

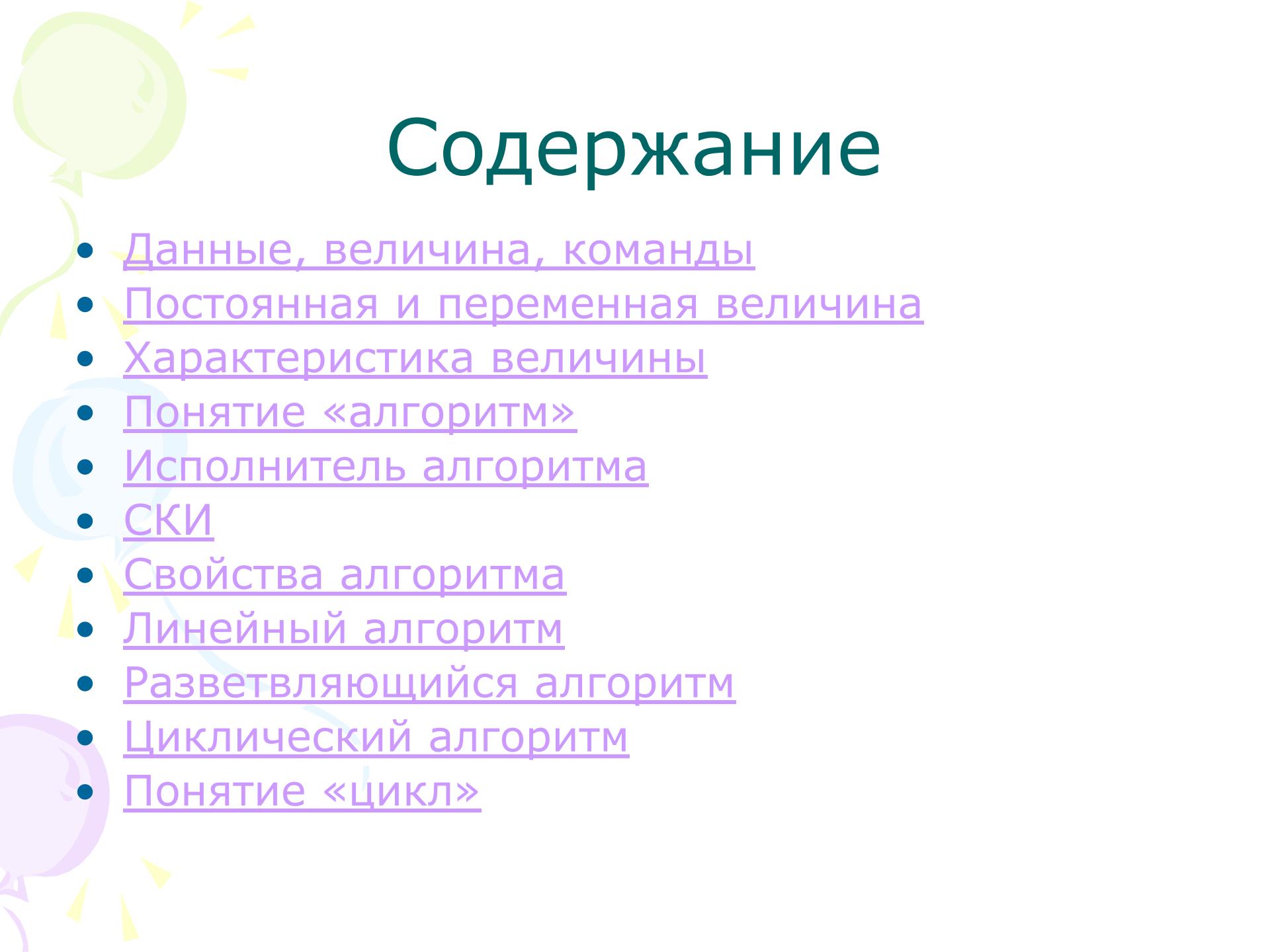
# Тема

*Алгоритмы*

*Виды алгоритмов*

*Свойства алгоритмов*

МБОУ «СОШ № 46 г.Белгорода»,  
Учитель информатики и ИКТ  
Голубятникова Т.В.



