

# ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Сидорович Александра  
Сергеевна

[a.s.sidorovich@gmail.com](mailto:a.s.sidorovich@gmail.com)

501a – 5

- Лекций 17
- Лабораторных работ 80
- Контрольных работ на лекциях 20
- Контроль знаний экзамен

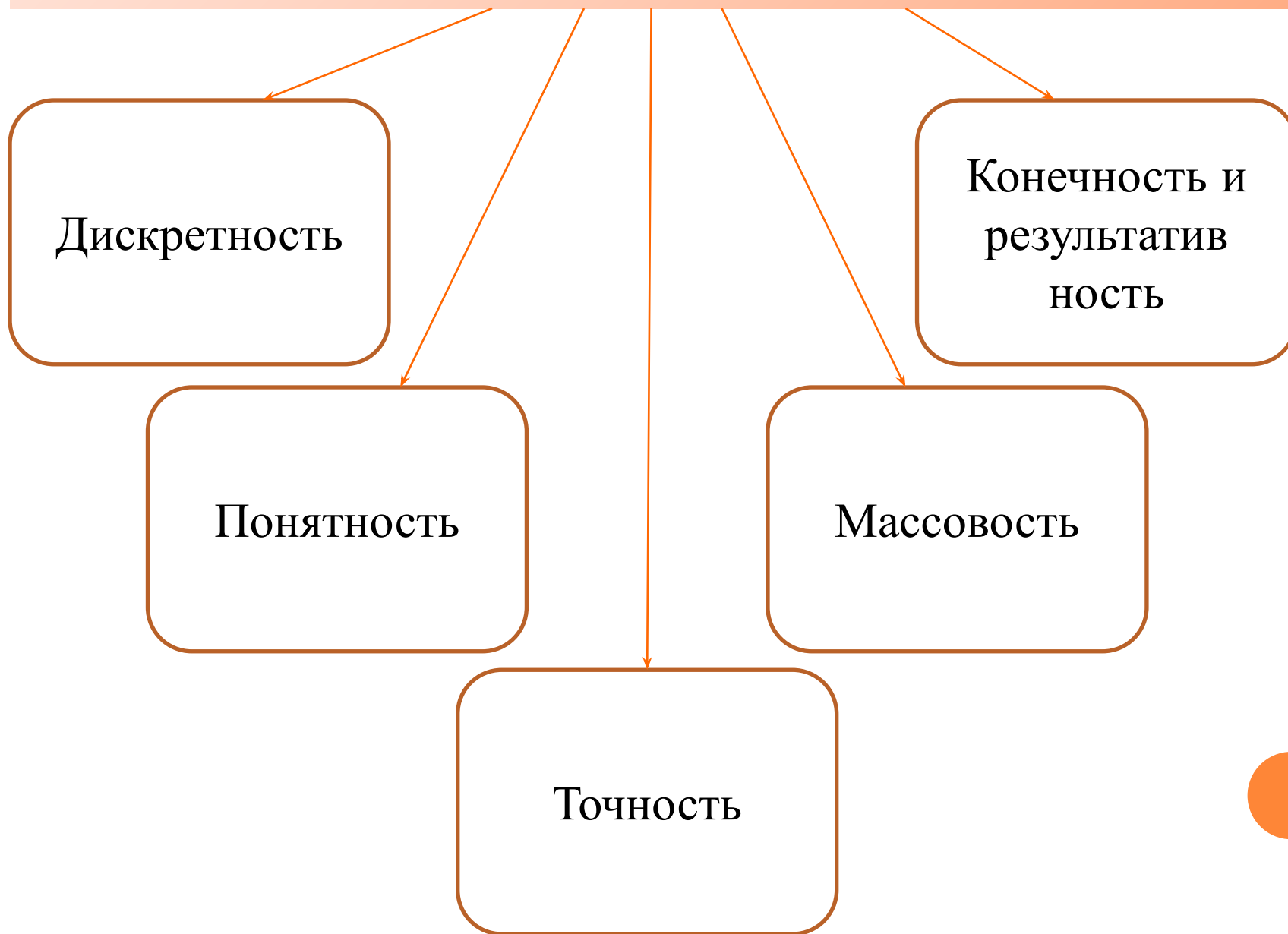


# Понятие и свойства алгоритма

**Алгоритм** – это точное предписание на выполнение последовательности действий, приводящих от исходных данных к результату.



# Свойства алгоритма:



# Дискретность

Алгоритм состоит из отдельных законченных шагов – операций, которые и составляют всю последовательность.

На первом шаге задаются исходные данные.

Каждый последующий шаг – выполнение какого-либо действия.

Последний шаг выдаёт результат алгоритма.



# Понятность

Каждый шаг алгоритма должен быть однозначно понятен и выполним для конкретного исполнителя. Не зависит от выбора языка



# Результативность и конечность

Количество шагов алгоритма должно быть конечным и их последовательность должна приводить к запланированному результату.



# Определенность

На каждом шаге алгоритма должны быть однозначно и точно определены действия.





# Массовость

Возможность многократного применения алгоритма (без изменения структуры) с различными исходными данными.

Необходимо продумать все варианты использования данных для избежания ошибок



# Формы представления

```
graph TD; A[Формы представления] --> B[Словесная]; A --> C[Программная]; A --> D[Графическая];
```

Словесная

Программная

Графическая



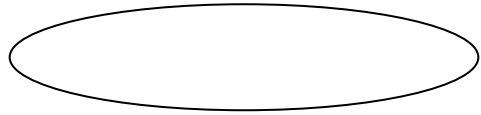
# Пример

## Алгоритм приготовления чая

- Подготовить исходные величины – чай, воду, чайник, ложку.
- Налить в чайник воду.
- Поставить чайник на огонь.
- Довести до кипения и снять с огня.
- Всыпать в чайник 2 ложки чая.
- Поставить чайник на огонь.
- Довести воду до кипения (но не кипятить), снять с огня.
- Процесс прекратить.



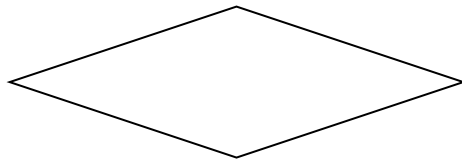
# Блоки для графического представления



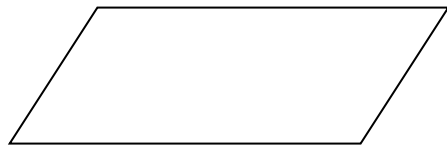
- начало(конец) алгоритма



- действие



- проверка условия



- ВВОД ИЛИ ВЫВОД ДАННЫХ



# Алгоритмические структуры

Линейная

Разветвляющаяся

Циклическая

Полное ветвление

Неполное ветвление

Цикл с предусловием

Цикл с постусловием

Цикл с параметром



# Линейный алгоритм

## СЛОВЕСНОЕ ОПИСАНИЕ

*Алгоритм, в котором команды выполняются последовательно одна за другой, называется линейным алгоритмом. Присутствуют только блоки начало / конец, ввод / вывод, действия*

