
    P:=1; X:=1;
    While X<=N do
        Begin P:=P*X; X:=X+1; end;
    Writeln(‘произведение‘, N, ‘ чисел равно ‘, P);
    Readln;
End.
```

Пример 2: **Напечатать таблицу стоимости порции сыра весом от 100 до 1000 граммов с шагом 50 граммов. Цена килограмма сыра задаётся с клавиатуры в процессе исполнения программы.**

```
Program Syr;
Var m: integer; t,s: real;
Begin
      Writeln('Введите цену за килограмм сыра'); readln(t);
       Writeln(' Масса ', '  Стоимость ');
       m:=100;
       Repeat
       S:=m*t/1000;
       Writeln(m:6,s:8:2);
       m:=m+50;
        Until m>1000;
       Readln;
End.
```

Примечание: стоимость каждой порции сыра в граммах выводится с новой строки и в формате руб.,коп.

Пример 3 : Вычислить сумму ряда чисел $\frac{1}{2}+\frac{2}{3}+\frac{3}{4}+\ldots+\frac{599}{600}$

```
Program sum;
   var a, s: integer;
Begin
      writeln (‘ вычисление суммы ряда’);
      a:=1; s:=0;
      while a<600 do
            begin
            s:=s+a/(a+1); a:=a+1;
            end;
      writeln (‘сумма ряда равна s=‘, s);
      readln;
end.
```

Вопросы для самоконтроля

- Для чего используются команды цикла?
- Как в цикле записать команду присваивания, чтобы она вычисляла сумму чисел? Количество?
- Какое начальное значение должно быть у переменной, в которой накапливаем сумму чисел?
- Можно ли изменять значение переменной цикла в теле цикла?
- Можно ли изменять значение верхней границы цикла в теле цикла?
- Какие трудности возникли у вас в ходе решения задач?
- Что вы рекомендуете предпринять для преодоления этих трудностей?

самостоятельного выполнения:

1. НАЙТИ СУММУ КВАДРАТОВ ВСЕХ НАТУРАЛЬНЫХ ЧИСЕЛ ОТ 1 ДО 100.

2. ВЫЧИСЛИТЬ СУММУ РЯДА ЧИСЕЛ 2+4+6+…+1000.

3. ВЫЧИСЛИТЬ СУММУ РЯДА ЧИСЕЛ

$$\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\ldots+\frac{1}{600}$$

ПРИМЕЧАНИЕ: РЕШИТЬ ЭТИ ЗАДАЧИ С ИСПОЛЬЗОВАНИЕМ ВСЕХ ТРЕХ ВИДОВ ЦИКЛОВ. Всего должно быть 9 программ.