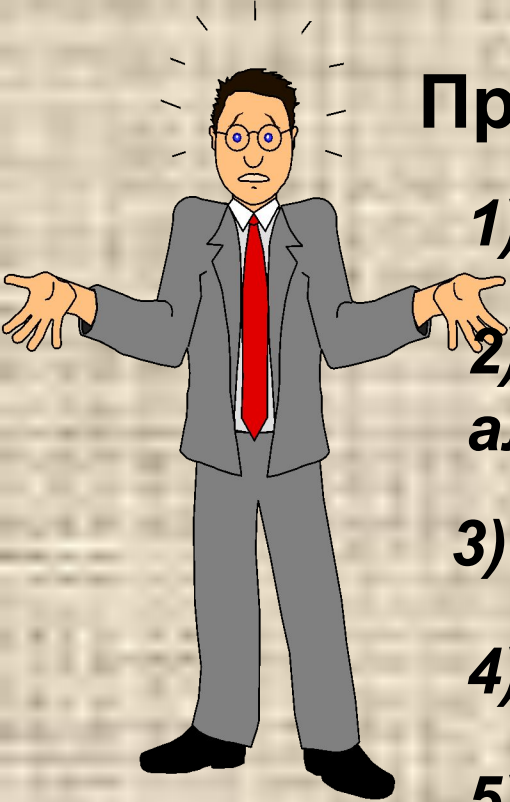


Информатика 9 класс

# Алгоритм и алгоритмические структуры





## Проверка домашнего задания

- 1) *Что такое алгоритм?*
- 2) *Перечислите основные свойства алгоритмов*
- 3) *Объясните свойство «дискретность»*
- 4) *Объясните свойство «понятность»*
- 5) *Объясните свойство «точность»*
- 6) *Объясните свойство «результативность»*
- 7) *Объясните свойство «массовость»*
- 8) *Как эти знания могут вам пригодиться в жизни?*

# Цели урока:

- Познакомиться с типами алгоритмов и с основными алгоритмическими структурами
- Развивать алгоритмическое мышление – искусство размышлять, умение планировать свои действия, способность предусматривать различные обстоятельства и поступать соответственно с ними



# Свойства алгоритма

**Дискретность**

Путь решения задачи разделён на отдельные шаги

**Понятность**

Алгоритм состоит из команд, входящих в СКИ

**Точность**

Команды понимаются однозначно

**Результативность**

Обеспечивается получение ожидаемого результата

**Массовость**

Обеспечивается решение задач с различными исходными данными

**Задание: Найти недостатки в алгоритмах**

## ***1. Алгоритм варки манной каши***



Молоко вскипятить добавить соль, сахар, засыпать тонкой струйкой, непрерывно помешивая манную крупу, довести до кипения, прокипятить минут 5-7, добавить масло и дать остыть.

# Алгоритм варки манной каши

1. Включить плиту
2. Влить в кастрюлю 1,5 литра молока
3. Добавить 5 грамм соли, 15 грамм сахара
4. Довести молоко до кипения
5. 8 столовых ложек манной крупы засыпать тонкой струйкой, непрерывно помешивая молоко
6. Довести до кипения
7. Кипятить 5 минут
8. Добавить 20 грамм сливочного масла
9. Выключить плиту, снять с плиты кастрюлю.



**Задание: Найти недостатки в алгоритмах**

## ***2. Алгоритм покраски забора***

1. Покрасить одну доску
2. Переместиться к следующей доске
3. Перейти к действию 1



## ***2.Алгоритм покраски забора***

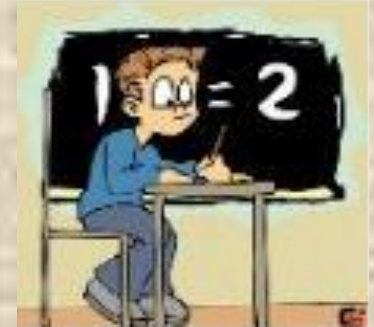
1. Покрасить одну доску.
2. Если есть еще доска, переместиться к следующей доске.
3. Перейти к действию 1.
4. Если доски закончились, завершить работу.