## БЛОК-СХЕМА

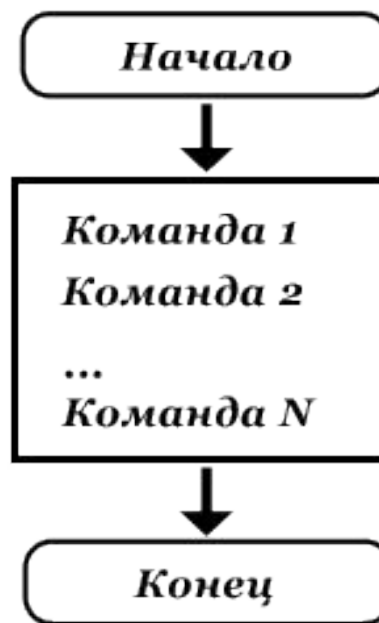


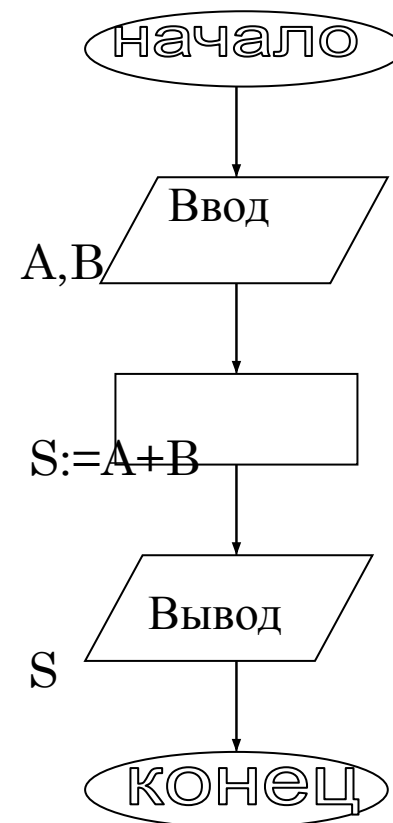
Рис. 1 Линейный алгоритм



# Пример

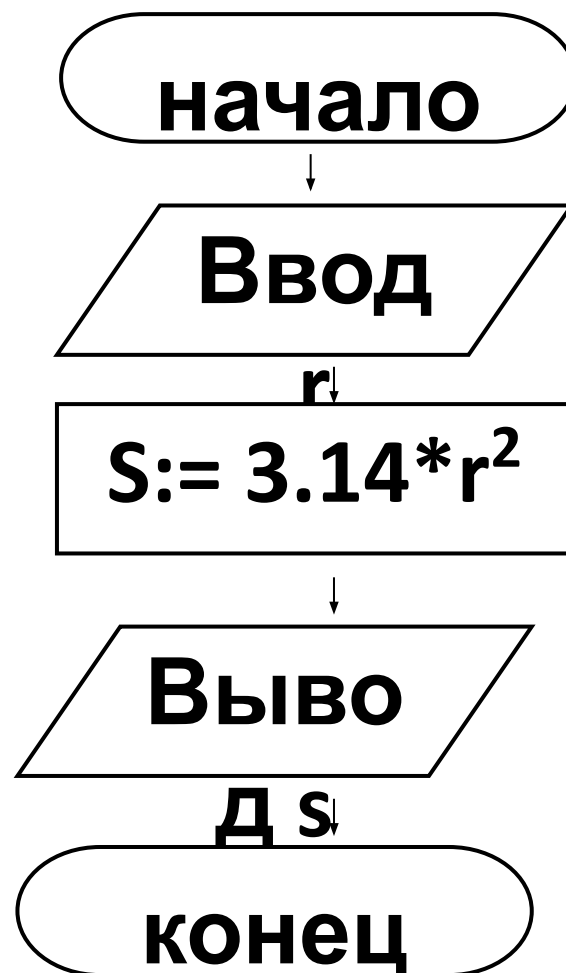
Даны два числа.

Вычислить их  
сумму.



# Пример

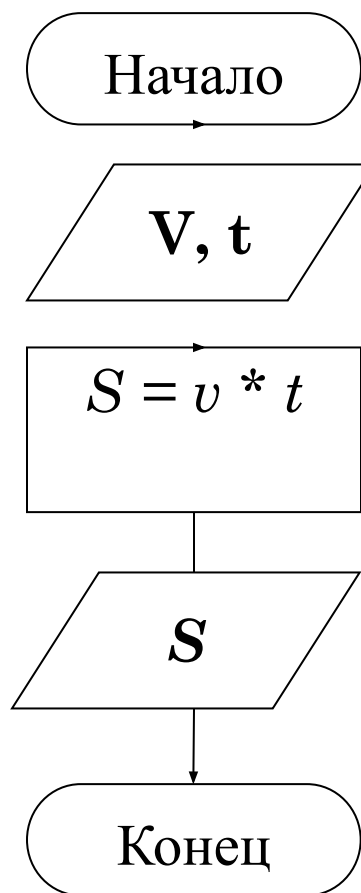
Вычислить  
площадь круга





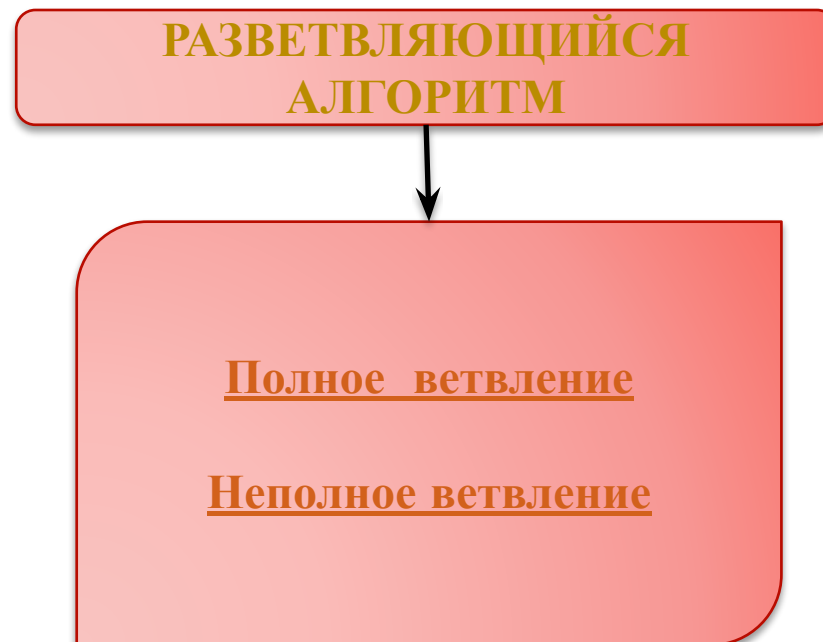
# Пример

Определить расстояние, пройденное человеком, если известно время движения, а так же известно, что движение было равномерным.



# Разветвляющиеся алгоритмы

Алгоритм, в котором появляется условие перехода в то или иное действие называется **разветвляющимся** (условным).