# Содержание

- Данные, величина, команды
- Постоянная и переменная величина
- Характеристика величины
- Понятие «алгоритм»
- Исполнитель алгоритма
- СКИ
- Свойства алгоритма
- Линейный алгоритм
- Разветвляющийся алгоритм
- Циклический алгоритм
- Понятие «цикл»

Всё, что бы мы ни делали, чаще всего имеет какую-либо цель. И не всегда эта цель достигается.

Если точно и правильно сформулировать желаемый результат, а потом продумать чёткий план его содержания, то эта цель будет достигнута



[начало](#)



- **Данные** – это информация, обрабатываемая компьютером.
- **Величина** – это отдельная единица данных.
- **Команды** – позволяют определить действия в компьютерной программе над величинами.

начало





# *По отношению к программе данные могут быть*

- ❖ исходные
- ❖ промежуточные
- ❖ результаты

[начало](#)



- **Постоянная** величина – величина, значение которой не изменяется в процессе исполнения алгоритма, а остается одним и тем же, указанным в тексте алгоритма.



- **Переменная** величина величина, значение которой меняется в процессе исполнения алгоритма.

[начало](#)

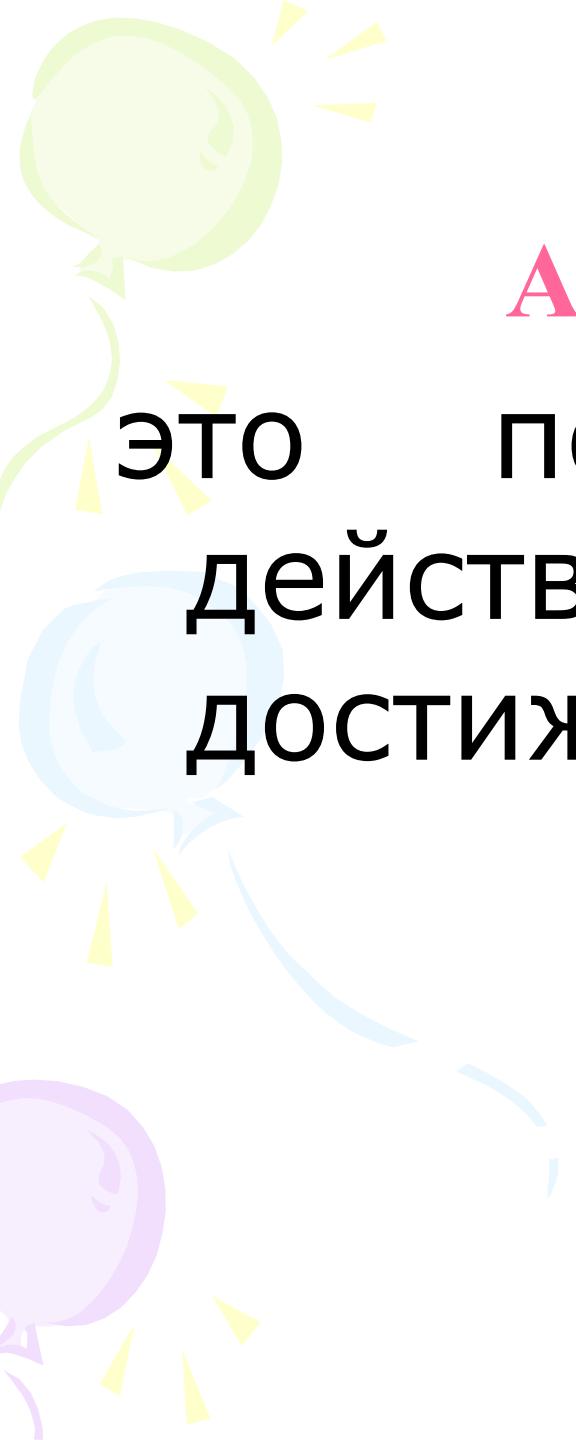


## **Характеристики величины:**

*Имя* (идентификатор) — это обозначение величины и место в памяти.

*Тип* — множество допустимых значений и множество применимых операций к величине.

*Значение* — характеристика, может меняться многократно в ходе исполнения алгоритма.



