

Программирование алгоритмов циклической структуры

Тараненко Светлана Николаевна
Учитель информатики и ИКТ
МОБУ СОШ № 22 г. Благовещенска

2) Запишите математическое выражение, используя, оператор присваивания и стандартные функции ($sqr(x) - x^2$, $sqrt(x) - \sqrt{x}$, $abs(x) -$ модуль x)

$$M(d) = \frac{1}{5d} + d^2$$

2) *Запишите результат выполнения фрагмента программы*
(div – операция целочисленного деления, mod – операция деления с остатком)

```
Begin  
a := 18 mod 5;  
b := 21 div 3;  
c := b - a;  
write ('c=',c);  
k:=5;  
r:=k-2;  
k:=r+5;  
r:=k-2;  
write (r,k);  
end;
```

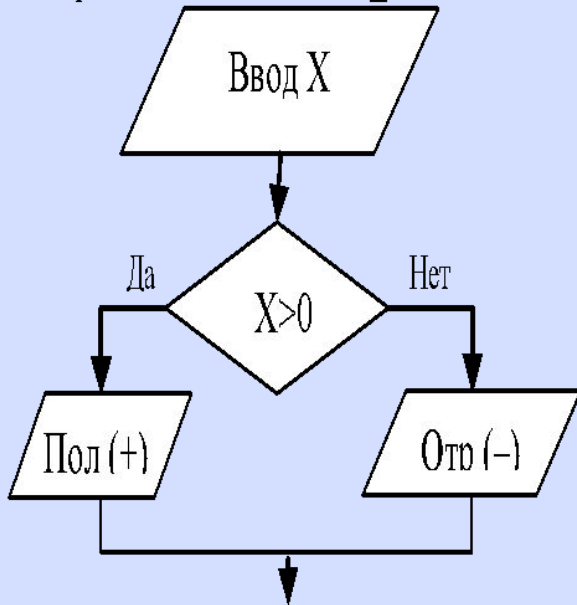
Ответ: _____

3. Выделите неправильно записанные условные операторы из списка, по возможности укажите место ошибки, либо исправьте её:

- а) If a>b theh write(b);
- б) If k<>m then write(m)elze write(k);
- в) If s>=5 then s:=s+5
- г) If a<b then c:=false; else:=true;
- д) If a=b else p:=p+1 then p:=p-1;

4. По блок-схеме определите назначение алгоритма и
допишите фрагмент программы

Алгоритм для задачи zVet_2



Ответ: _____

```
Program zVet_2;  
Var x:real;  
BEGIN  
  Readln(____);  
  If _____  
  then writeln ('_____')  
  else writeln ('_____');  
END.
```

начало

```
graph TD; A([начало]) --> B[Испекла бабушка Колобок и поставила его на окно остужать]; B --> C[Убежал Колобок от деда и бабушки и покатился по дорожке]; C --> D[...]; D --> E[Что привело Колобка к трагической гибели?];
```

Испекла бабушка Колобок и поставила его на окно остужать

Убежал Колобок от деда и бабушки и покатился по дорожке

...

Что привело Колобка к трагической гибели?

Программирование алгоритмов циклической структуры

Циклический алгоритм – это алгоритм команды которого выполняются несколько раз подряд.

В языке Паскаль имеется три различных оператора цикла:

1. Оператор цикла с параметром.
2. Оператор цикла с предусловием.
3. Оператор цикла с постусловием.

Оператор цикла с параметром.

Оператор цикла с параметром применяют тогда, когда заранее известно число повторений одной и той же последовательности операторов. Начальные и конечные значения параметра цикла могут быть представлены константами, переменными или арифметическими выражениями.

FOR *K:=M1* TO *M2* DO оператор1



FOR K:=M1 TO M2 DO

BEGIN

операторы циклической части

END;

FOR K:=M1 DOWNTO M2 DO

BEGIN

операторы циклической части

END;

ГДЕ *K* - параметр цикла, *M1* и *M2* - начальное и конечное значение параметра цикла. ***DOWNTO*** изменяет параметр с шагом -1.



Внимание !