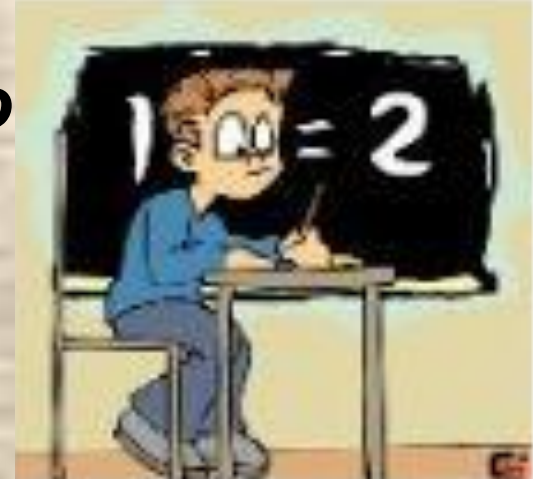
**Задание: Найти недостатки в алгоритмах**

### ***3. Алгоритм нахождения большего из двух данных чисел***



1. Из числа  $A$  вычесть число  $B$ .
2. Если получилось отрицательное значение, то сообщить, что число  $B$  больше.
3. Если получилось положительное значение, то сообщить, что число  $A$  больше

## *Алгоритм нахождения большего из двух данных чисел.*



- Из числа  $A$  вычесть число  $B$ .
- Если получилось отрицательное значение, то сообщить, что число  $B$  больше.
- Если получилось положительное значение, то сообщить, что число  $A$  больше
- Если получился ноль, сообщить, что числа равны

## Задание: Найти недостатки в алгоритмах.

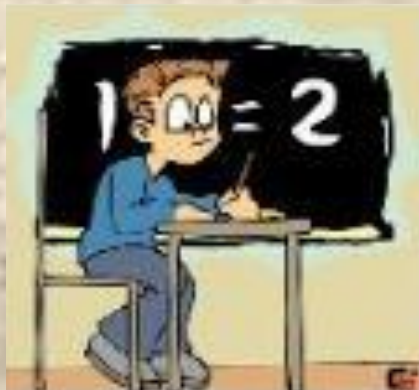
1. Возьмите длинную колотину.
2. Насадите на конец колотины борика.
3. Привяжите колотину к дереву.
4. Ждите, пока не услышите рёв репалой борельки. Бегите к дереву и запозите репалой борельке буркалы.
5. Кладите борельку в кустицу и несите домой.



# Типы алгоритмов



**Линейные алгоритмы** – это алгоритмы, в которых все действия выполняются последовательно одно за другим и только один раз.



**Разветвляющиеся алгоритмы** – это алгоритмы, в которых последовательность зависит от каких-либо условий.



**Циклические алгоритмы** – это алгоритмы, в которых подразумевается, что действия повторяются определенное число раз или до наступления какого-либо события.

**Тип алгоритма**

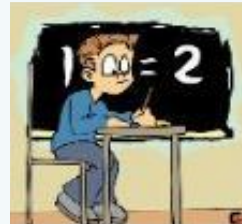
**Структура  
алгоритма**

**Линейный алгоритм**



**Следование**

**Разветвляющийся  
алгоритм**



**Ветвление**

**Циклический алгоритм**

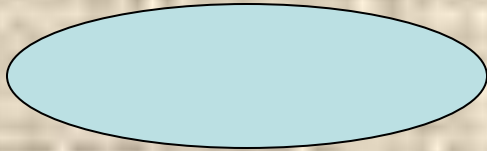


**Цикл**

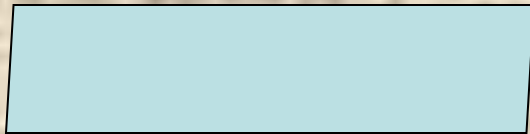
# Способы записи алгоритмов

- **Словесное описание**
- **Таблицы**
- **Формулы**
- **Блок – схемы**
- **Языки программирования**

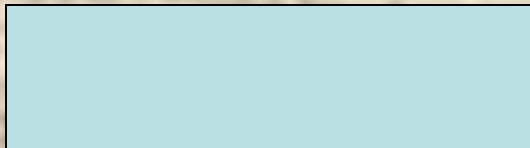
# Блок-схемы



**Начало / конец алгоритма**



**Ввод / вывод данных**



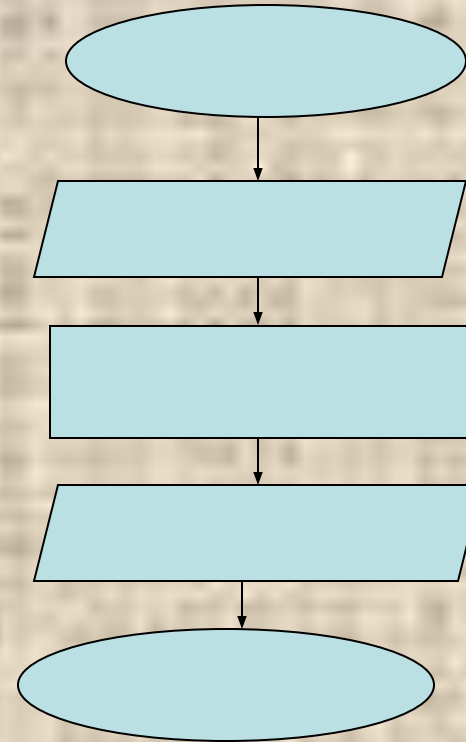
**Действие, которое не требует  
принятия решения от исполнителя**