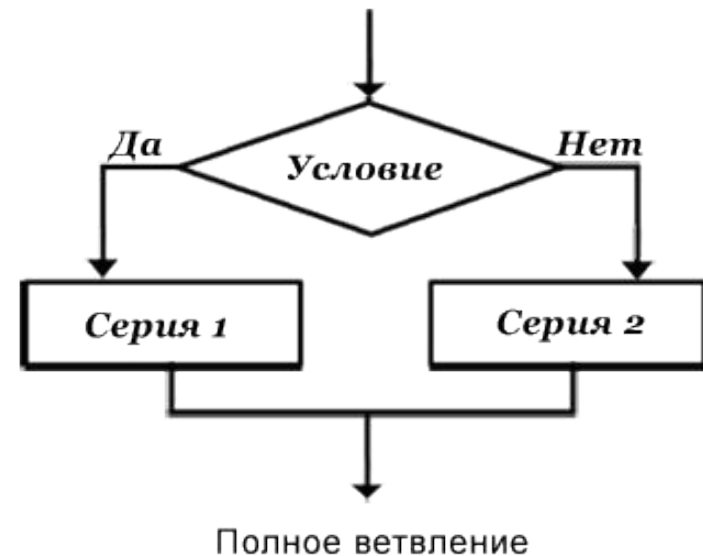


# Полное ветвление

## СЛОВЕСНОЕ ОПИСАНИЕ

В алгоритмической структуре ветвление (полное) в зависимости от истинности или ложности условия выполняется одна или другая серия команд.

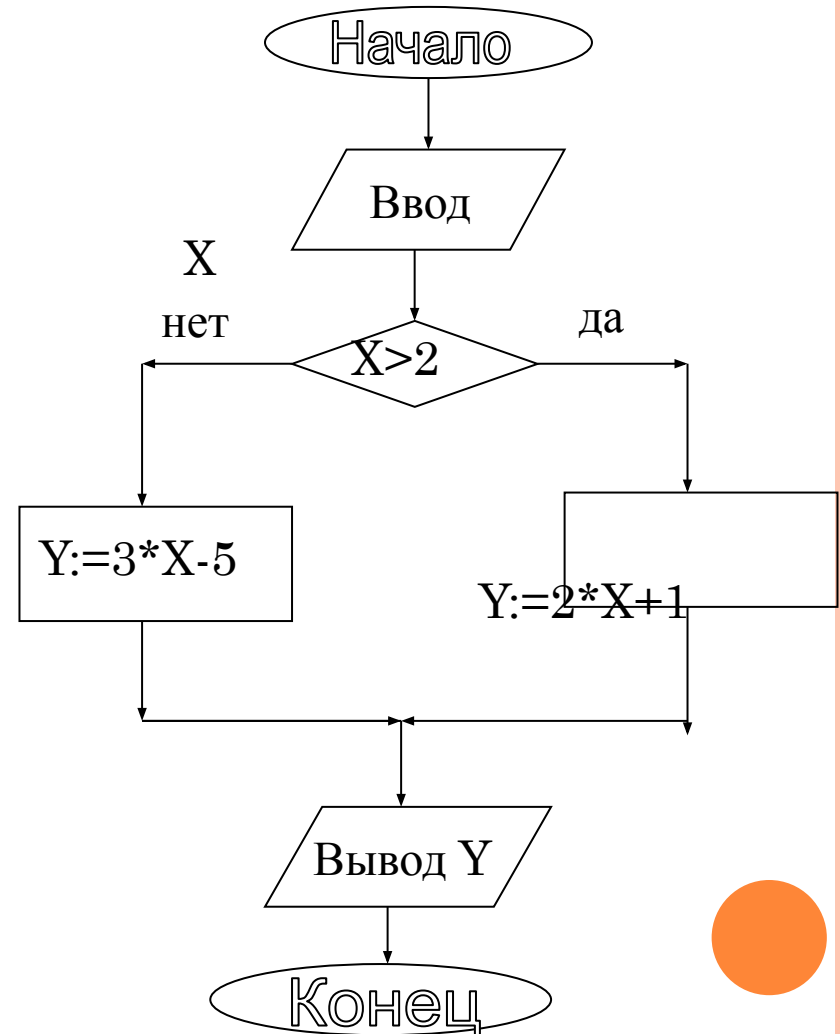
## БЛОК-СХЕМА



# Пример

Вычислить по формулам значения  $Y$ , если известен  $X$ .

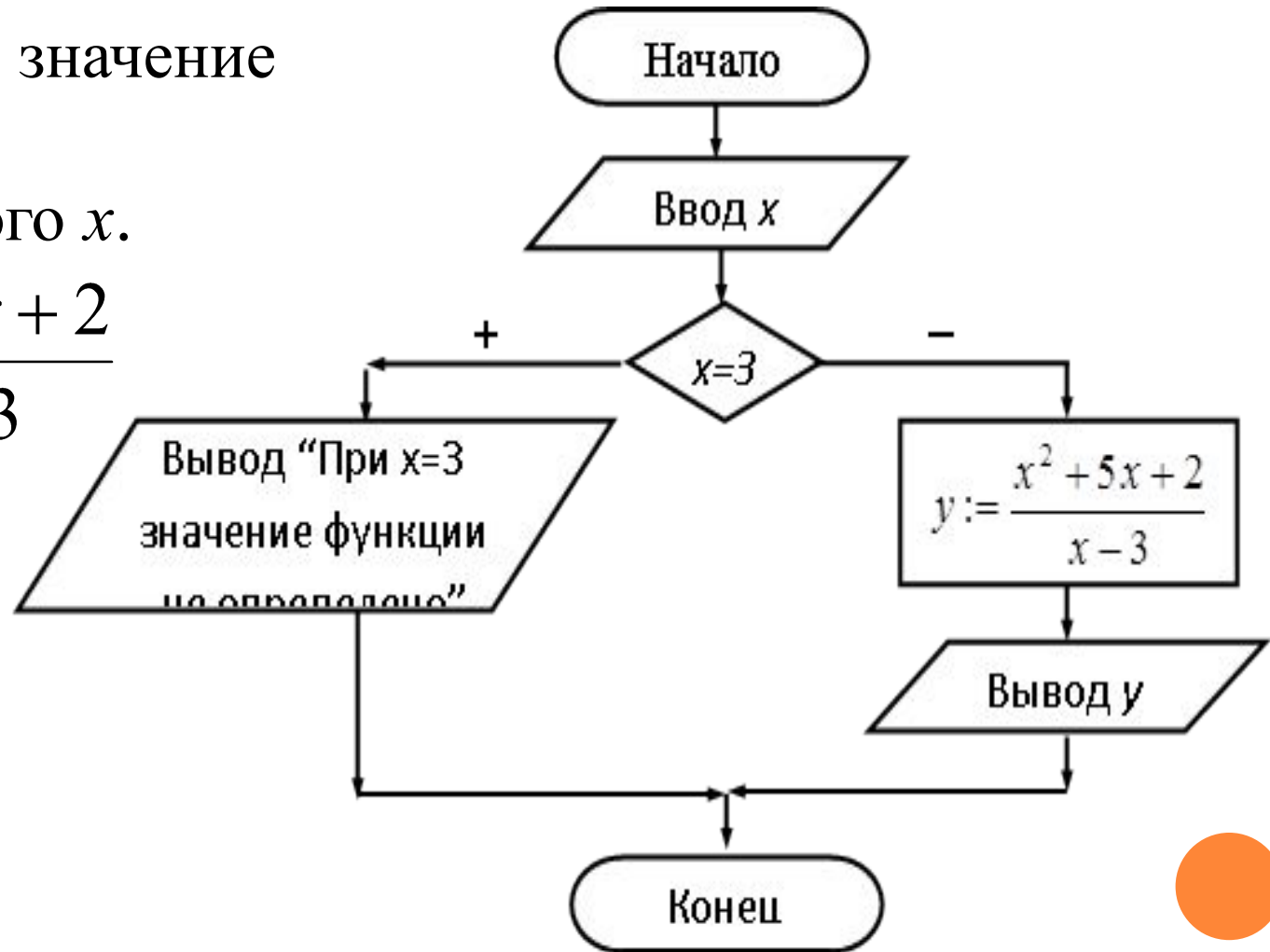
$$Y = \begin{cases} 2X+1, & \text{если } X > 2 \\ 3X-5, & \text{если } X \leq 2 \end{cases}$$



