

# АЛГОРИТМ

Понятие алгоритм – одно из фундаментальных в информатике.

Алгоритм — это совокупность правил выполнения определенных действий, обеспечивающих решение задачи.

В жизни мы постоянно выполняем разные алгоритмы.

Составляем распорядок дня, чтобы многое успеть.



# ПРОГРАММА

Каждый исполнитель имеет свою систему команд (СКИ).

Программа — это алгоритм, записанный на языке исполнителя.

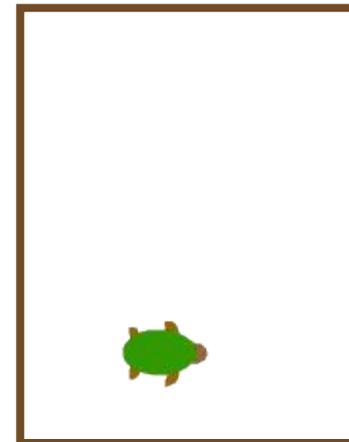
**Рассмотрим пример:** возьмем учебного исполнителя Черепашку. Пусть этот исполнитель имеет три команды: вперед (1 см), направо( $90^{\circ}$ ), налево ( $90^{\circ}$ ).

Исходное положение исполнителя:



Какой код программы надо написать, чтобы Черепашка начертила букву Г ?

Код программы будет выглядеть так:  
налево ( $90^{\circ}$ )  
вперед(1 см)  
вперед 1 см  
направо( $90^{\circ}$ )  
вперед(1см)



# СВОЙСТВА АЛГОРИТМА

## (Требования к составлению алгоритма)

1. **Дискретность.** Процесс решения задачи должен быть разбит на последовательность отдельных шагов.
2. **Однозначность (точность).** Команды алгоритма должны быть точно определены (например, нельзя написать 3-4 стакана муки, надо указать 3 стакана).
3. **Результативность.** После выполнения всех команд алгоритма, должен быть получен результат.
4. **Универсальность (массовость).** Важное свойство при решении задач на ЭВМ. Алгоритм должен быть применим для решения ни одной конкретной задачи, а для некоторого класса задач. Например, для решения квадратного уравнения с разными коэффициентами).
5. **Понятность.** Алгоритм должен быть написан на языке понятном исполнителю.



# СПОСОБЫ ОПИСАНИЯ АЛГОРИТМА

Так часто бывает, что алгоритм составляет один автор, а пишет программу другой человек. Алгоритмы бывают очень сложными и большими по объему. Бывает, что над алгоритмом трудятся сразу несколько человек. Учитывая все эти причины и еще ряд других, алгоритмы записывают или описывают на бумажных или электронных носителях.

## Как можно описать алгоритм?

1. **Словами.** Например, распорядок дня.
2. **Графически (блок-схемой).** Так делают программисты.
3. **Алгоритмическим языком (псевдокод)** – это учебный язык. Он применяется во многих тестах по информатике.
4. **Таблицей.**



# ОСНОВНЫЕ БЛОКИ

## ГРАФИЧЕСКОГО ОПИСАНИЯ АЛГОРИТМА

Блоки

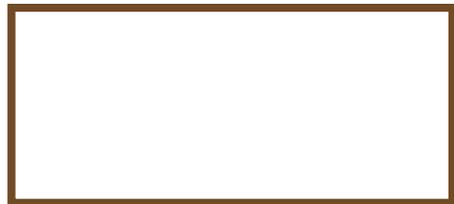


Что ими обозначают

Начало/конец алгоритма



Ввод/вывод данных



Обработку данных

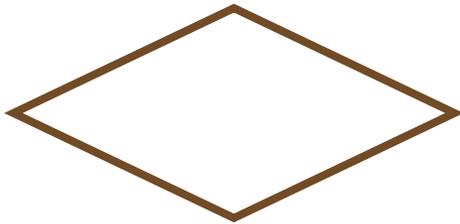


# ОСНОВНЫЕ БЛОКИ

## ГРАФИЧЕСКОГО ОПИСАНИЯ АЛГОРИТМА

Блоки

Что ими обозначают



Проверку условия



Начало цикла FOR/ NEXT



Подпрограмму



# Виды алгоритмов

---

- линейные;
  - ветвящиеся;
  - циклические.
-

# Линейные алгоритмы

---

***В линейном алгоритме*** операции выполняются последовательно, в порядке их записи. Каждая операция является самостоятельной, независимой от каких-либо условий. На схеме блоки, отображающие эти операции, располагаются в линейной последовательности.

Линейные алгоритмы имеют место, например, при вычислении арифметических выражений, когда имеются конкретные числовые данные и над ними выполняются соответствующие условию задачи действия.