- 1) Счетчик изменяется на единицу при каждом следующем исполнении оператора цикла.
- 2) Не забывайте, что счетчику присваивается начальное значение, т.е. $K = \text{начальное значение}$.
- 3) Если начальное значение совпадает с конечным значением, то операторы цикла (еще говорят «тело цикла») выполняются один раз.
- 4) Если начальное значение счетчика больше конечного значения, то тело цикла не выполняется ни разу.
- 5) При выходе из цикла значение счетчика совпадает с конечным значением, т.е. $K = M2$



Пример программы (цикл с параметром):

Вычислить сумму $S=1+2+3+\dots+N$:

```
Program zadacha;  
var n,s, i : integer;  
Begin  
  write('введите число N '); readln(n);  
  s:=0;  
  for i:=1 to n do  
    s:=s+i;  
  writeln('сумма чисел равна', s);  
End.
```



Блок-схема цикла с параметром:



Оператор цикла с предусловием

*Цикл с предусловием WHILE
используется тогда, когда число
повторений операторов цикла заранее не
известно, а задается некоторым
условием.*

WHILE <условие (логическое выражение) > **DO**
BEGIN
тело цикла
END;



**WHILE (пока) <условие> DO (выполнять)
BEGIN <тело цикла > END;**

- 1. Сначала проверяется значение условия. Пока оно «истинно» выполняются операторы циклической части. Когда значение условия «ложно» происходит выход из цикла.*
- 2. Можно в тело цикла поставить операторы перехода (EXIT и GOTO). Они передают управление за пределы цикла.*
- 3. Этот цикл может работать как цикл с параметром, но при этом необходимо изменять переменную, являющуюся счетчиком цикла, с помощью оператора присваивания.*
- 4. !!! Если при первой проверке условие «ложно», то цикл не выполнится ни разу.*



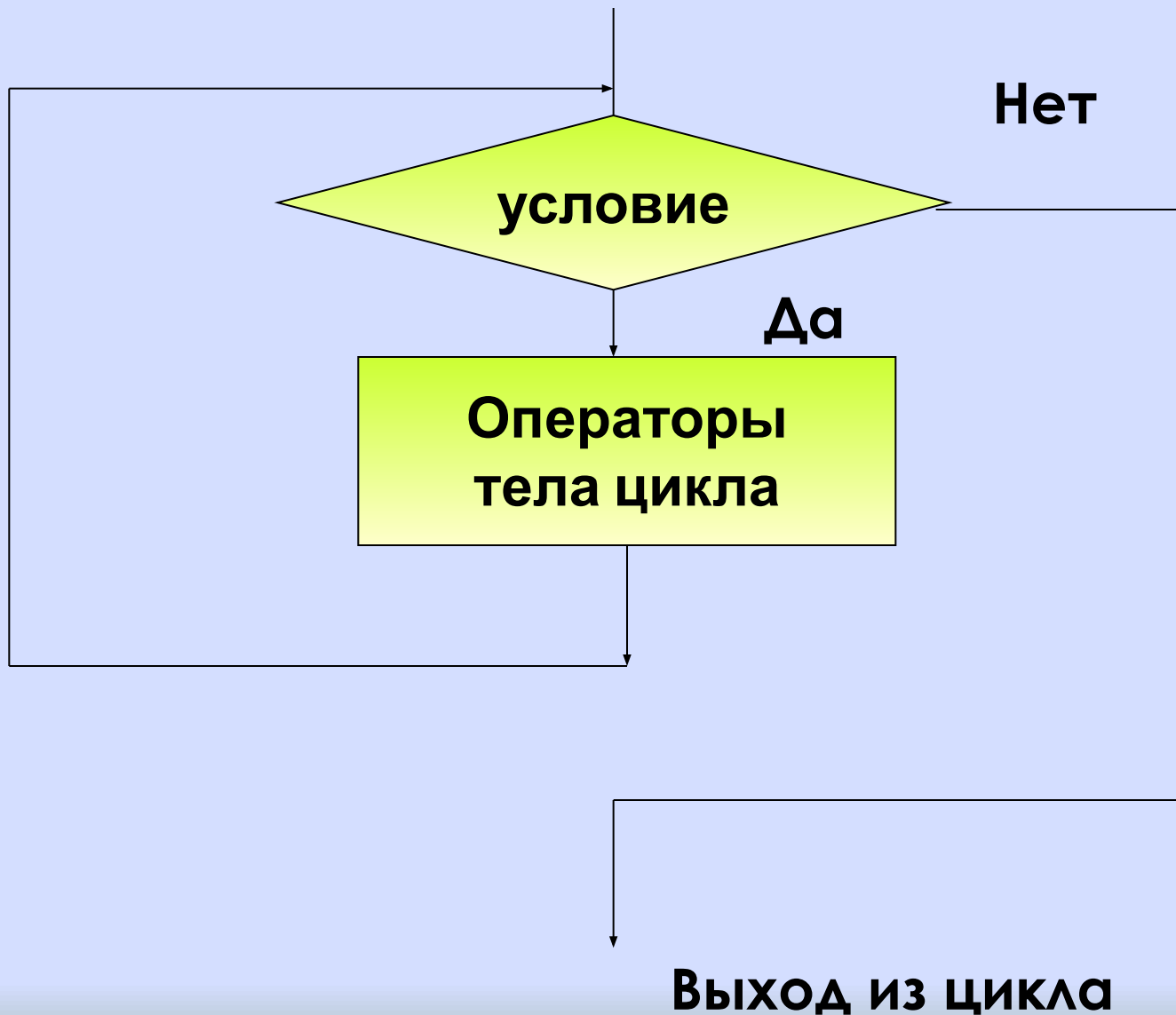
Пример программы (цикл с предусловием):