# Пример

Вычислить значение функции для заданного  $x$ .

$$y = \frac{x^2 + 5x + 2}{x - 3}$$

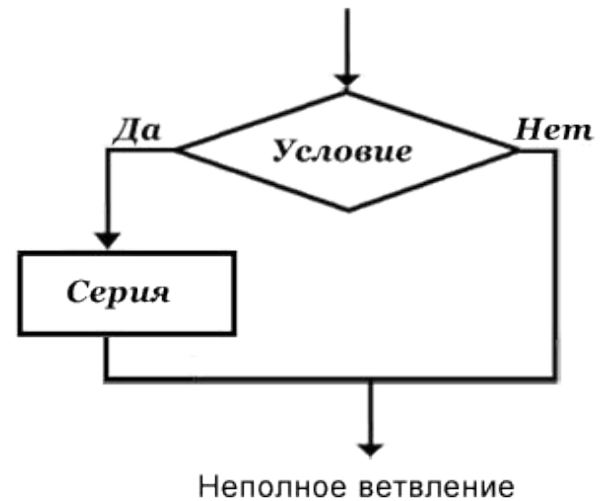


# Неполное ветвление

## СЛОВЕСНОЕ ОПИСАНИЕ

В алгоритмической структуре ветвление (неполное) в зависимости от истинности или ложности условия выполняется только одна команда.

## БЛОК-СХЕМА



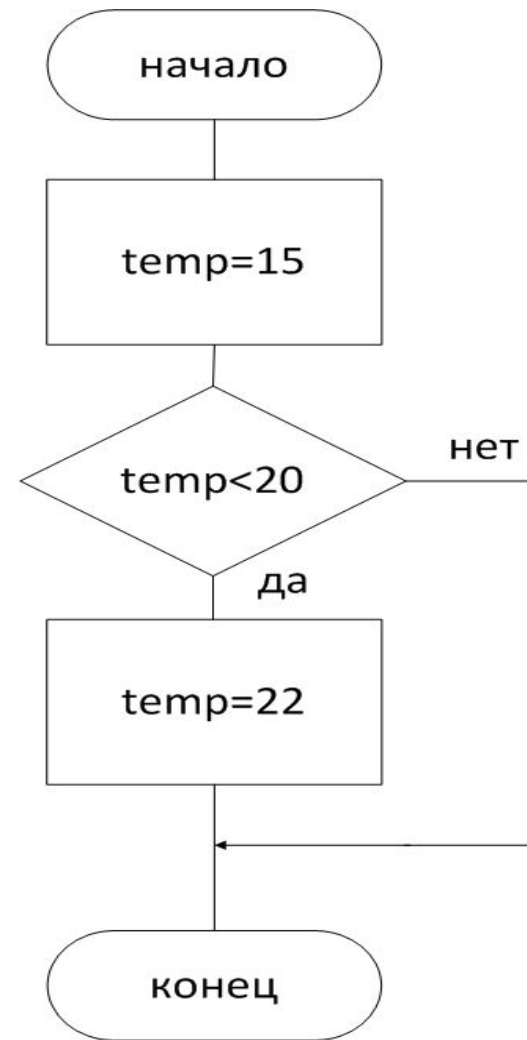
# Пример

Определить  
делимость числа на  
7



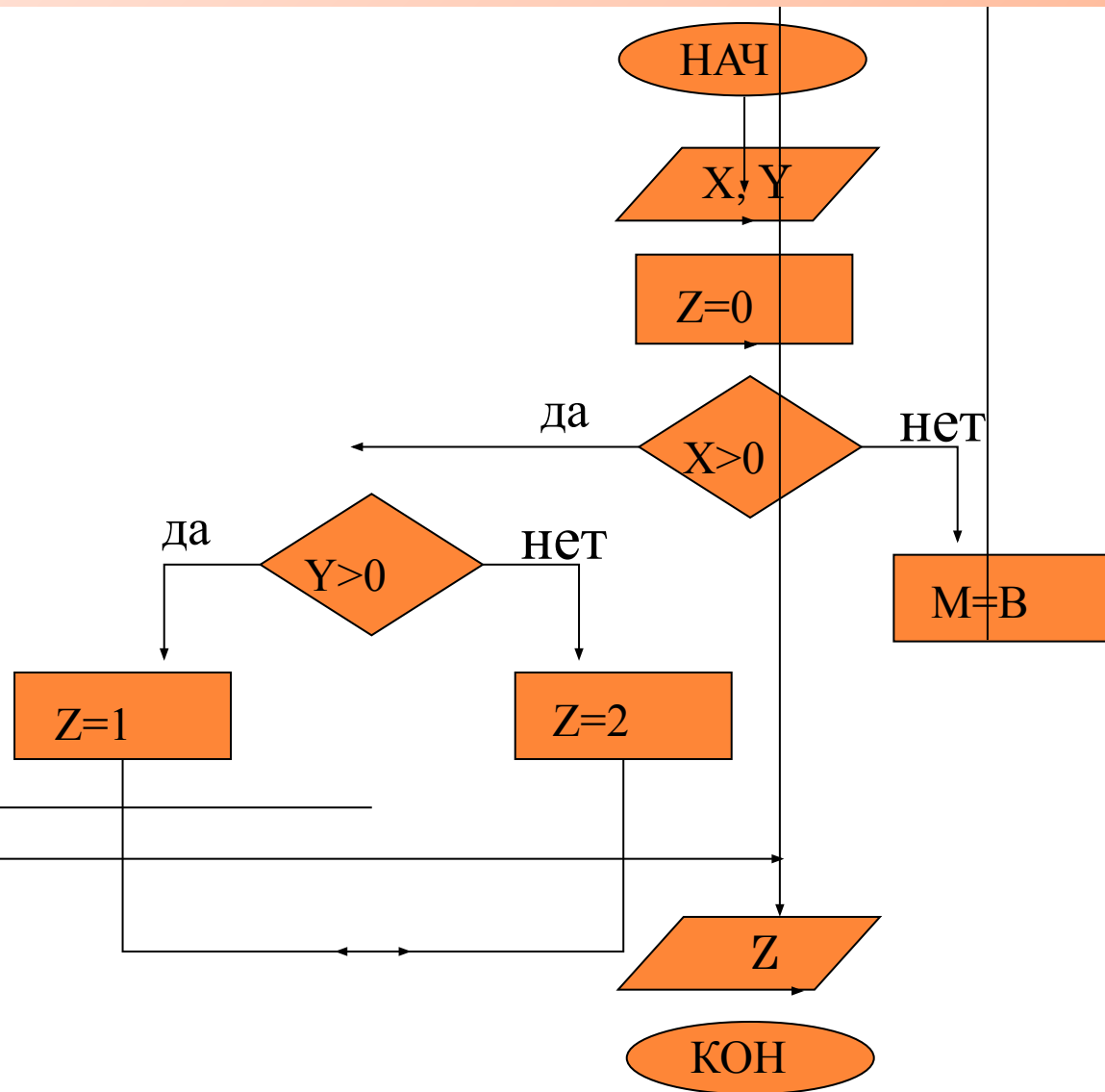
# Пример

Например, необходимо проверить температуру в комнате, и, если она меньше  $20\text{ }^{\circ}\text{C}$ , увеличить температуру до  $22\text{ }^{\circ}\text{C}$ .





