

Лекция 6

***Понятие об алгоритмах.
Блок-схемы алгоритмов.
(в конце – самостоятельная!)***

Этапы решения задач на компьютере

- Постановка задачи
- Построение математической модели
- Разработка алгоритма
(алгоритмизация)
- Составление программы
- Реализация программы на
компьютере
- Анализ результатов

Понятие алгоритма

- *Алгоритм – это однозначное, точное и полное описание последовательности элементарных действий для решения данной задачи.*
- Слово *алгоритм* происходит от имени арабского математика Аль-Хорезми (точнее – латинизированной формы его имени – Algorithmi), который еще в IX веке сформулировал правила выполнения четырех арифметических действий. Эти правила называли правилами Аль-Хорезми (algorithmi), а позднее просто стали называть *алгоритмом*.

*Абу Джафар Мухаммад ибн Муса Аль-Хорезми,
скульптурный портрет
(790 – 840 гг.)*



Свойства алгоритма

- **Дискретность** — описываемый процесс должен быть разбит на последовательность отдельных шагов
- **Понятность** — предписания алгоритма должны быть понятны исполнителю
- **Определенность** — алгоритм не должен оставлять места для произвола исполнителя, т.е. после выполнения одного предписания д.б. ясно, какое следующее
- **Массовость** — применимость алгоритма не к одной, а к ряду однотипных задач
- **Результативность** — алгоритм должен состоять из конечного числа шагов и при этом д.б. получен результат

Способы записи алгоритма

- 1) на естественном языке (русском, английском и т.д.) ,
- 2) на алгоритмическом языке (или языке программирования: Бейсик, Паскаль и т.д.),
- 3) на языке блок–схем.

Блок-схема алгоритма

Блок–схема алгоритма – *это графический способ записи алгоритма, представляющий собой систему определенным образом связанных блоков, изображаемых в виде плоских геометрических фигур.*

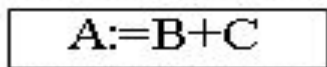
Элементы блок –схемы располагаются сверху вниз, линии соединения отдельных блоков показывают направление процесса обработки схемы. Каждое такое направление называют ветвью.

Основные блоки

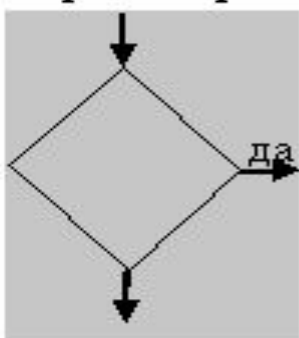
1. “Начало” и “конец” алгоритма изображаются овалом:



2. Блок “действия” изображается прямоугольником. Внутри его указываются необходимые вычисления и присваивания результата :



3. Блок “условия” изображаются ромбом. Внутри блока записываются условия выбора направления действия алгоритма:



4. Блоки “ввода” и “вывода” информации изображаются параллелограммами. С их помощью вводят исходные данные задачи, выводят результат решения :



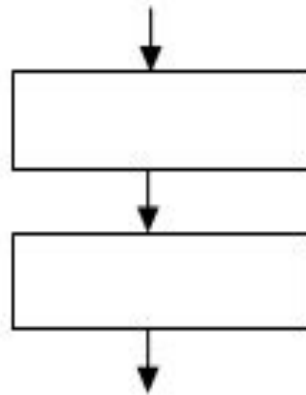
Основные структуры

Структурами называют ограниченный набор блоков и стандартных способов их соединения для выполнения типичных последовательностей действий.

- Следование
- Развилка (полная и неполная)
- Цикл (цикл-ПОКА и цикл-ДО, а также цикл с параметром)

Структура «следование»

1. Структура «следование» состоит из двух (или более) блоков «действие»:



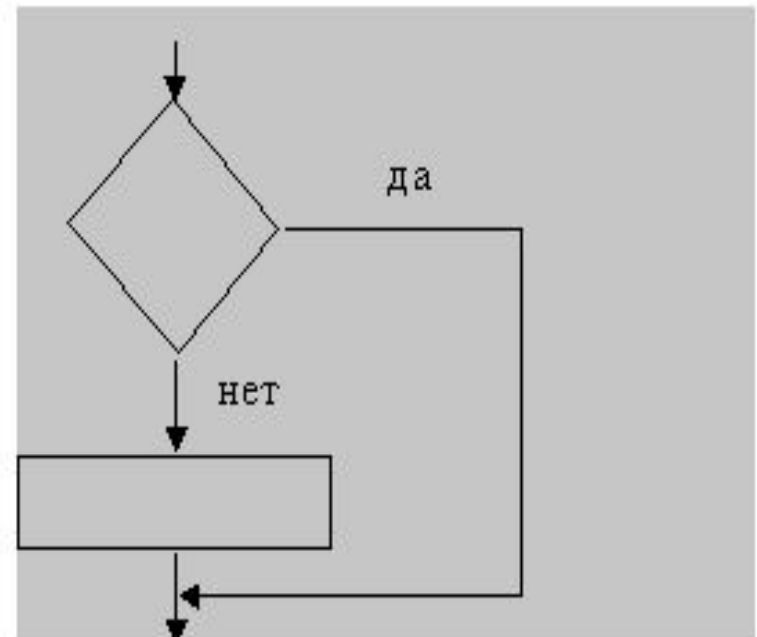
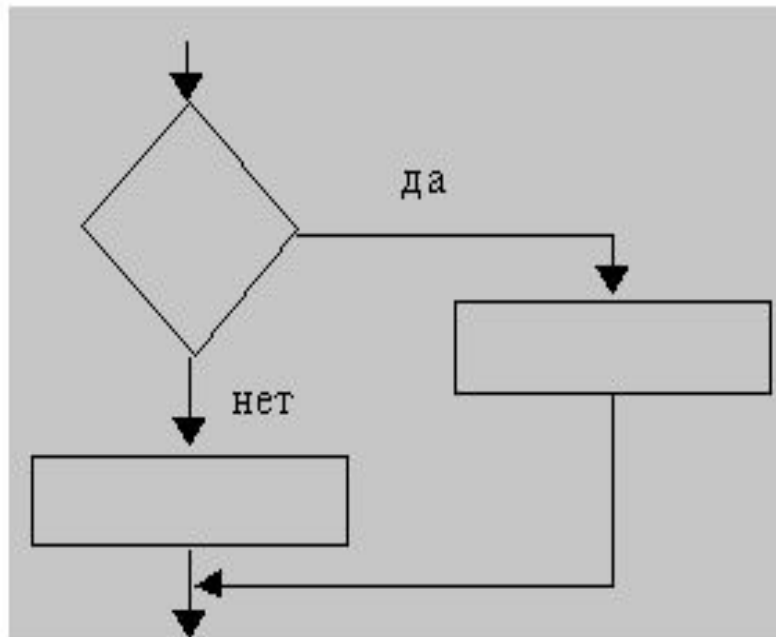
Однако в качестве элементов этой структуры могут выступать и базовые структуры. Именно поэтому правильнее называть элементы такой структуры **функциональными блоками** и в дальнейшем под функциональными блоками будем понимать не только блок «действие», но и любую базовую структуру.

Структура «развилка»

2. Структура «развилка» состоит из логического элемента с проверкой некоторого условия и функциональных блоков, которые в простейшем случае есть блок «действие».

«Развилка» может быть двух видов:

а) полная условная конструкция: б) неполная условная конструкция:

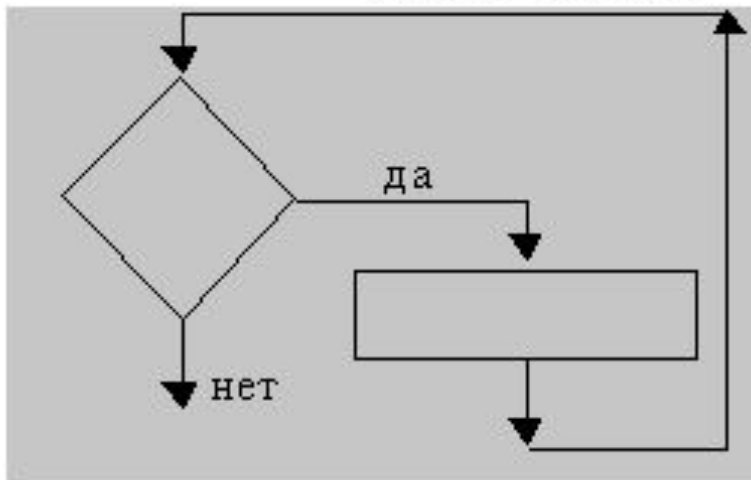


Структура «цикл»

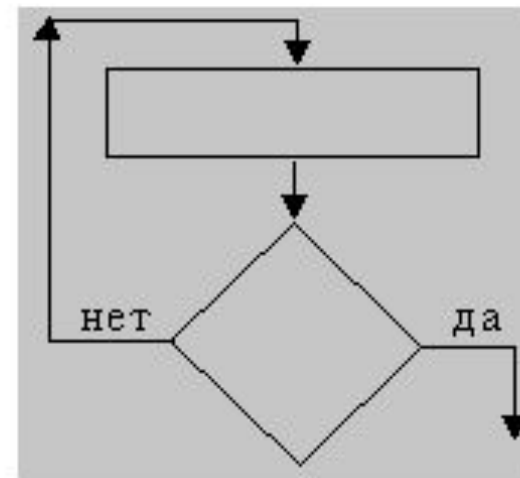
3. Структура «цикл» состоит из логического элемента с проверкой условия и функционального блока, называемого телом цикла. Ясно, что тело цикла может выполняться неоднократно.