**Действие, которое требует  
принятия решения от исполнителя**

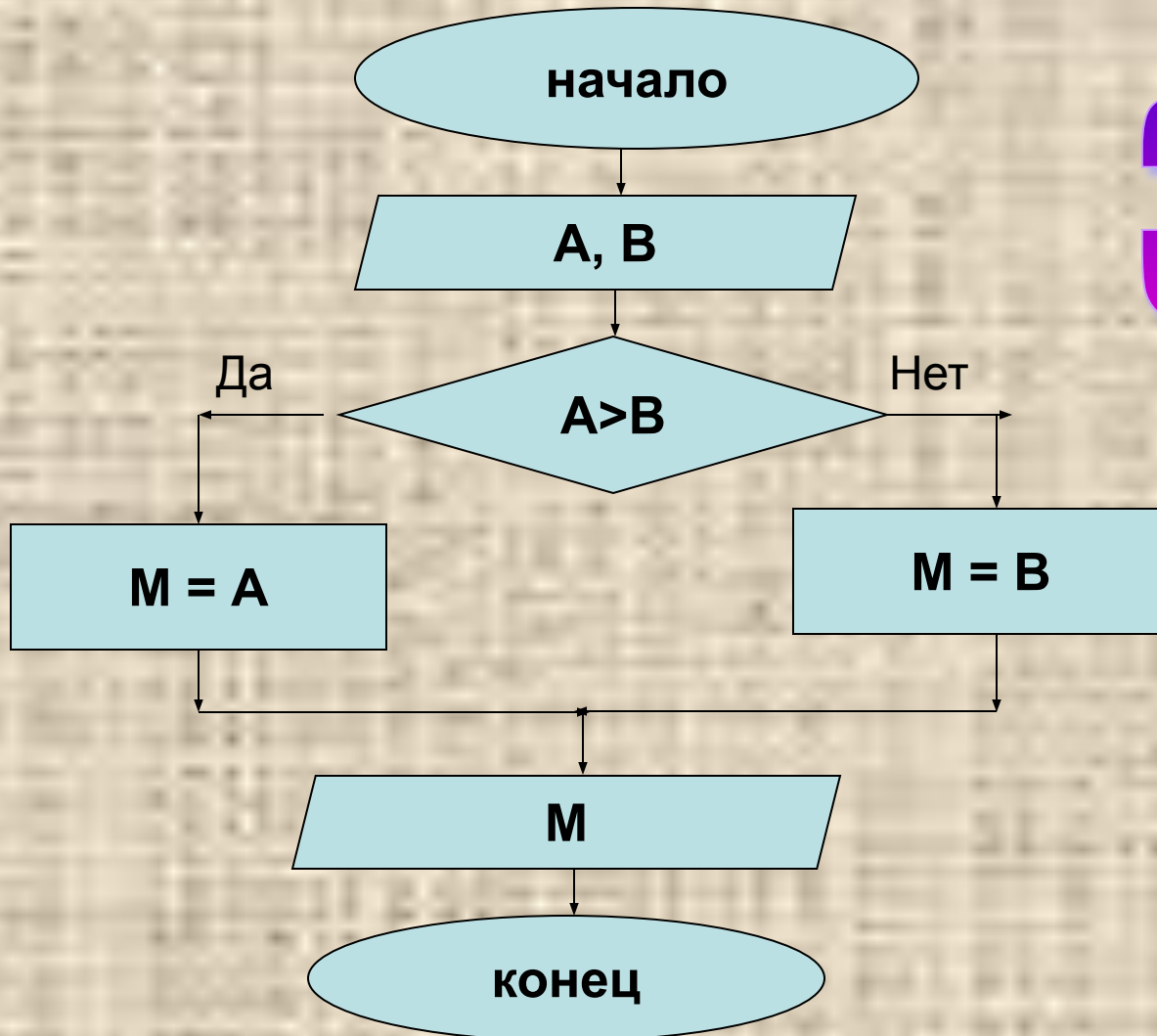
# Следование (Линейная структура)

**Следование** — это такая структура алгоритма, в которой все операции выстроены последовательно друг за другом в единую цепь.