**Алгоритм -**

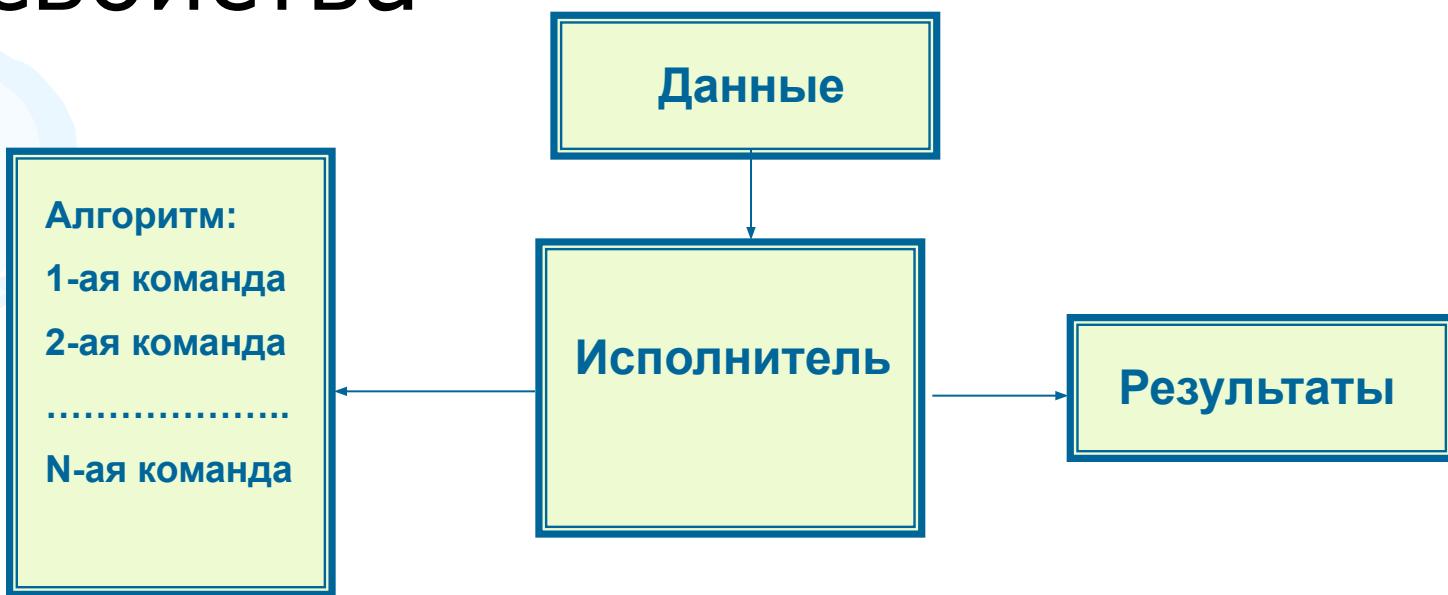
это последовательность действий, приводящая к достижению результата



[начало](#)



В определении «алгоритм» содержатся основные понятия, связанные с ним и его главные свойства



Взаимосвязь понятий:

[начало](#)



# Исполнитель

Центральным объектом в схеме является **Исполнитель** – это тот объект (или субъект) для управления которым составляется алгоритм



# СКИ

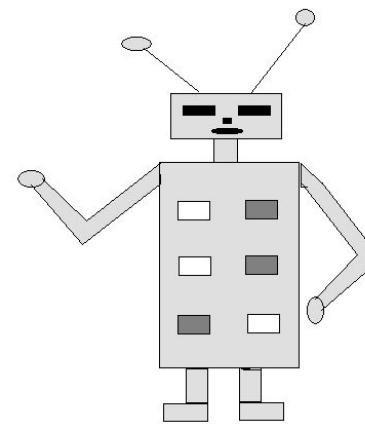
Основной характеристикой исполнителя, с точки зрения управления, является **система команд исполнителя (СКИ)** - это конечное множество команд, которые понимает исполнитель, т.е. умеет их выполнять

[начало](#)



Для выполнения всякой работы, решения поставленной задачи исполнитель на входе получает алгоритм и исходные данные, а на выходе - требуемые результаты. Алгоритм может включать в себя **только команды,**

*входящие в СКИ*



# Свойства алгоритмов:

- **Результативность** (или конечность) – выполнение алгоритма должно приводить к результату за конечное число шагов;
- **Дискретность** (или детализация) – алгоритм поддаётся расчленению на элементарные (дискретные) шаги, которые могут быть исполнены при помощи системы команд исполнителя;

[начало](#)