Данная структура может быть двух видов:

цикл – ПОКА



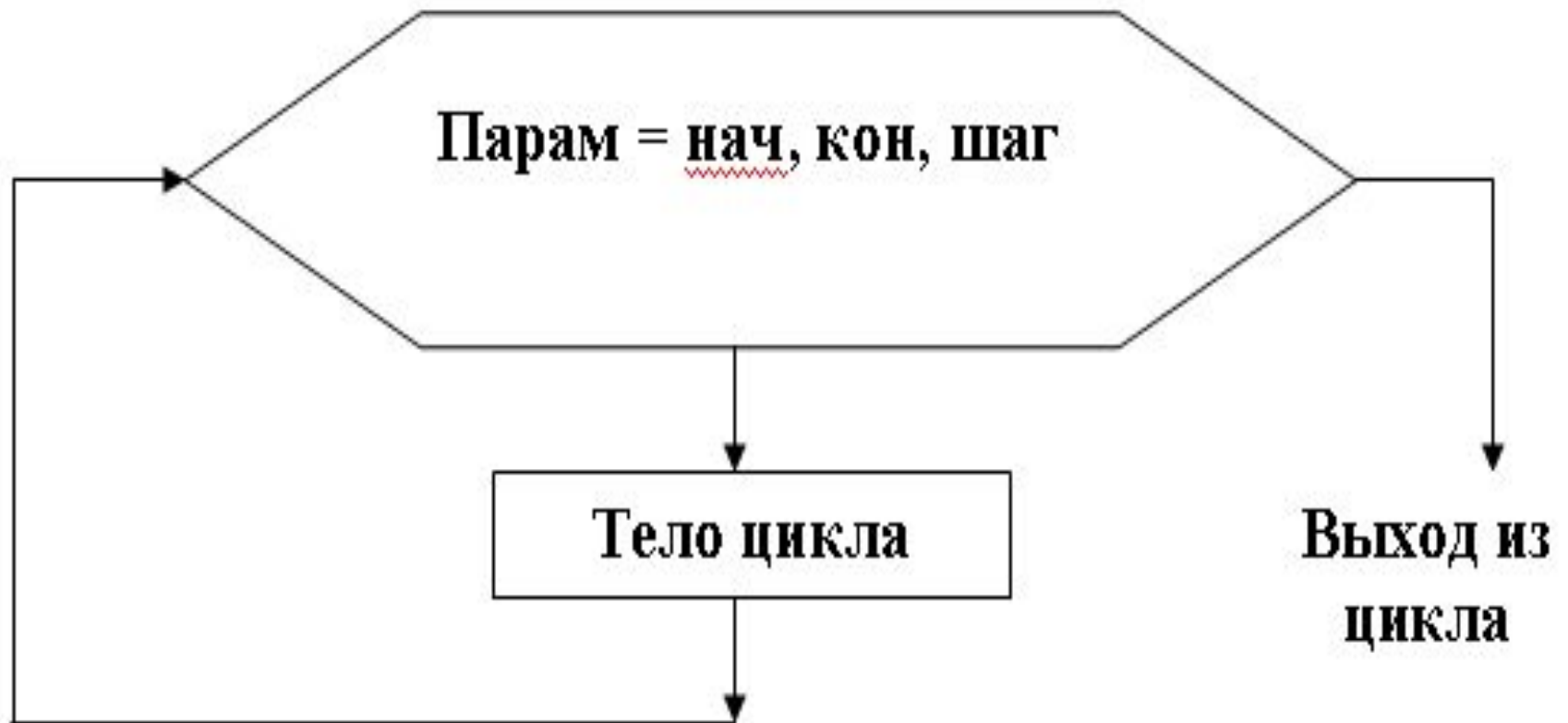
цикл – ДО



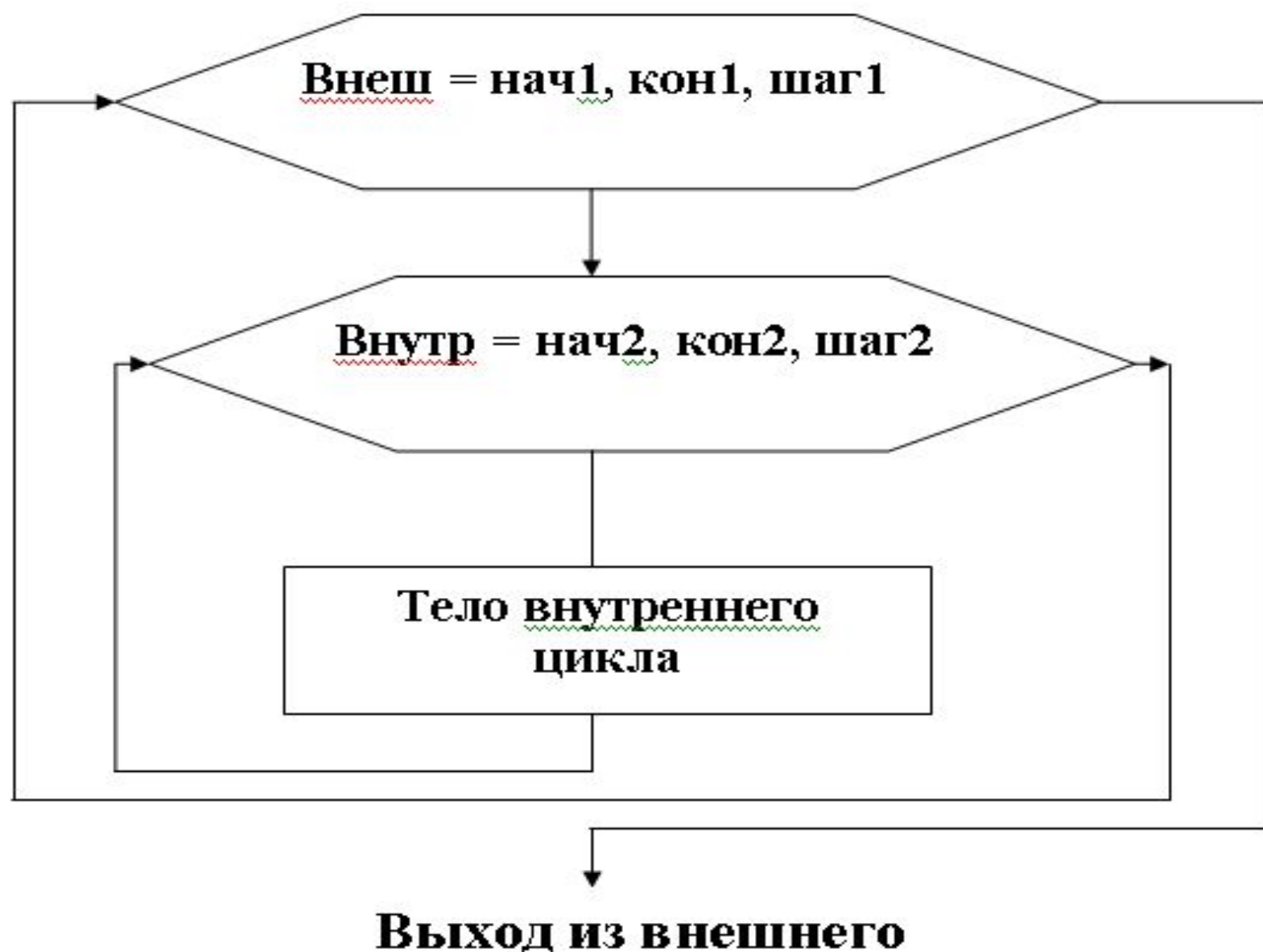
Особенности цикла-ПОКА и цикла-ДО

- В случае “цикл –ПОКА” функциональный блок размещен после проверки условия, поэтому может оказаться, что тело цикла не выполнится ни разу. Однако, если условие выполняется – выполняется и весь цикл. Проще говоря, “цикл – ПОКА” выполняется, п о к а выполняется условие.
- В “цикле – ДО” функциональный блок размещен до проверки выполнения условия, так что в этом варианте тело цикла в любом случае будет выполнено по крайней мере один раз. Условие в данном случае является условием выхода из цикла. Проще говоря, “цикл – ДО “ выполняется ДО наступления выполнения условия.