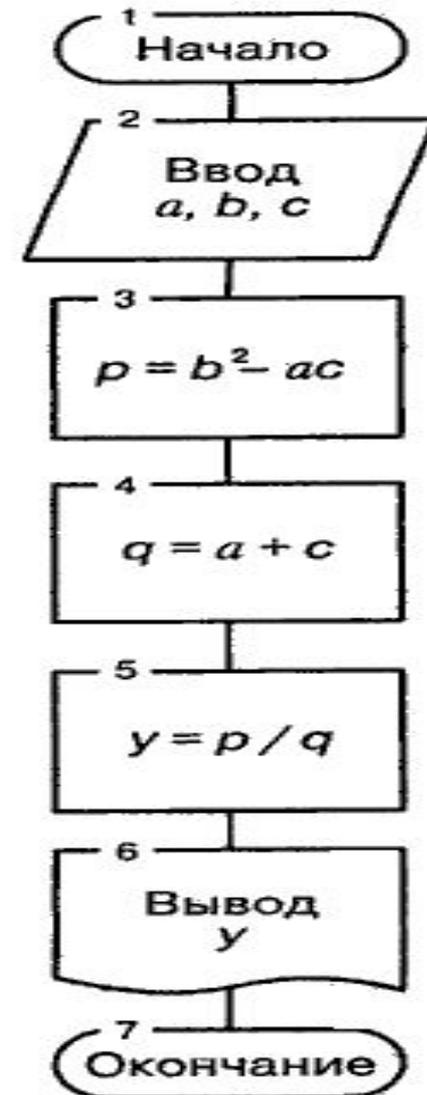
---

# Пример линейного алгоритма

---

Составить блок – схему  
алгоритма вычисления  
арифметического выражения  
 $y = (b^2 - ac) : (a + c)$

---



# Алгоритм с ветвлением

---

Алгоритм называется **ветвящимся**, если для его реализации предусмотрено несколько направлений (ветвей). Каждое отдельное направление алгоритма обработки данных является отдельной ветвью вычислений.

Ветвление в программе — это выбор одной из нескольких последовательностей команд при выполнении программы. Выбор направления зависит от заранее определенного признака, который может относиться к исходным данным, к промежуточным или конечным результатам. Признак характеризует свойство данных и имеет два или более значений.

Ветвящийся процесс, включающий в себя две ветви, называется **простым**, более двух ветвей — **сложным**.

~~Сложный ветвящийся процесс можно представить с помощью простых ветвящихся процессов.~~

# Алгоритм с ветвлением

---

Направление ветвления выбирается логической проверкой, в результате которой возможны два ответа:

1. «да» — условие выполнено
2. «нет» — условие не выполнено.

Следует иметь в виду, что, хотя на схеме алгоритма должны быть показаны все возможные направления вычислений в зависимости от выполнения определенного условия (или условий), при однократном прохождении программы процесс реализуется только по одной ветви, а остальные исключаются.

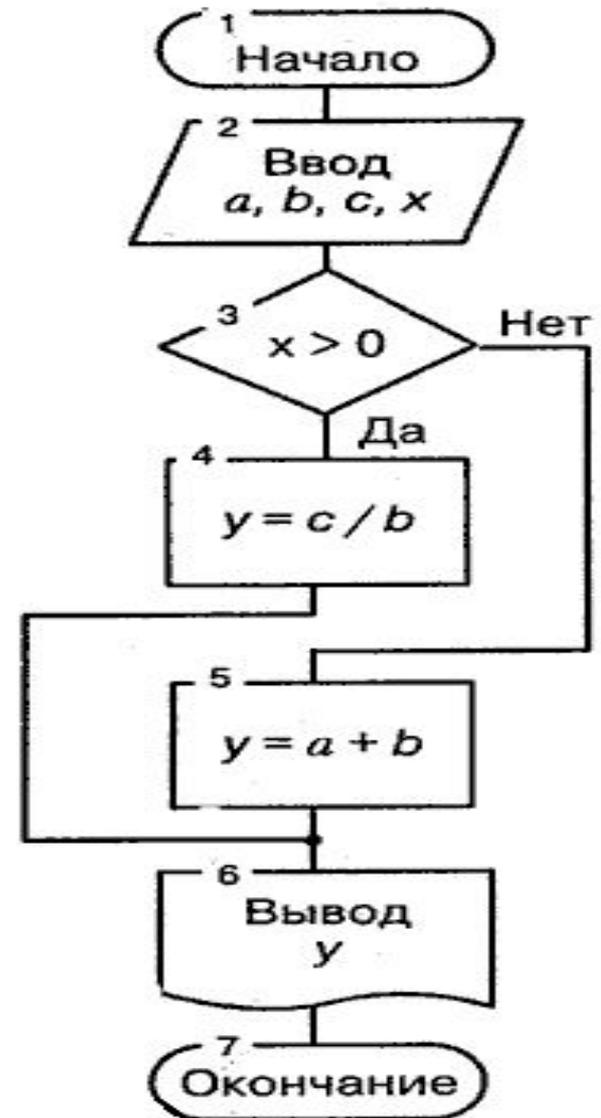
Важно! Любая ветвь, по которой осуществляются вычисления, должна приводить к завершению вычислительного процесса.

---

# Пример алгоритма с ветвлением

Составить блок-схему алгоритма с ветвлением для вычисления следующего выражения:

$Y = (a+b)$ , если  $X < 0$ ;  
 $c/b$ , если  $X > 0$ .



# Циклические алгоритмы

---

**Циклическими** называются алгоритмы, содержащие циклы.

**Цикл** — это многократно повторяемый участок алгоритма.

---

# Этапы организации цикла

---

- подготовка (инициализация) цикла (**И**);
- выполнение вычислений цикла (тело цикла) (**Т**);
- модификация параметров (**М**);
- проверка условия окончания цикла (**У**).

Порядок выполнения этих этапов, например, **Т** и **М**, может изменяться.

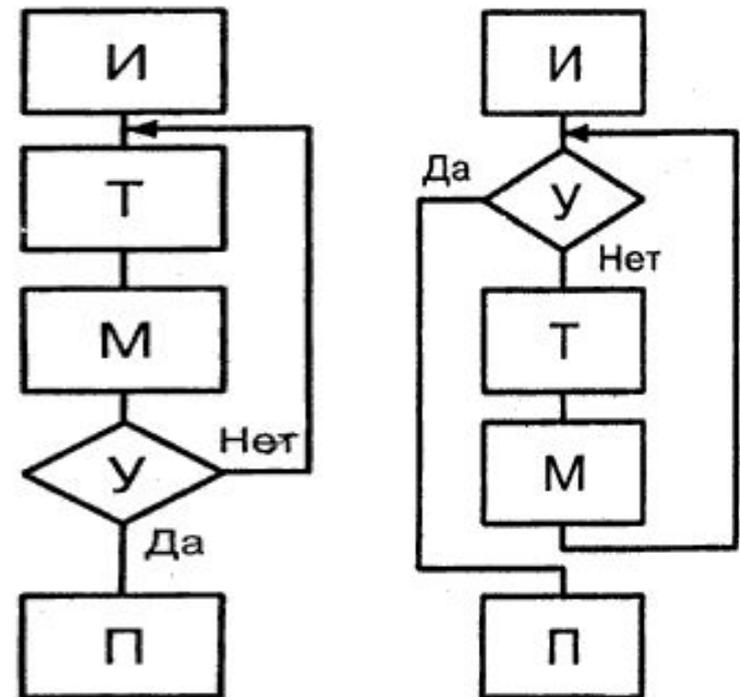
---

# Типы циклов

В зависимости от расположения проверки условия окончания цикла различают циклы с нижним и верхним окончаниями.

Для цикла с нижним окончанием (рис. а) тело цикла выполняется как минимум один раз, так как сначала производятся вычисления, а затем проверяется условие выхода из цикла.

В случае цикла с верхним окончанием (рис. б) тело цикла может не выполняться ни разу в случае, если сразу соблюдается условие выхода.



а

б

*Примеры циклических алгоритмов*

# Виды циклов

---

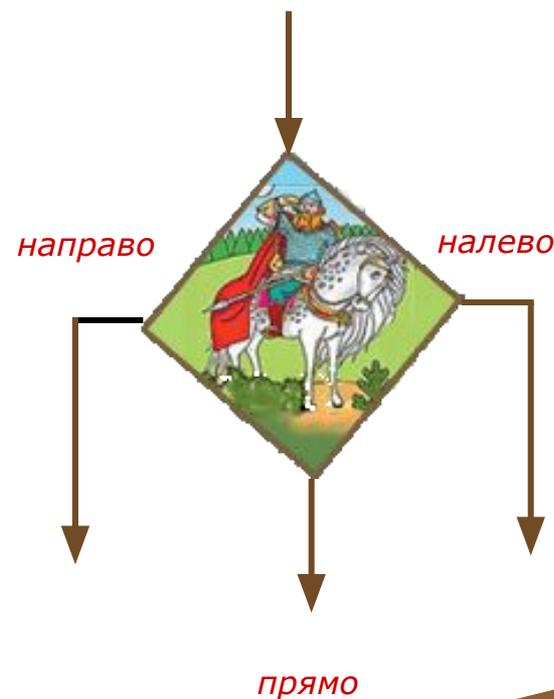
- Цикл называется **детерминированным**, если число повторений тела цикла заранее известно или определено.
  - Цикл называется **итерационным**, если число повторений тела цикла заранее неизвестно, а зависит от значений параметров (некоторых переменных), участвующих в вычислениях.
-

# ТИПЫ АЛГОРИТМОВ

**Линейный.** Команды такого алгоритма выполняются последовательно сверху вниз. Например, *нахождение гипотенузы прямоугольного треугольника по двум его катетам.*

**Разветвляющийся.** В зависимости от поставленного условия алгоритм позволяет выбрать один из вариантов решения задачи. Примерами могут быть *нахождение корней квадратного уравнения или богатырь на распутье из русских сказок.*

**Циклический.** В алгоритме встречаются повторяющиеся действия. Например, *при заучивании стихотворения вам придется перечитывать и повторять одни и те же строки.*



---

**Домашняя работа  
конспект в  
тетради, стр  
22-29, 22-32 чит.,  
учить.**