Вычислить сумму $S=1+2+3+\dots+N$:

```
Program zadacha;  
var n,s, i : integer;  
Begin  
  write('введите число N '); readln(n);  
  s:=0; i:=1;  
  While i<= n do  
    begin  
      s:=s+i;  i:=i+1;  
    end;  
  writeln('сумма чисел равна', s);  
End.
```



Блок-схема цикла с предусловием:



Оператор цикла с постусловием.

Этот оператор отличается от цикла с предусловием While тем, что проверка условия производится после очередного выполнения тела цикла. Это обеспечивает выполнение цикла хотя бы один раз.

REPEAT

тело цикла

UNTIL *<условие (логическое выражение)>;*

Операторы тела цикла выполняются до тех пор пока значение условия ложно.

Условием прекращения циклических вычислений является истинное значение логического выражения.



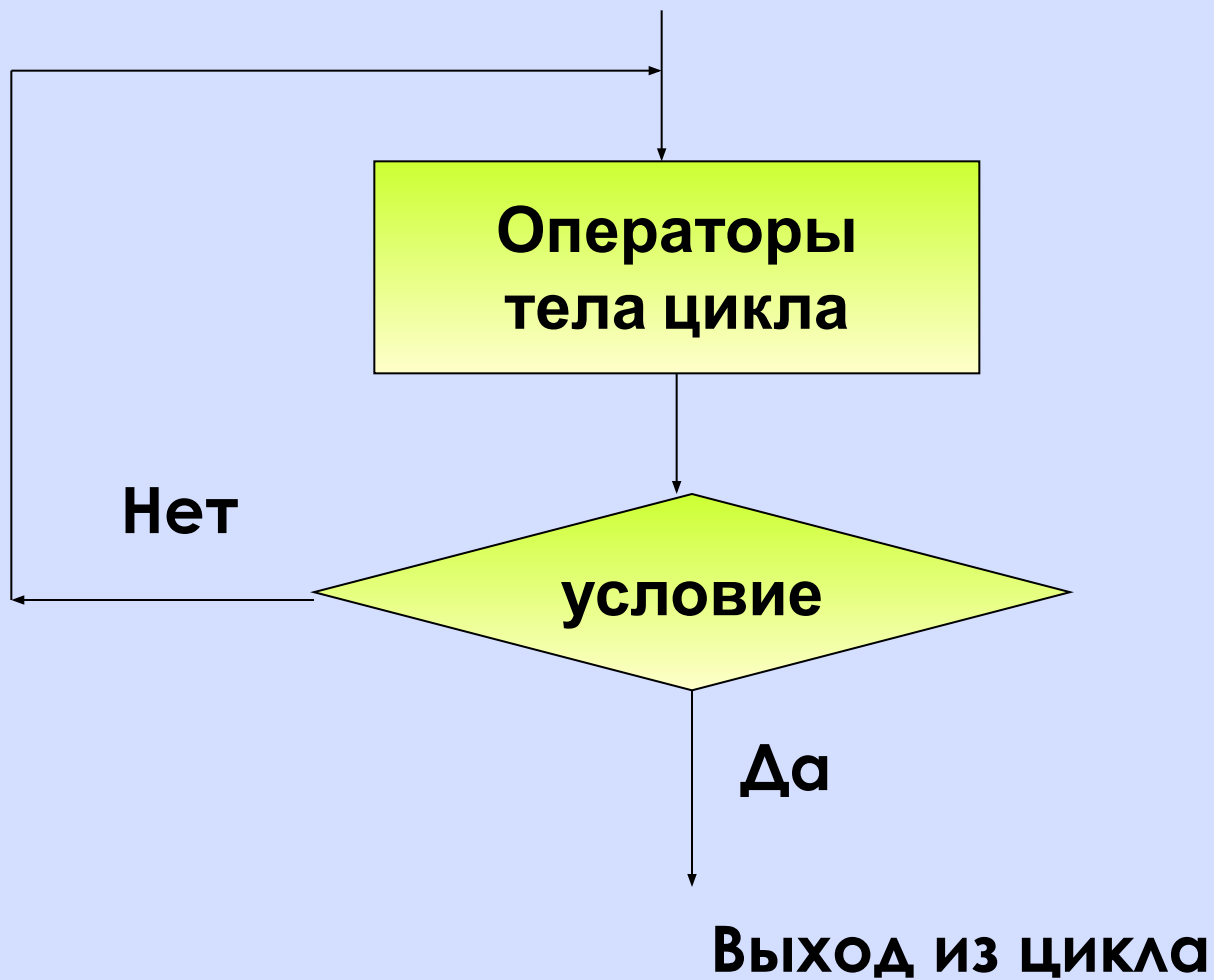
Пример программы (цикл с постусловием):