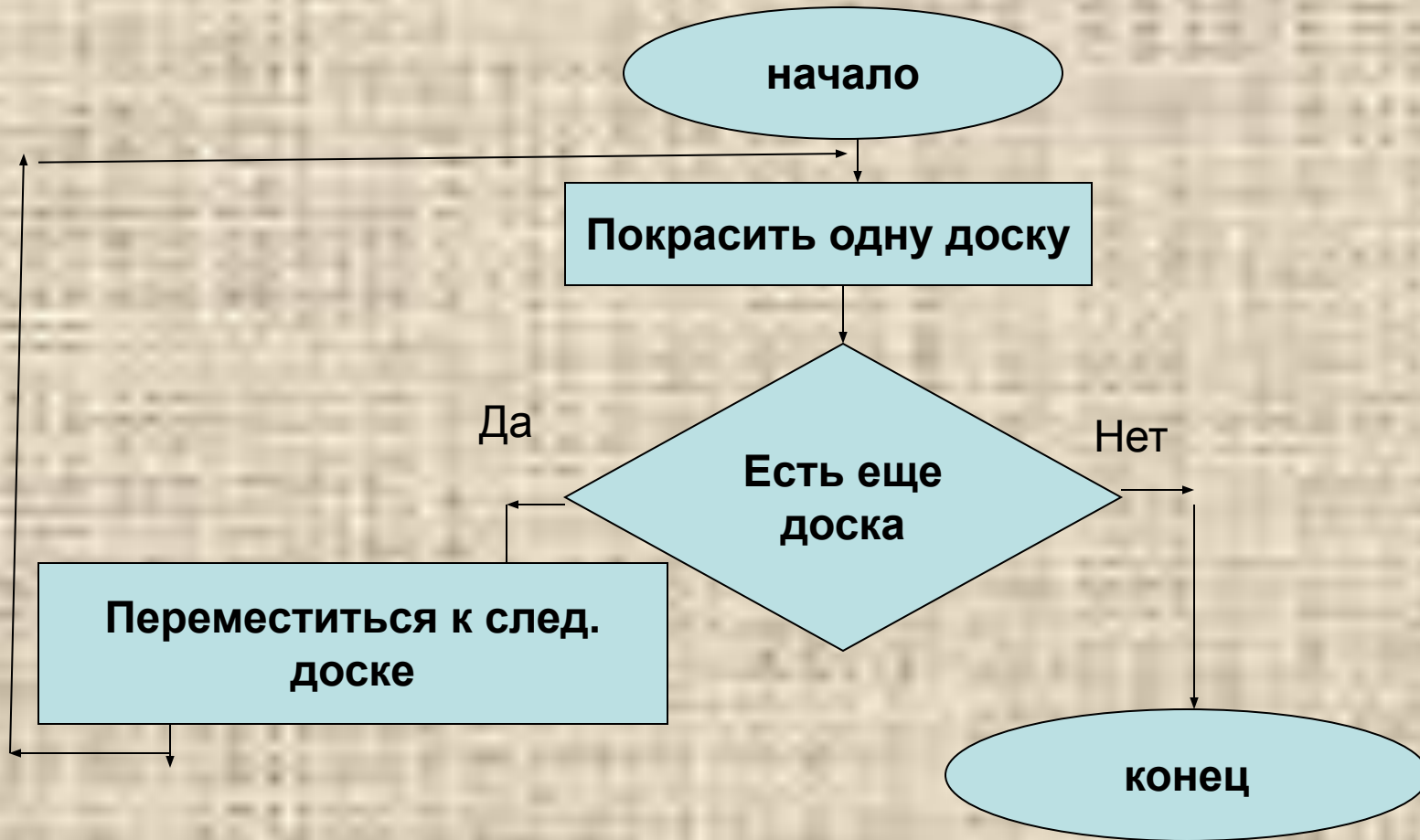
# Алгоритм с ветвлением «Найти наибольшее из двух заданных произвольно чисел»



34 > 23

7 < 9

# Алгоритм покраски забора с циклом



## Итог урока:

- С какими типами алгоритмов вы познакомились?
- С какими алгоритмическими структурами вы познакомились?
- Какие фигуры блок-схемы используются для записи алгоритмов?



## Домашнее задание:

- Задача: В полосатой корзине лежат шары черного и белого цвета. Их достают по одному и кладут в черную или белую корзину в зависимости от цвета. Составить алгоритм сортировки шаров

- п.30,31



Спасибо за урок!