Цикл с параметром



Вложенный цикл



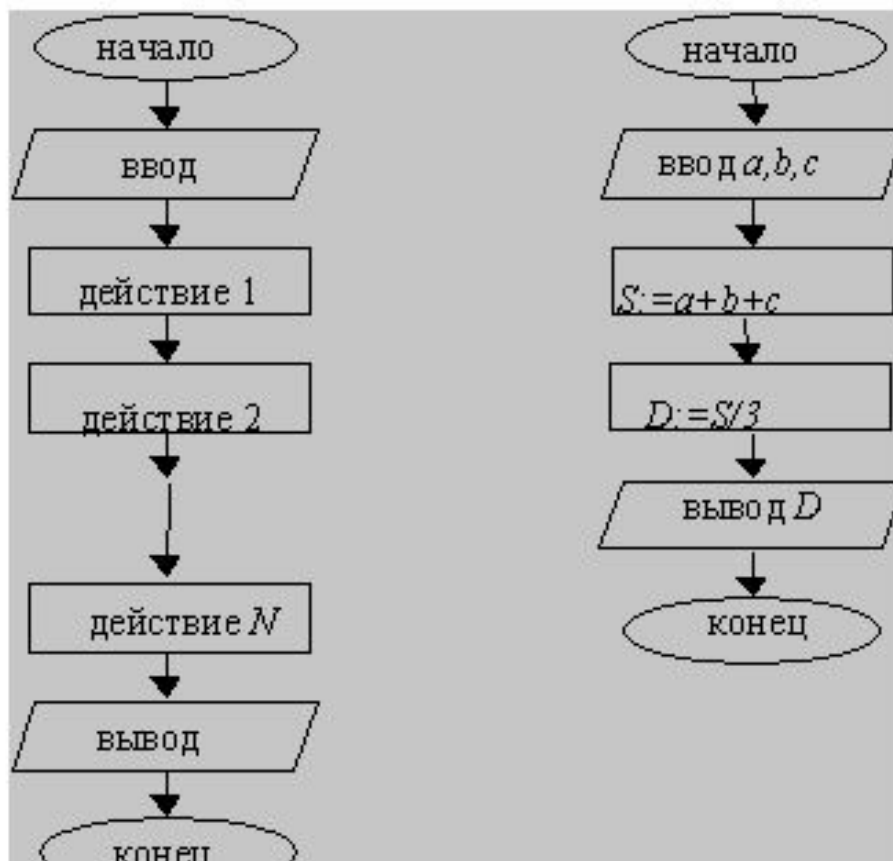
Основные виды алгоритмов

- **Линейный**
- **Разветвленный**
- **Циклический**

1. Линейный алгоритм не содержит логических условий, имеет одну ветвь обработки и изображается линейной последовательностью блоков.

Проще говоря, линейный алгоритм строится на основе структуры “следование”.

Условное изображение линейного алгоритма:

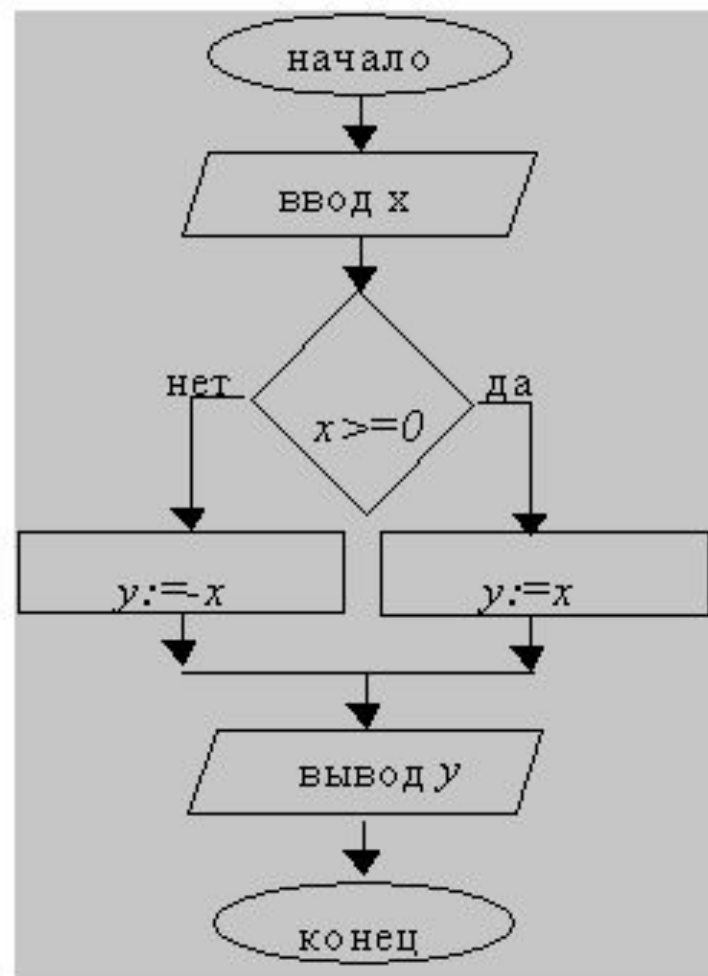
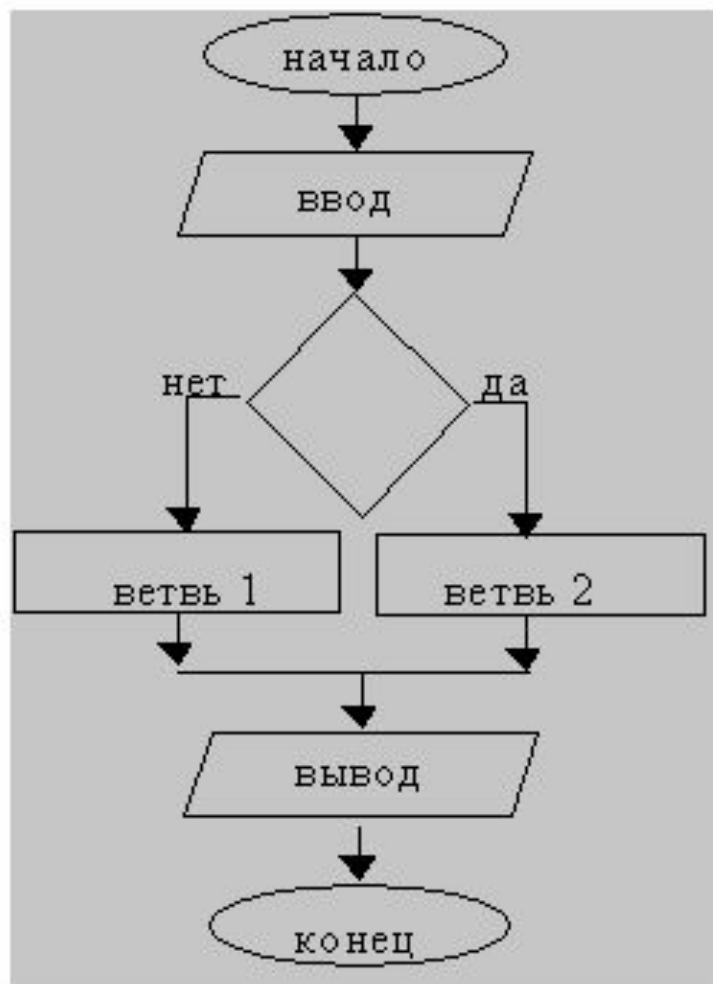