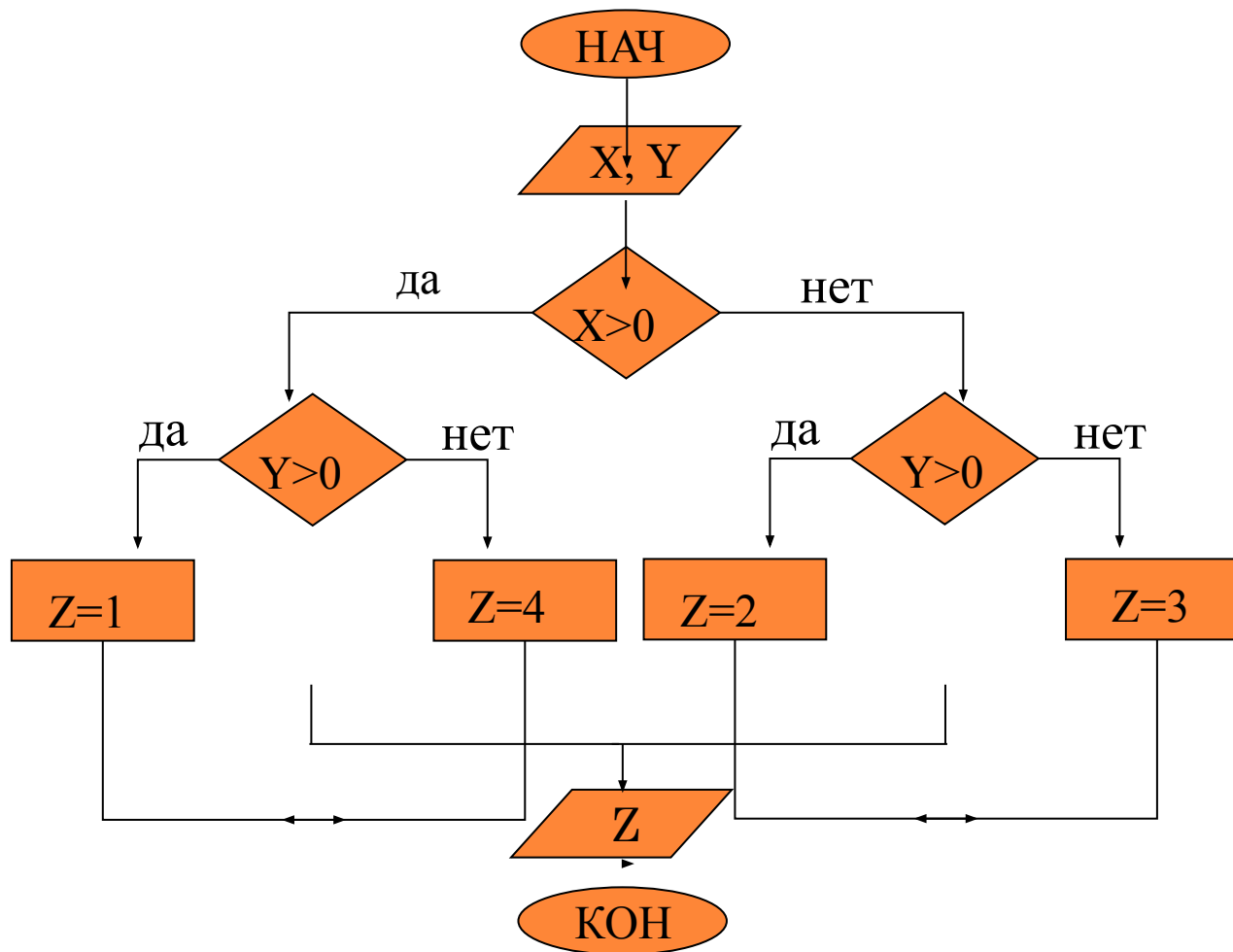
# Пример



Какое значение получит переменная  $Z$  в результате выполнения алгоритма?

1.  $X=1, Y=1$ ;
2.  $X=1, Y=-1$ ;
3.  $X=-1, Y=1$ .

# Пример



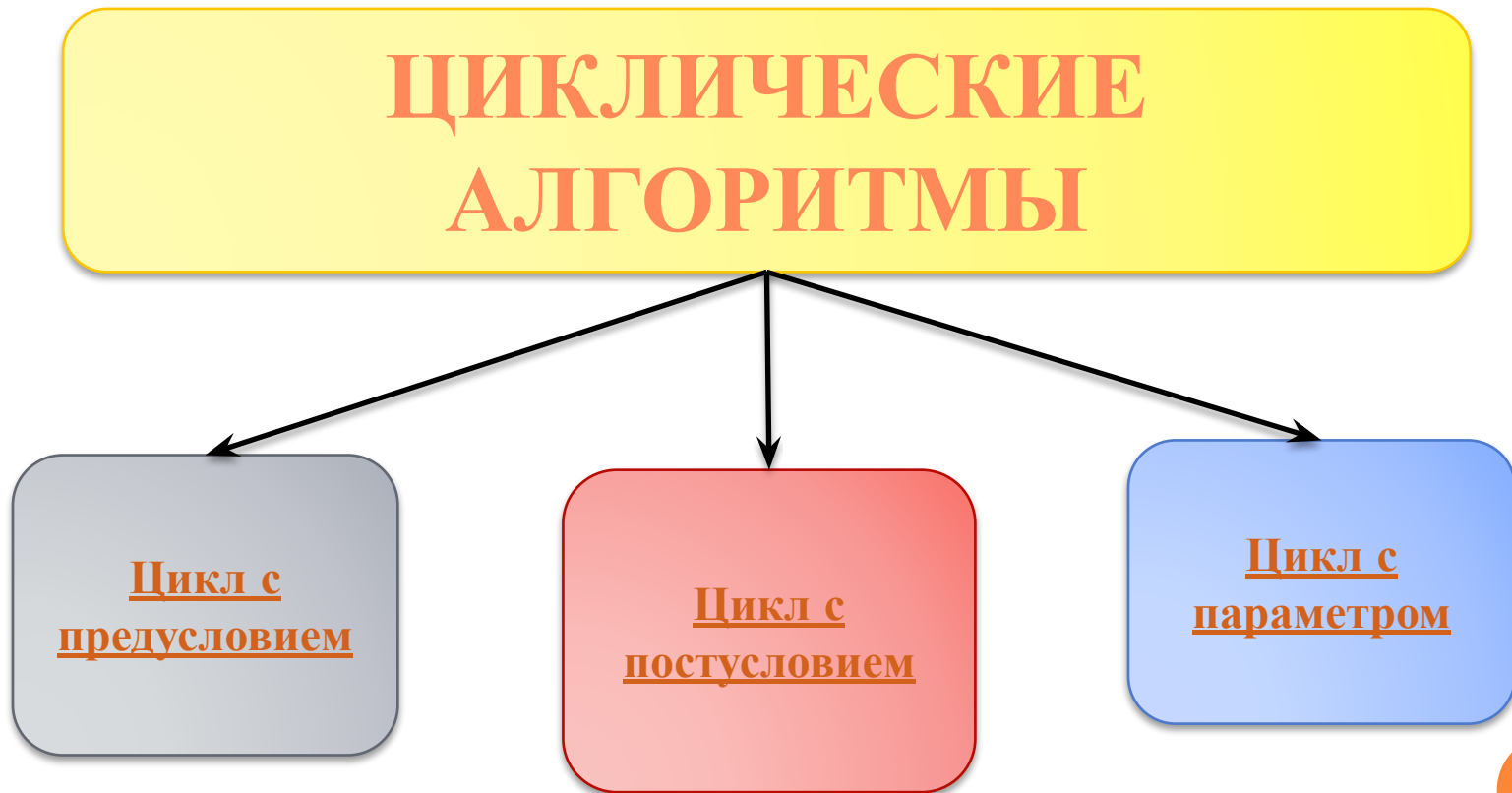
Какое значение получит переменная  $Z$  в результате выполнения алгоритма?