Вычислить сумму $S=1+2+3+\dots+N$:

```
Program zadacha;  
var n,s, i : integer;  
Begin  
  write('введите число N '); readln(n);  
  s:=0; i:=1;  
  Repeat  
    s:=s+i;  i:=i+1;  
  Until i>n;  
  writeln('сумма чисел равна', s);  
End.
```



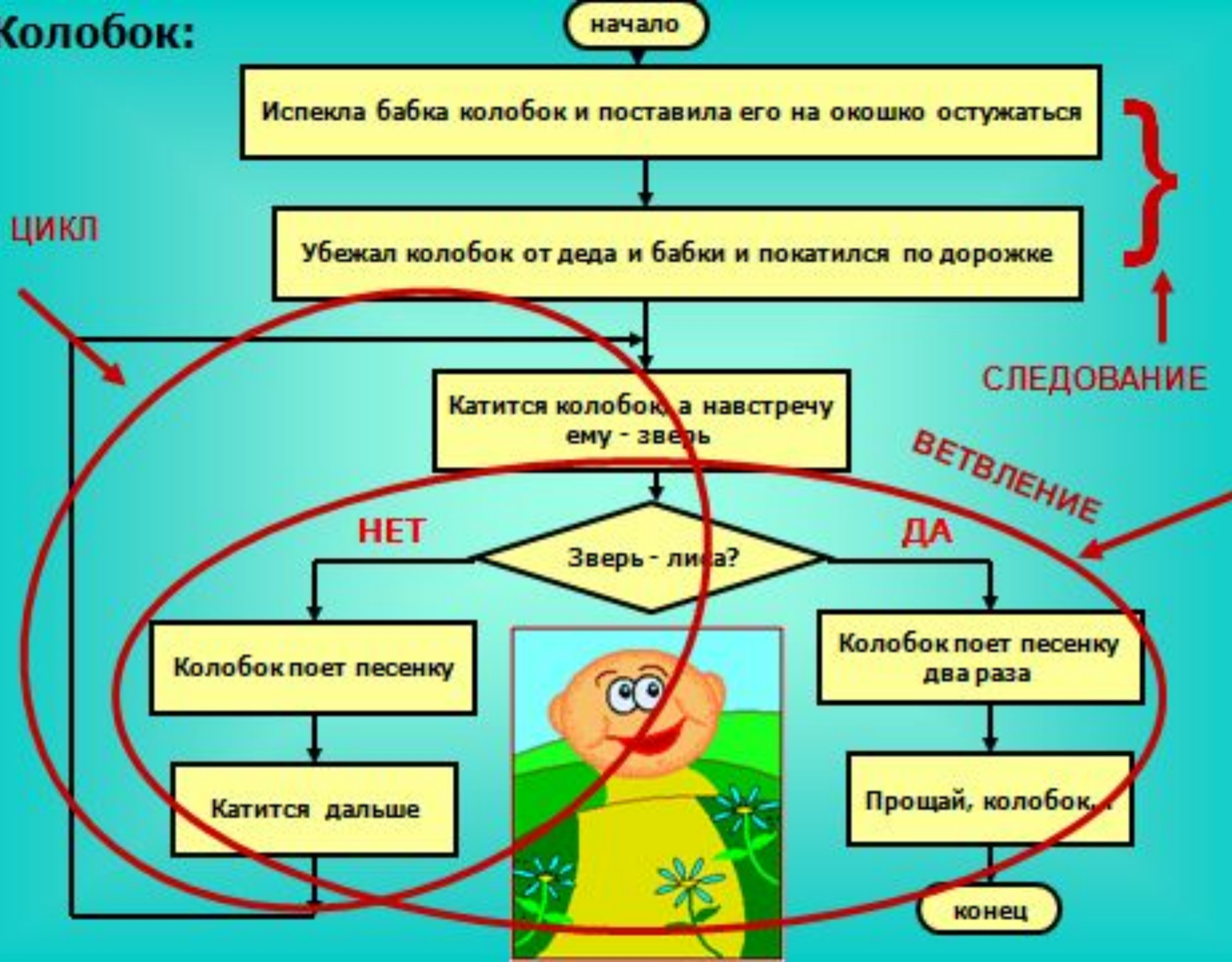
Блок-схема цикла с постусловием:



Практическое задание:

Найти сумму всех четных чисел в диапазоне от 1 до 40.

Колобок:



Домашнее задание:

Вывести на экран числа:

```
10  10,4  
11  11,4  
.....  
25  25,4
```