**Запись
линейного алгоритма
на языке Блок-схем.**

В качестве примера можно привести алгоритм вычисления, среднего арифметического трех чисел a, b, c (на рисунке изображено справа).

2. Разветвленный алгоритм содержит одно или несколько логических условий и имеет несколько ветвей обработки. Иначе говоря, разветвленный алгоритм строится на основе структуры “развилка”.

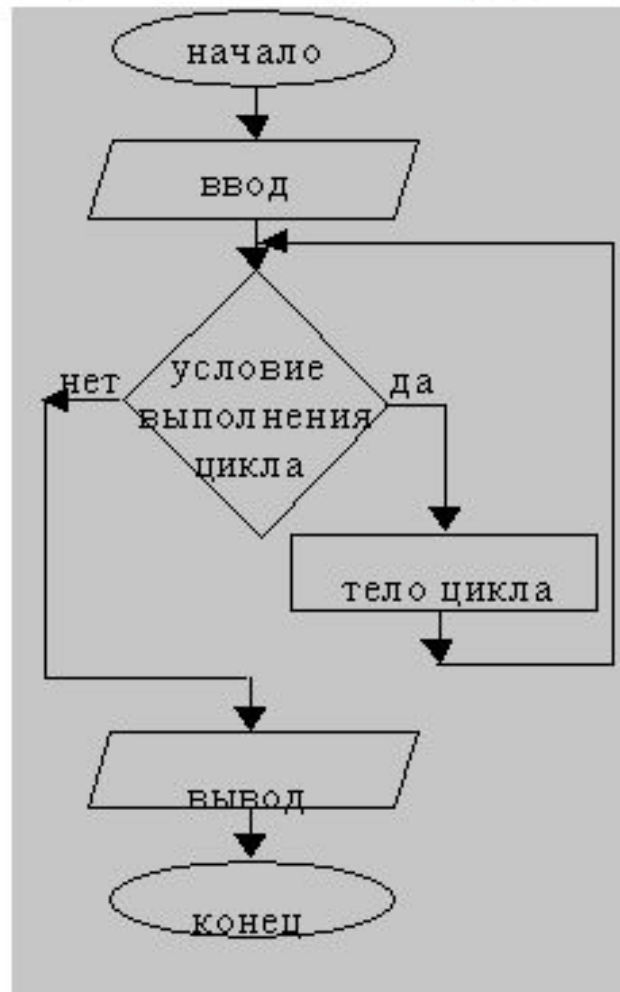
Условное обозначение такого алгоритма:



3. Циклический алгоритм содержит один или несколько циклов. Он строится, таким образом, на основе структуры “цикл”. Все ветви алгоритма должны, в конце концов сойтись.

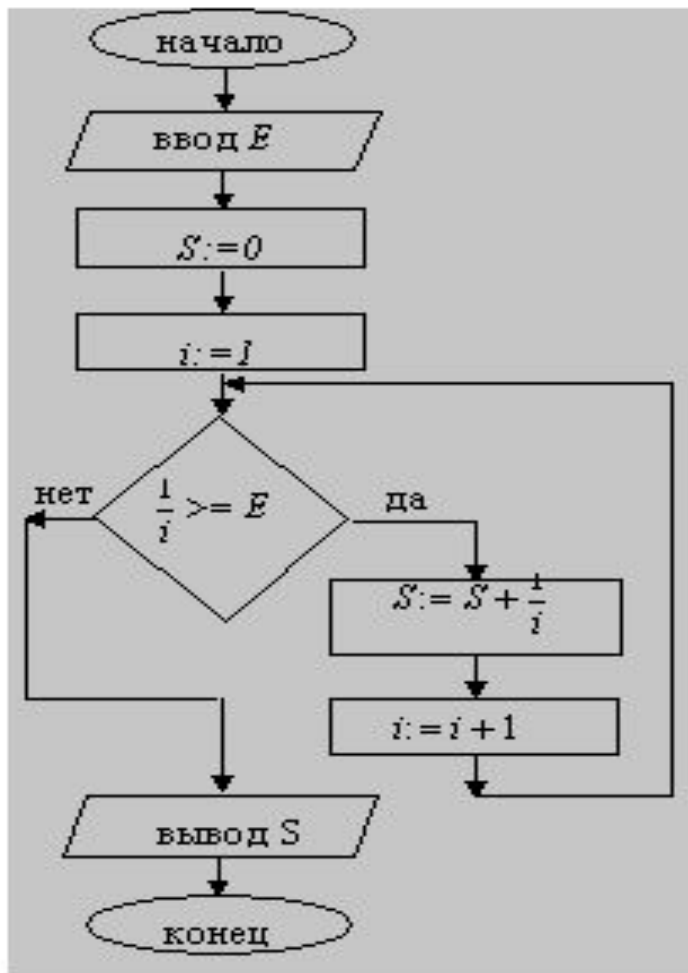
Условное обозначение циклического алгоритма:

(на основе цикла – ПОКА)

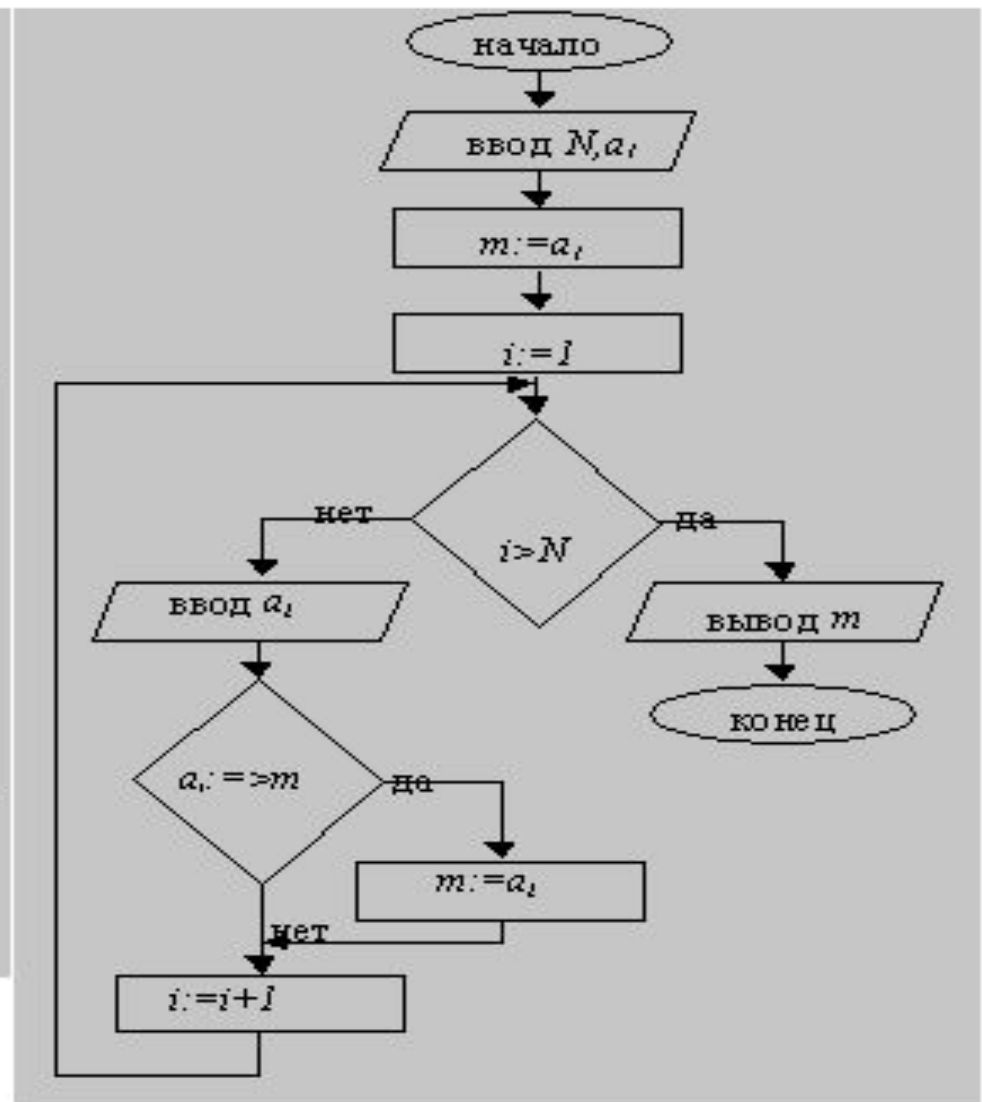


(на основе цикла – ДО)