1.  $X=1, Y=1$ ;
2.  $X=1, Y=-1$ ;
3.  $X=-1, Y=1$
4.  $X=-1, Y=-1$



# Циклические алгоритмы

**Циклом** называется блок кода, который для решения задачи требуется повторить несколько раз.

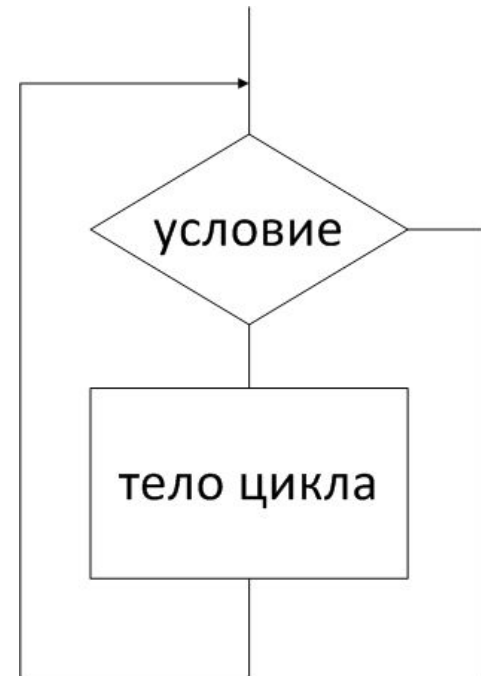


# Цикл с предусловием

## СЛОВЕСНОЕ ОПИСАНИЕ

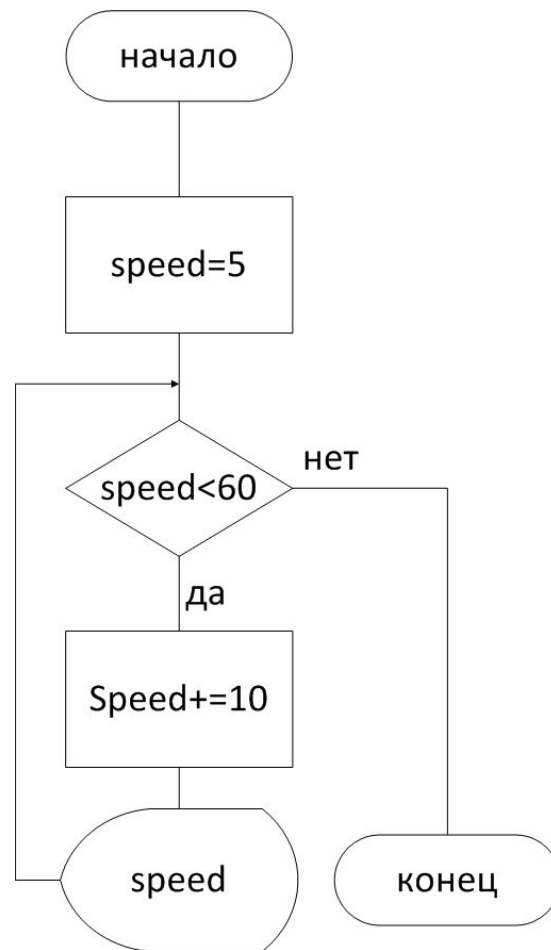
**Цикл с предусловием** — цикл, который выполняется пока истинно условие, указанное перед его началом. Это условие проверяется **до** выполнения тела цикла, поэтому тело может быть не выполнено ни разу (если условие с самого начала ложно).

## БЛОК-СХЕМА



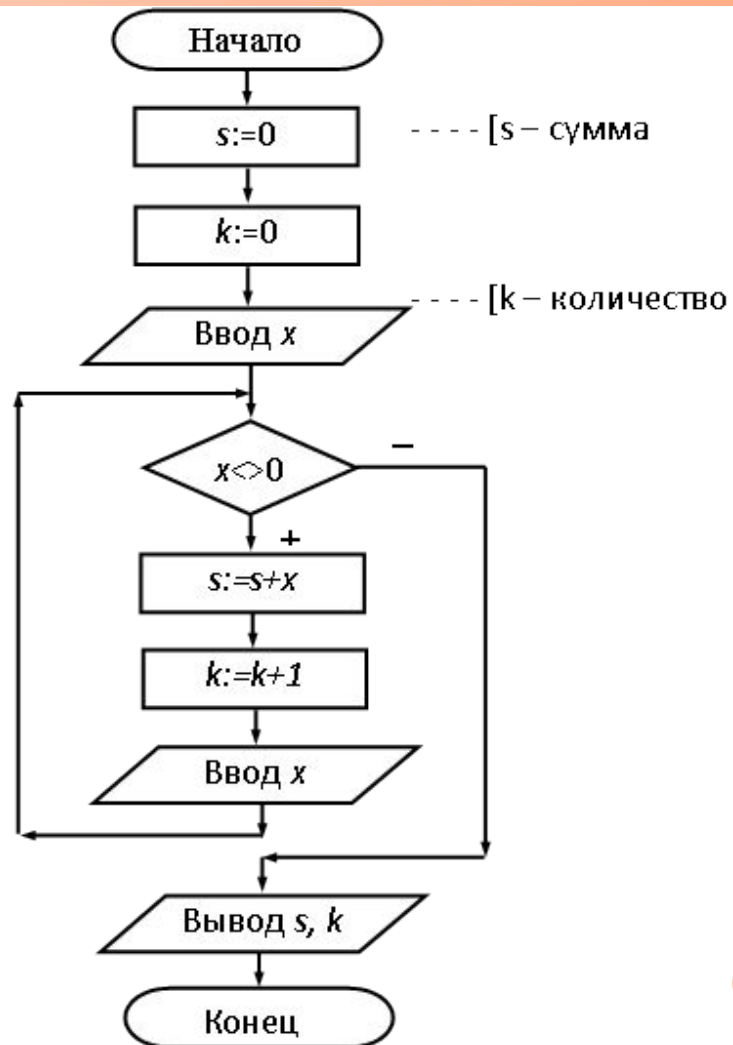
# Пример

Пока скорость движения автомобиля меньше 60 км/ч, продолжать наращивать скорость на 10 км/ч.