## Свойства алгоритмов:

- **Однозначность** – каждый шаг исполнителя может и должен быть истолкован одним и только одним способом;
- **Понятность** – алгоритм должен быть составлен только из команд, входящих в систему команд исполнителя;



# Свойства алгоритмов:

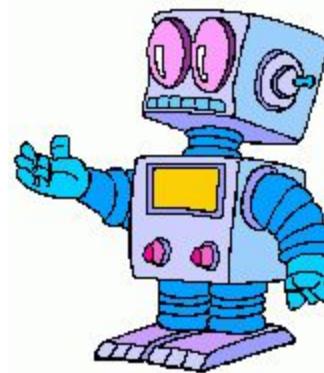
- **Массовость** – алгоритм должен решать однотипные задачи с различными исходными данными;
- **Переносимость**  
(или совместимость) – алгоритм не должен зависеть от типа используемой вычислительной техники или выбранного языка программирования;

[начало](#)



## Виды алгоритмов

Существует три основных вида алгоритмов, которые являются базовыми при написании программ



# Первый тип алгоритмов

**Линейный алгоритм** – это алгоритм, в котором все выполняются в строгом порядке, последовательно, одно за другим



Например: включение персонального компьютера

[начало](#)



# Второй тип алгоритма

Алгоритм, в котором осуществляется выбор действий в зависимости от какого-то условия, называют **разветвляющимся**

[начало](#)



## Пример разветвляющегося алгоритма



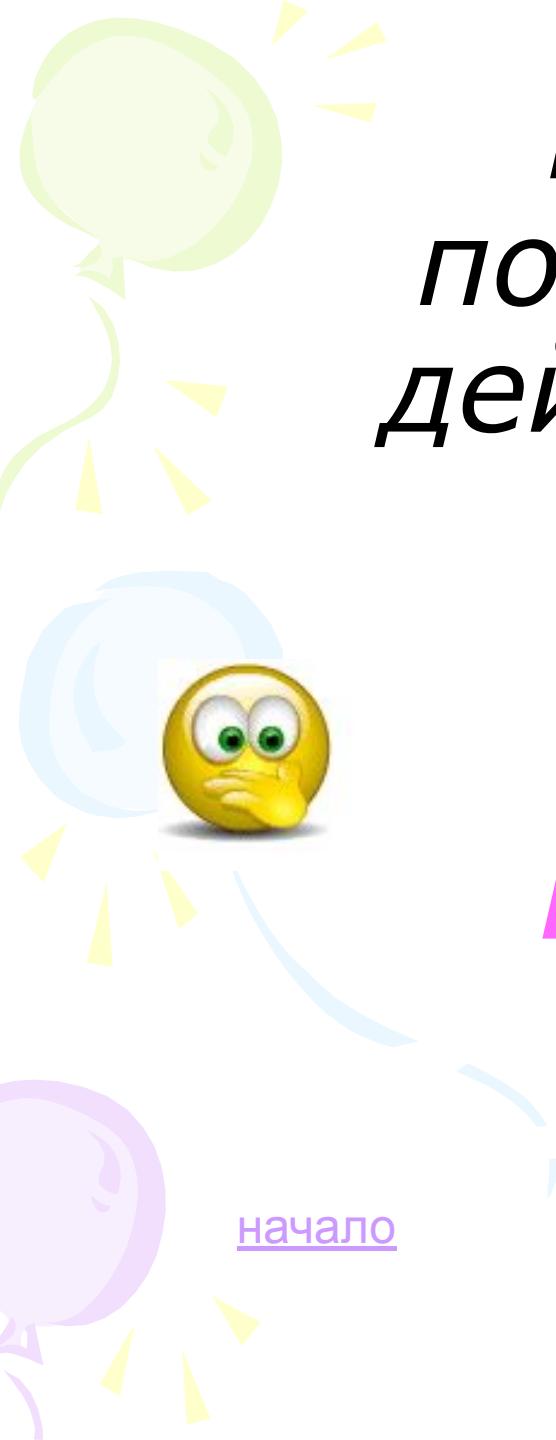
# Третий тип алгоритмов

## Циклический алгоритм –

**это** алгоритм, содержащий повторяющие действия с какой-либо изменяющейся величиной  
**(параметром)**

начало





Повторяющаяся  
последовательность  
действий называется  
**циклом,**



а эти действия –  
**циклическими**

начало



## Пример циклического алгоритма

