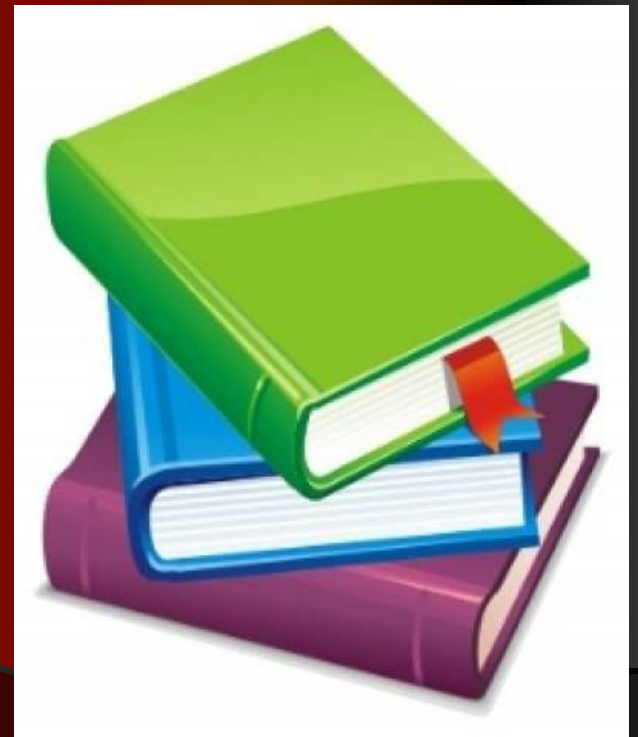


Вопросы

1. Что такое линейный алгоритм?
2. Как реализуется линейный алгоритм в программе?
3. Какая форма организации действий называется ветвлением?
4. Как оформляют алгоритм ветвления ?

Цель урока:

- знакомство с понятием цикла и развитие умения решать различные задачи по данной теме.



Скажите где в жизни вы
встречаетесь с понятием цикл?



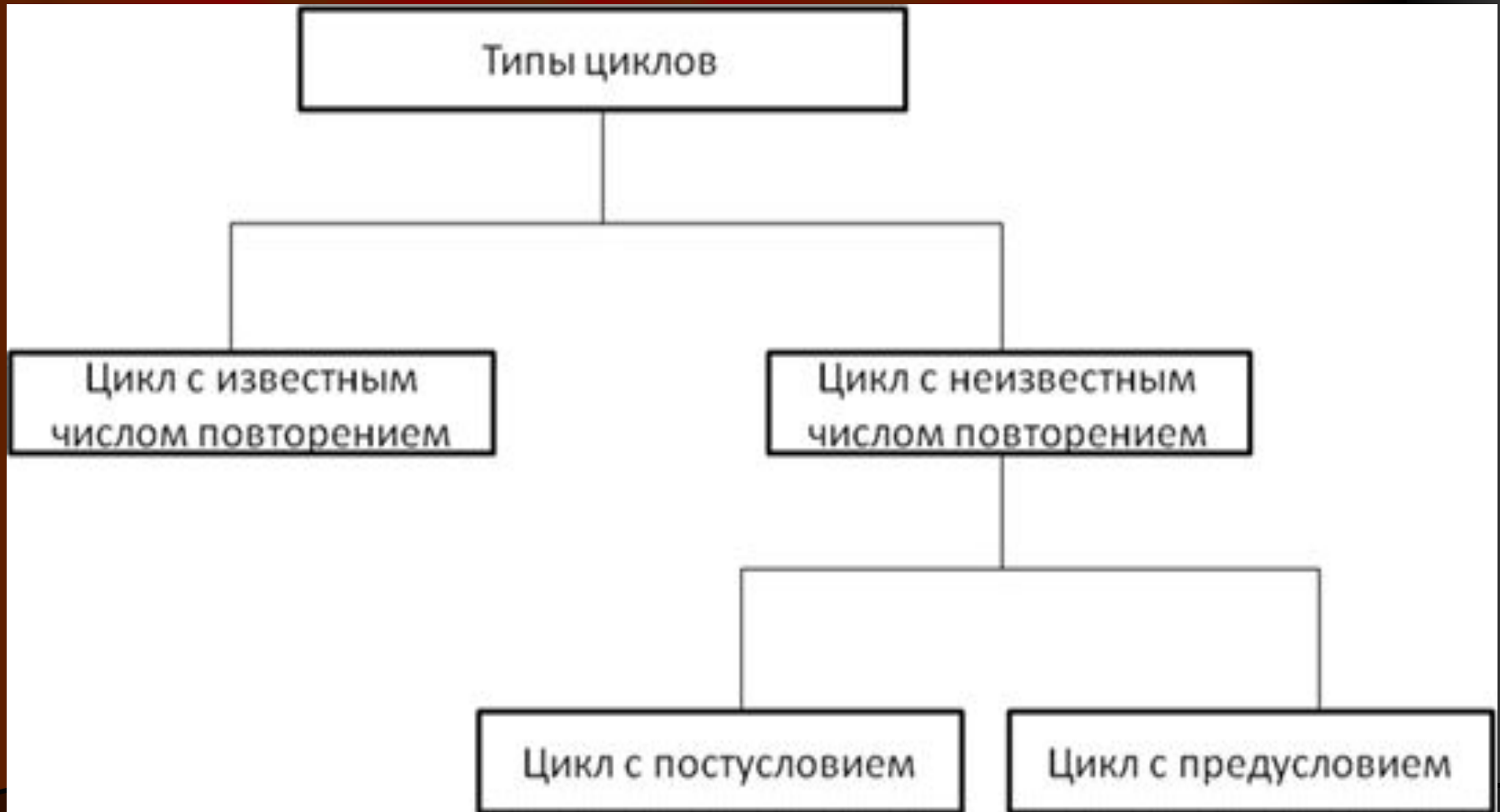
Что такое цикл?