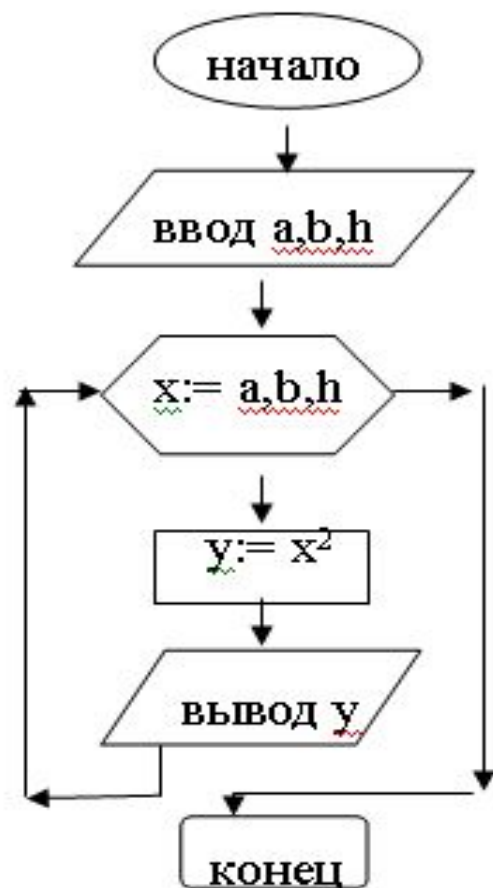




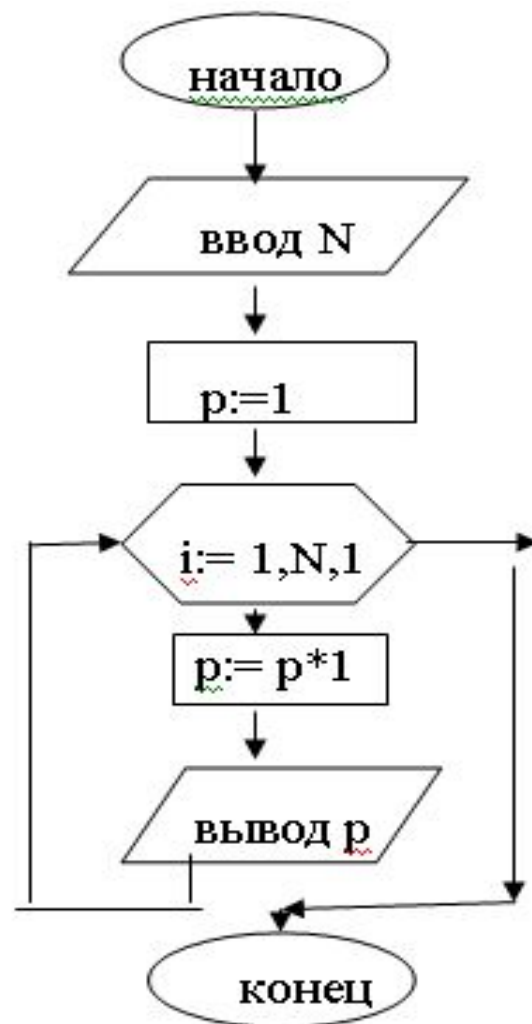
*Алгоритм вычисления суммы:
 $1 + 1/2 + 1/3 + \dots$ с заданной
 точностью (цикл – ПОКА)*



*Алгоритм поиска наибольшего
 из N последовательно
 вводимых чисел (цикл – ДО)*



Блок-схема алгоритма вычисления функции $y=x^2$ на интервале $[a,b]$ с шагом h . (цикл-Для)



Блок-схема алгоритма вычисления факториала числа N. (цикл со счетчиком)

Практические задания

1. Линейный алгоритм

- 1.1. Вычислить площадь треугольника со сторонами a , b , c по формуле Герона:
- 1.2. Дана длина ребра куба. Найти объем куба и площадь его боковой поверхности.
- 1.3. Найти площадь равнобокой трапеции с основаниями a и b и углом при большем основании x .
- 1.4. Найти угол между отрезком прямой, соединяющей начало координат с точкой $A(x, y)$ и осью OX (точка лежит в 1-й четверти).

2. Алгоритмы с ветвлением

- 2.1. Даны числа a , b , c . Проверить выполняется ли неравенство $a < b < c$.
- 2.2. Даны три действительных числа. Выбрать из них те, которые принадлежат интервалу $(1, 3)$.
- 2.3. Даны числа X , Y ($X \neq Y$). Меньшее из этих двух заменить их полусуммой, а большее – их удвоенным произведением.
- 2.4. Найти наибольшее из трех заданных чисел.

3. Циклический алгоритм

- 4.1. Дано 10 вещественных чисел: a_1, a_2, \dots, a_{10} . Найти порядковый номер того из них, которое наиболее близко к какому-нибудь целому числу.
- 4.2. Дано 10 вещественных чисел. Определить, сколько из них принимает значение, большее заданного A .
- 4.3. Дано 100 чисел. Определить, сколько из них больше своих соседей, т.е. предыдущего и последующего чисел.
- 4.4 . Вычислить K – количество точек с целочисленными координатами, попадающих в круг радиуса R с центром в начале координат.