# Пример

Вводить числа,  
пока не  
встретится 0.  
Определить  
сумму и  
количество  
введенных чисел.



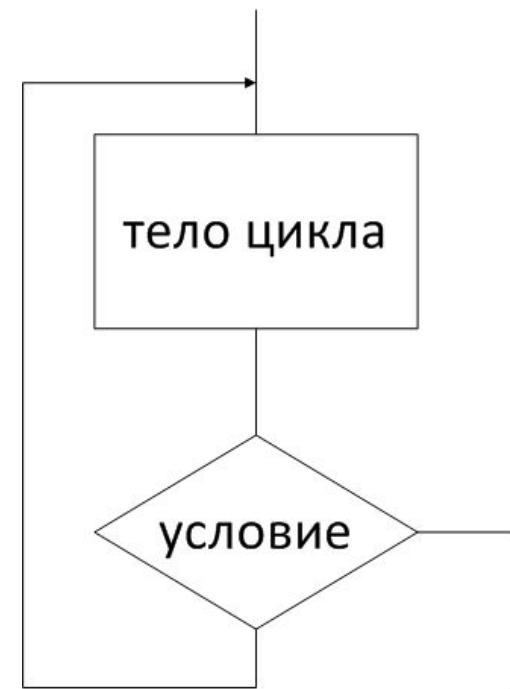
# Цикл с постусловием

## СЛОВЕСНОЕ ОПИСАНИЕ

*Цикл называется циклом с постусловием, если условие выхода из цикла стоит в конце, после тела цикла.*

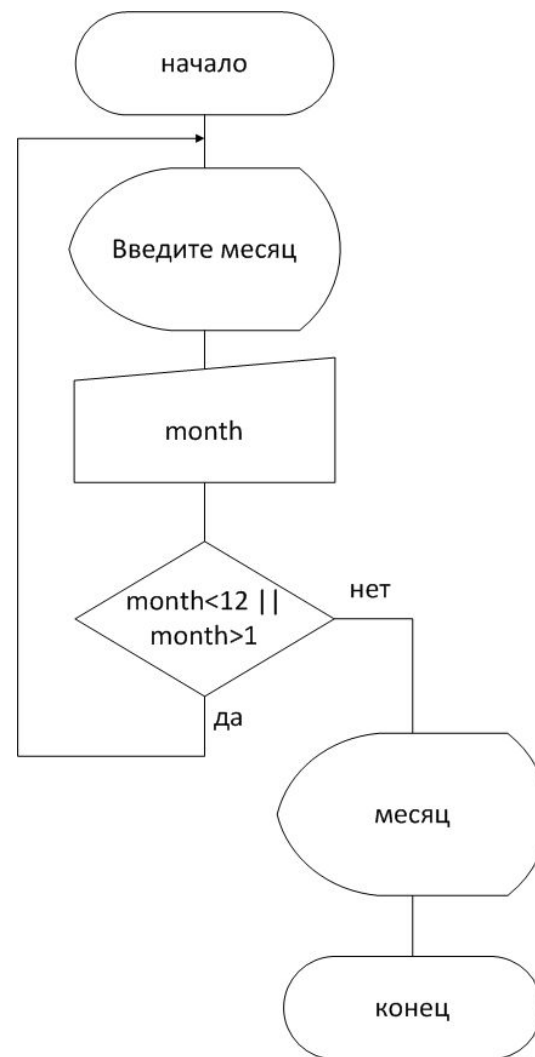
Цикл с постусловием выполняется обязательно, как минимум, один раз, независимо от того, истинно условие или нет.

## БЛОК-СХЕМА



# Пример

Цикл с постусловием удобно использовать при проверке вводимых пользователем данных. Например необходимо пользователю необходимо ввести номер месяца. Он не может быть отрицательным числом и находится в диапазоне от 1 до 12. Таким образом ввод будет продолжаться, пока пользователь не введёт корректное значение месяца.



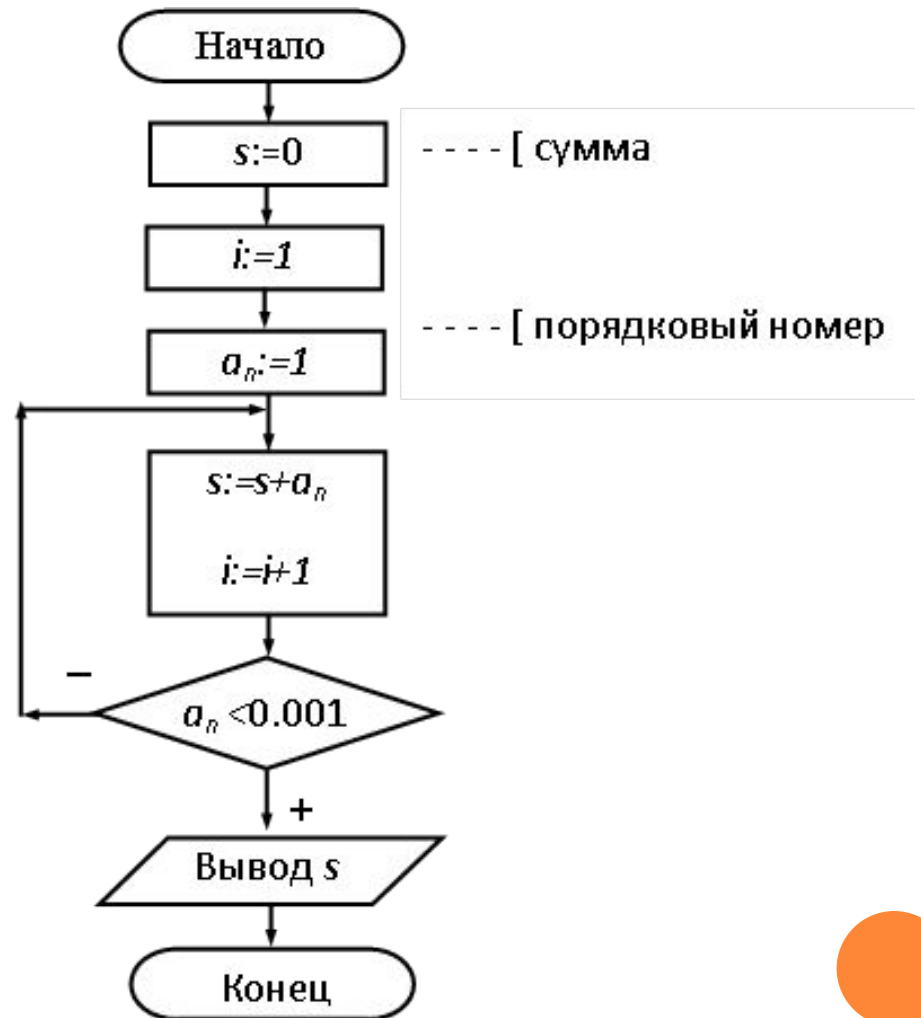


# Пример

Вычислить  
сумму ряда

$$1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \dots$$

с ТОЧНОСТЬЮ  
0.001.