Цикл — форма организации действий, при которой одна и та же последовательность действий совершается несколько раз до тех пор, пока выполняется какое — либо условие.

Для организации цикла необходимо выполнить следующие действия:

- 1. перед началом цикла задать начальное значение параметра;
- 2. внутри цикла изменять параметр цикла с помощью оператора присваивания;
- 3. проверять условие повторения или окончания цикла;
- 4. управлять циклом, т.е. переходить к его началу, если он не закончен, или выходить из цикла в противном случае.

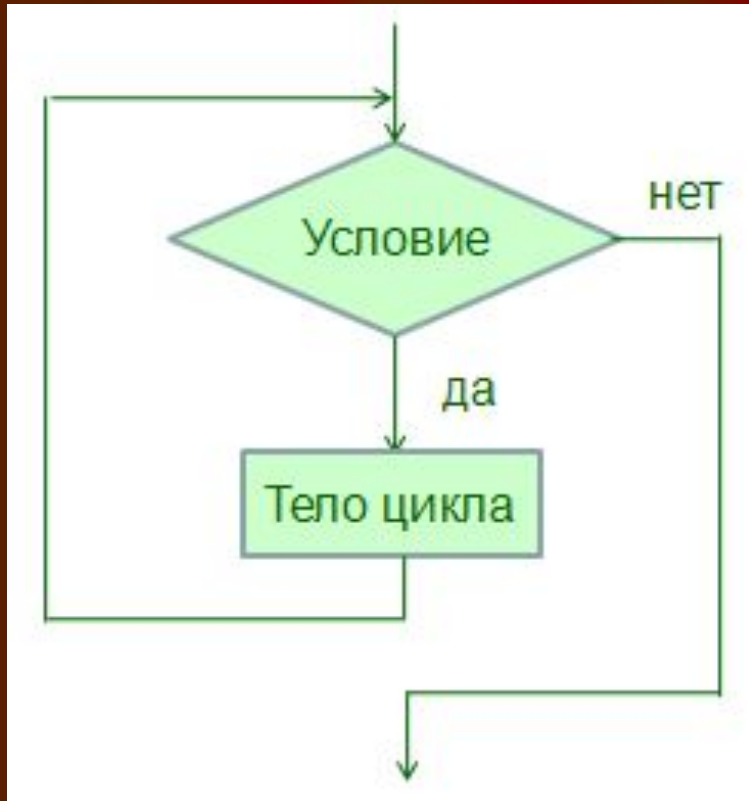
Циклы делятся на:



Язык Паскаль имеет три оператора, с помощью которых можно организовать циклическую структуру:

- Цикл с предусловием "Пока"
(*While ... do*)
- Цикл с постусловием "До"
(*Repeat ... until*)
- Цикл с параметром (счетчиком)
"Для" (*For ...*)

Циклом с предусловием «Пока»



Пока

<условие>

нц

<тело

цикла>

кц

Формат оператора цикла с предусловием «Пока»:

- <присвоение начальных значений переменным, входящим в условие>

На языке Turbo Pascal записывается:

While <условие> **do**

begin

<действие 1>

<действие 2>

.....

<действие N>

<изменение условия>

end;

При построении программ со структурой цикла WHILE:

- 1) начальное значение логического выражения должно быть определено перед входением в цикл, т.е. перед WHILE;
- 2) последовательность выполняемых операторов в теле цикла должна быть логически связанной и содержать хотя бы один оператор, изменяющий значение параметра цикла, иначе цикл будет продолжаться бесконечно;
- 3) логическое выражение в заголовке цикла WHILE, чтобы цикл закончился, должно принять значение FALSE.

Работа оператора while

- 1.Проверяется условие, стоящее после while;
- 2.Если условие истинно, то выполняется оператор(тело цикла), стоящий после do;
- 3.Переход к пункту 1;
- 4.Если условие ложно, то происходит ВЫХОД ИЗ ЦИКЛА.

Написать алгоритм, выводящий таблицу умножения на 3.

Решение:

алг Таблица умножения на 3

арг К

нач

К:=1

пока К<=10

нц

ВЫВОД 3*К

К:=К+1

кц

КОН

Пока

<условие>

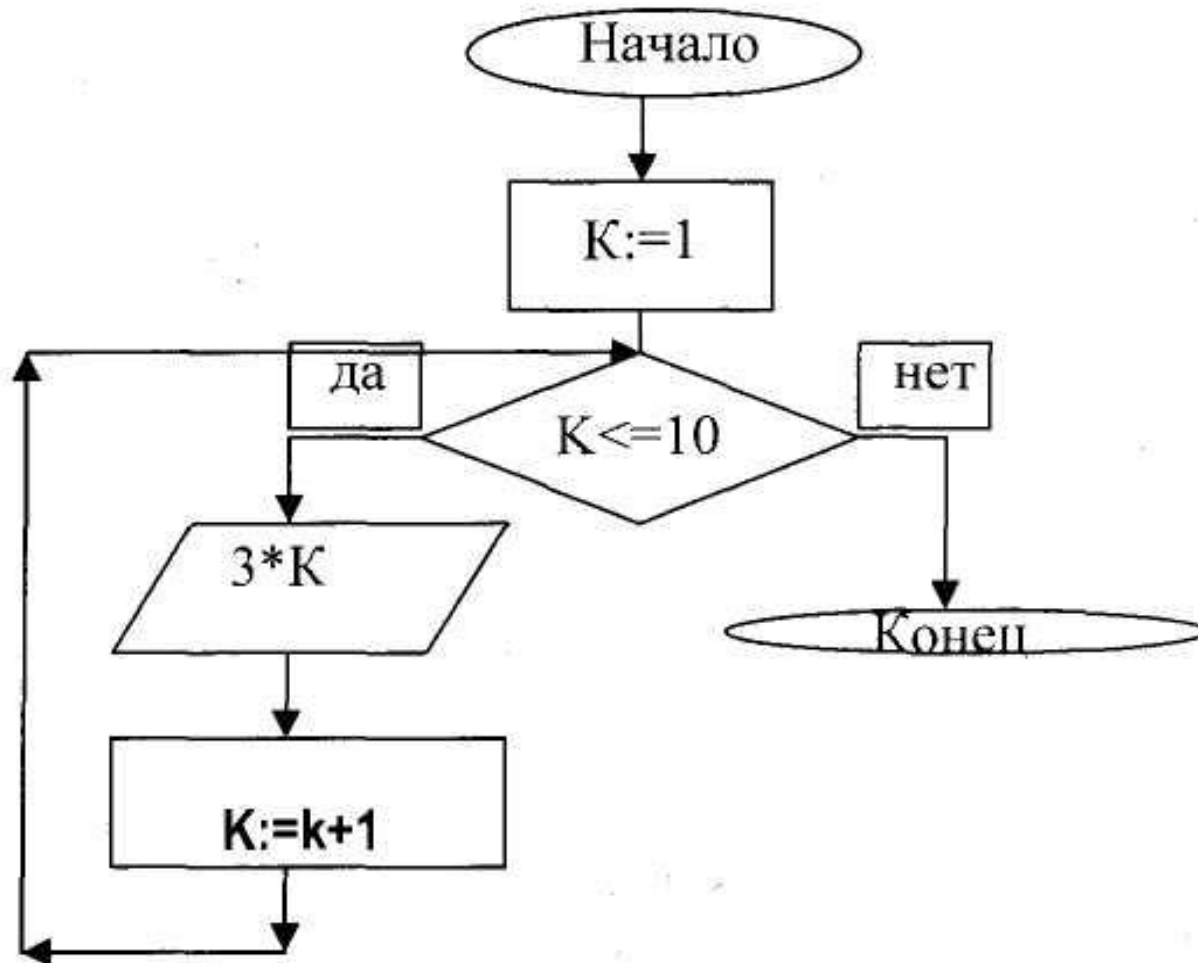
нц

<тело

цикла>

кц

Решение в виде блок-схемы



Особенности цикла `While...do`



Так как условие проверяется на входе в цикл, то **при неверном условии цикл не выполняется ни разу**, т.е. не выполняются операторы, стоящие в теле цикла.



Операторы, входящие в тело цикла, обязательно заключаются в **операторные скобки**, если в теле цикла более одного оператора. В противном случае будет выполняться только первое действие, стоящее под заголовком цикла.



В теле цикла должно обязательно выполняться **действие, приводящее к изменению условия**, иначе цикл станет бесконечным. Оператор, в котором изменяются переменные, входящие в условие, может стоять не обязательно в конце цикла.

Цикл предусловием **While ... do**

Решение задачи о выводе 10 целых чисел на экран с использованием цикла *While... do*:

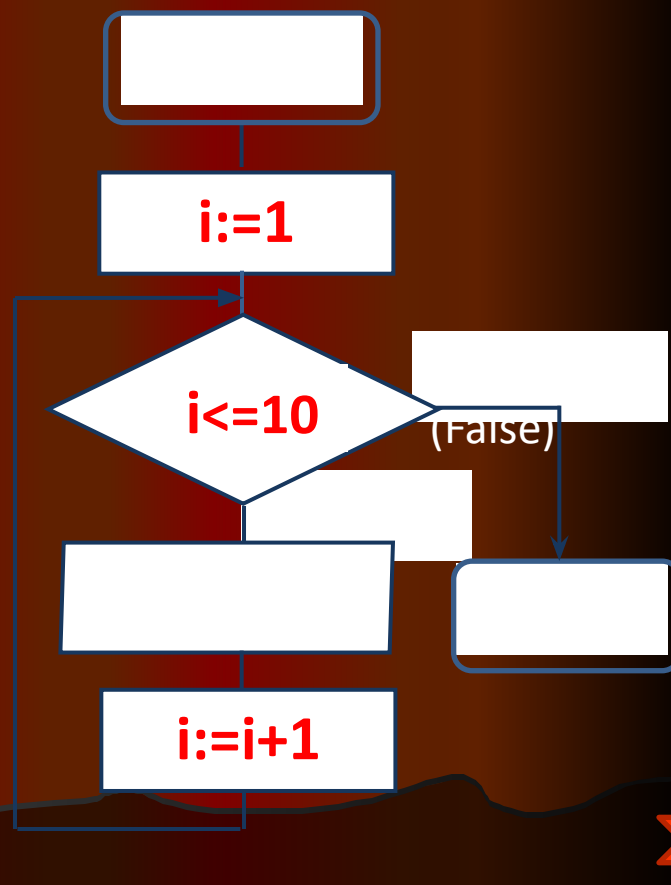


Программа

```
var i: integer; {сЧЕТЧИК}
Begin
  i := 1; {начальное значение}
  While i <= 10 do
    begin
      Writeln(i);
      i := i + 1
    end; {While}
End.
```



Блок-схема алгоритма



Задача

- Вычислить сумму натурального ряда чисел от 1 до N .

Решение

- Program summa;
- Var i,s,n: integer;
- Begin
- writeln('Введите n');
- readln(n);
- i:=1; {Первое слагаемое}
- s:=0; {Значение суммы}
- while i<=n do
- Begin
- s:=s+i; {Сумма увеличивается}
- i:=i+1; {Слагаемое увеличивается на 1}
- End;
- writeln('Сумма= ',s);
- End.

Задача

- Когда Василисе Премудрой исполнилось 18 лет, Кощей Бессмертный решил взять ее замуж. Василиса спросила, сколько у Кощея сундуков с золотом. Кощеи сказал, что у него 27360 сундуков полных, и каждый год прибавляется еще по 33 сундука. Василиса обещала выйти за Кощея тогда, когда у него будет 30000 сундуков, полных золота. Сколько лет будет невесте Кощея в день свадьбы?

Решение

- Program KOSHEY_vasilisa;
- Var v,s:integer;
- Begin
- v:=18; {Возраст Василисы}
- s:=27360; {Начальный капитал Кощея}
- while s<30000 do
- Begin
- s:=s+33; {Каждый год Кощей становится богаче}
- v:=v+1; {Каждый год Василиса становится старше}
- end;
- writeln('Василисе ',v,' лет');
- End.

Цикл предусловием **While ... do**

- Начав тренировки, лыжник в первый день пробежал 10 км. Каждый следующий день он увеличивал длину пробега на 10% от предыдущего дня. Определить в какой день он пробежит больше 20 км, в какой день суммарный пробег за все дни превысит 100км.

Решение

- Program prim;
Var S:real;
 N:integer;
Begin
 S:=10;N:=1;
 While s<20 do
 Begin
 N:=n+1;
 S:=s*0.1
 End;
 Writeln(' дневной пробег больше 20 км на ',n,' день');
 S:=10;N:=1;
 While s<100 do
 Begin
 N:=n+1;
 S:=s+s*0.1 {накапливаем суммарный пробег}
 End;
 Writeln('за ',n,' пробежит больше 100 км');
 Readln
End.

Цикл с постусловием «До»

Выполнение тела цикла повторяется до тех пор, пока условие ложно.

Тело цикла выполнится хотя бы один раз.



Цикл с постусловием «До»

- **Форматы записи операторов на Паскале**

repeat <серия команд>

until <условие>

*Последовательность инструкций между **repeat** и **until** всегда будет выполнено хотя бы один раз .*

- Repeat имеет формат:

```
repeat { повторяй ... }  
    <оператор 1>;  
    < оператор 2>;
```

```
    until {до...} <условие>
```

Begin и **end** не требуются.

Как произносится конструкция

- Repeat(ри'пит) – повторяй;
- Until(ан'тил) – до тех пор пока;
- Конструкция:
- Repeat $y := a + 1$ until $a + 2 > 10$
- Читается так:
- **Повторяй** y присваивать $a + 1$ **до тех пор, пока** $a + 2$ **не станет больше 10.**

Работа оператора repeat

- 1.Выполняется тело цикла;
- 2.Проверяется условие выхода из цикла; (Поэтому цикл выполняется хотя бы один раз)
- 3.Если условие ложно, то переход к пункту 1;
- 4.Если условие истинное, то происходит выход из цикла.

Особенности цикла **Repeat...until**

- ❗ Так как условие проверяется на выходе из цикла, то *цикл выполняется хотя бы один раз.*
- ❗ Все операторы, стоящие в теле цикла, выполняются ДО проверки условия, поэтому *операторные скобки не ставятся.*
- ❗ В *теле цикла* должно обязательно выполняться *действие, приводящее к изменению условия*, иначе цикл станет бесконечным. Оператор, в котором изменяются входящие в условие переменные, может стоять не обязательно в конце цикла.



Цикл с постусловием Repeat ... until

Решение задачи о выводе 10 целых чисел на экран с использованием цикла

Repeat...until:

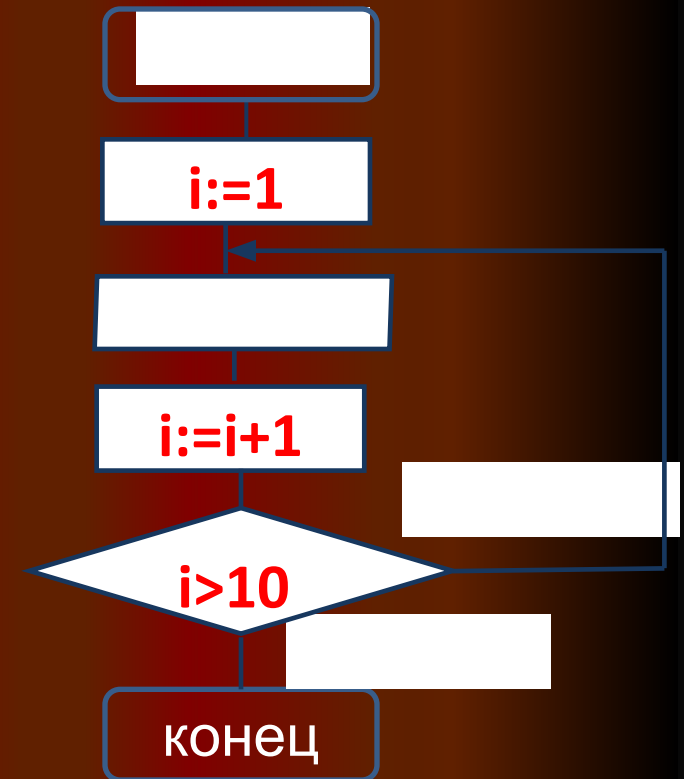


Программа

```
var i: integer; {счетчик}  
Begin  
  i := 1; {начальное значение}  
  
  Repeat  
    Writeln(i);  
    i := i + 1;  
  Until i > 10  
End.
```



Блок-схема алгоритма



Сравнение циклов **While** и **Repeat**

Формат оператора цикла с предусловием:
<присвоение начальных значений переменным, входящим в условие>

While <условие>

do
Тело цикла
begin
<действие 1>
<действие 2>
.....
<действие N>
<изменение условия>

end:

1. **До** цикла задается начальное значение переменных, входящих в условие. **Условие** - выражение **булевского** типа.
2. **В теле цикла** значение переменных, входящих в условие, должны **обязательно изменять свое значение**, иначе возникнет ситуация «зависания».



Формат оператора цикла с постусловием:
<присвоение начальных значений переменным, входящим в условие >

Repeat

Тело цикла
<действие 1>
<действие 2>
.....
<действие N>
<изменение условия>
until <условие> ;

Вопросы

1. Что такое циклический алгоритм?
2. Что такое цикл?
3. Какую конструкцию имеет цикл с предусловием и цикл с постусловием?
4. В чем отличие цикла с постусловием от цикла с предусловием?

Геометрическая прогрессия

Геометрической последовательностью называется последовательность, в которой отношение между ее членом и членом, ему предшествующим, есть величина постоянная.



В символьной записи это можно записать так:

$$a, a*r, a*r^2, a*r^3, \dots, a*r^{(N-1)}$$

Здесь буквой **a** обозначен первый член последовательности, буквой **r** – ее знаменатель и буквой **N** – число членов последовательности.

Например, если $a=4$, $r=0.5$, $N=7$, получаем последовательность

$$4, 2, 1, 0.5, 0.25, 0.125, 0.625$$



Сумма геометрической прогрессии **S** определяется

формулой: $S = a + a*r + a*r^2 + a*r^3 + \dots + a*r^{(N-1)}$

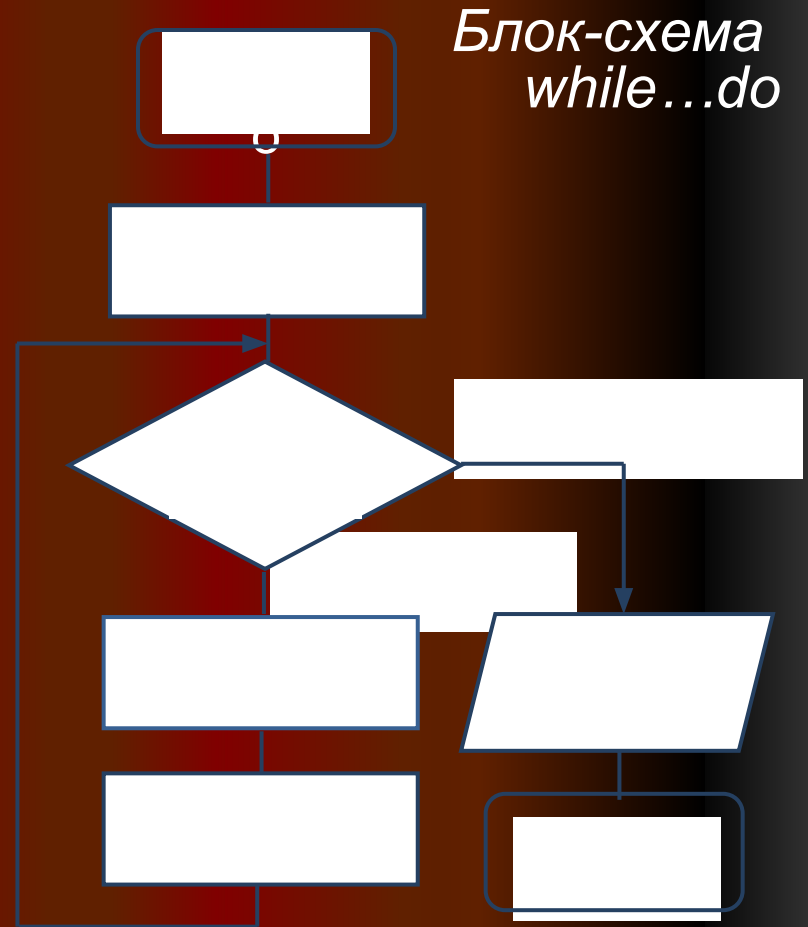
Задача «Изумруды»

Задача

У царя было семь сыновей. В сундуке лежали изумруды. Пришел первый сын и взял половину того, что было. Пришел второй сын и взял половину того, что осталось и т.д. **Каждый из сыновей приходил и забирал половину того, что осталось.** Наконец, пришел последний, седьмой сын и увидел почти пустой сундук — с двумя изумрудами. Сколько изумрудов лежало в сундуке первоначально?

При решении задачи будем использовать цикл **Repeat...until.**

Блок-схема



$r=2$ (разность геометрической прогрессии).

Помощь!!!

- В этой задаче, мы имеем геометрическую прогрессию, для которой известен начальный член $a=1$ и разность $r=2$:
- Для 7-го сына имеем: $i=7$ $q=a*r = 1*2 = 2$ изумруда
- Для 6-го сына имеем: $i=6$ $q=a*r*r = 1*2*2$
изумруда
- Для 5-го сына имеем: $i=5$ $q=a*r*r*r = 1*2*2*2$
изумруда
-
- Надо продолжать вычисление до $i=1$, т.е. для 1-го сына. Определив q , узнаем, сколько изумрудов лежало в сундуке первоначально.

Программа

- **Program** izumrud;
- **Var** i, q, r: integer;
- **Begin**
- i:=7;
- q:=2;
- r:=2;
- WriteLn('i=', i, 'q=',q) ;
- **Repeat**
- q:=q*r;
- i:=i-1;
- WriteLn('i=', i, 'q=', q);
- **Until**
- i<=1;
- WriteLn('всего', q, 'изумрудов');
- **End.**

Самостоятельно!

- Какое значение будет принимать переменная y после выполнения программы:

```
Var x, y:integer;  
Begin  
  y:=1;x:=15;  
  repeat  
    y:=y*x;  
    x:=x-3  
  until x<5;  
  Writeln(' y=',y);  
End.
```

x	y	Y=y*x
15	1	15

- Заполнить таблицу до конца.