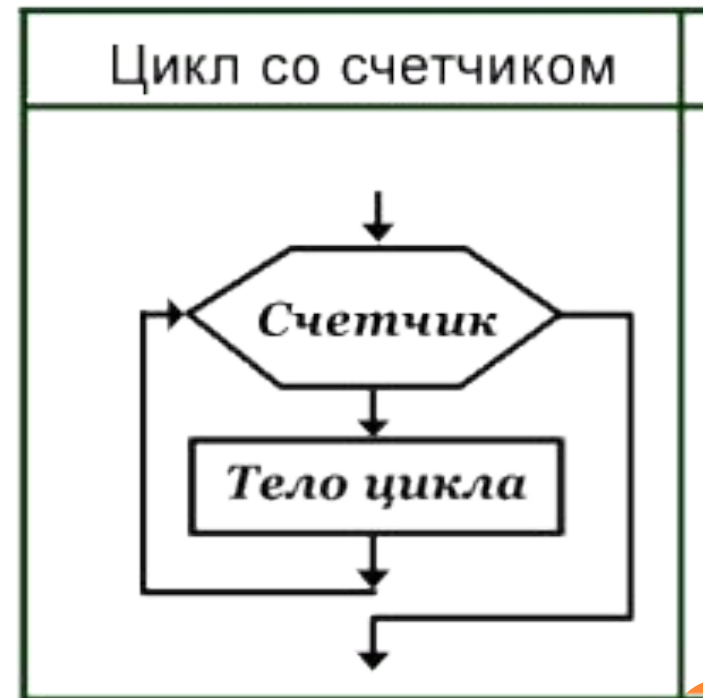


# Цикл с параметром

## СЛОВЕСНОЕ ОПИСАНИЕ

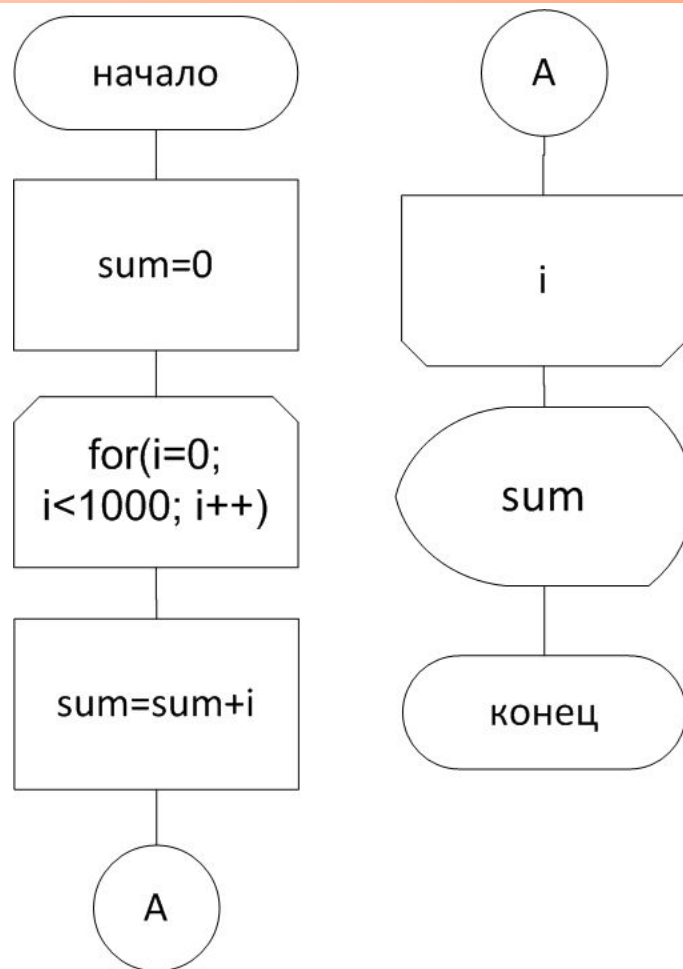
Алгоритмическая структура *цикл со счетчиком* используется, если известно заранее, какое число повторений тела цикла необходимо выполнить.

## БЛОК-СХЕМА



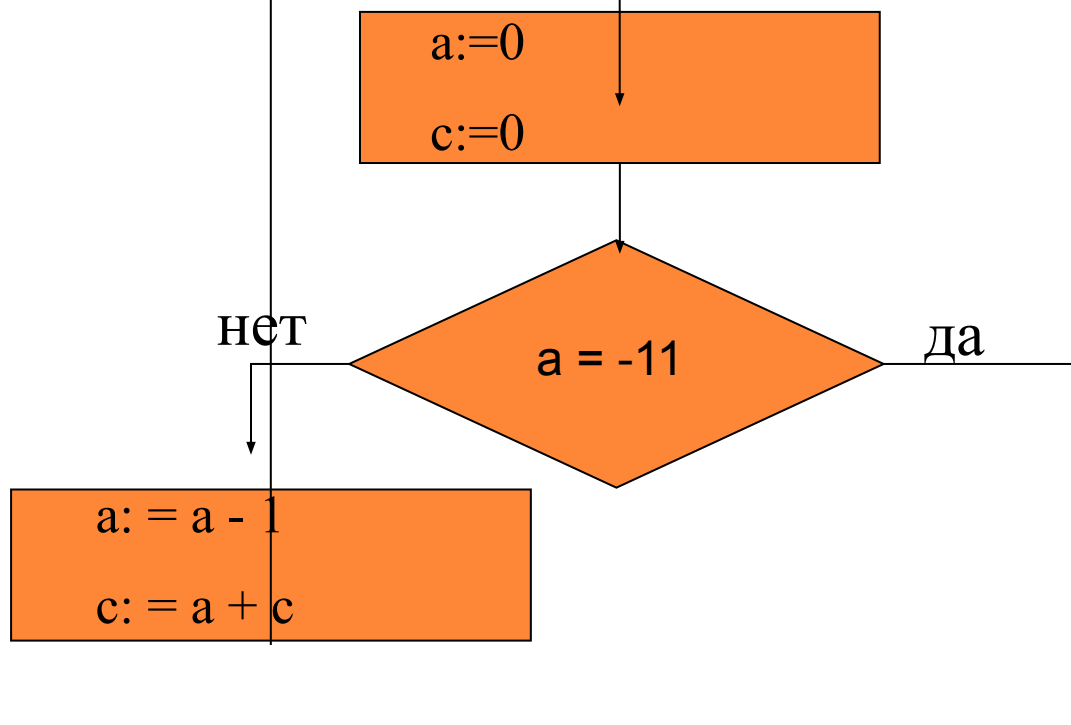
# Пример

Например, необходимо посчитать сумму чисел от 0 до 1000.



# Пример

Определите значение переменной **c** после выполнения фрагмента алгоритма:



1) - 11

2) - 78

3) - 55

4) - 66



# Пример

Определите значение переменной **m** после выполнения фрагмента алгоритма:

