

# Алгоритм и его формальное исполнение

Алгоритм - это строго определенная последовательность действий при решении задачи.

Алгоритм содержит несколько шагов.

Шаг алгоритма - это каждое отдельное действие алгоритма.



«Алгоритм – это  
порядок  
действий».

**Исполнитель** - это объект выполняющий определенный набор действий.

*Исполнителем может быть человек, робот, животное, компьютер.*

**Система команд исполнителя (СКИ)** - это совокупность команд, которые может выполнять исполнитель.

**Среда исполнителя** - обстановка, в которой функционирует исполнитель.



# РАЗРАБОТКА И ИСПОЛНЕНИЕ

- ⦿ Разрабатывает алгоритмы: **человек**
- ⦿ **Исполняют** алгоритмы: **люди и устройства** – компьютеры, роботы, станки, спутники, сложная бытовая техника, детские игрушки.
- ⦿ *Исполнитель решает задачу по заданному алгоритму, строго следуя по предписаниям (программе) не вникая и не рассуждая, почему он так делает.*



# КТО МОЖЕТ БЫТЬ ИСПОЛНИТЕЛЕМ АЛГОРИТМА?

- **!!!** *Исполнителем алгоритма* может быть не только **человек**, но и **автоматическое устройство** (реальное или воображаемое).
- В этом случае **шаги** алгоритма часто называют **командами** и вводят их в устройство в той форме, в которой оно сможет их обрабатывать.



# Задание: Назови исполнителей следующих видов работы:

Уборка мусора во дворе

Обучение детей в  
школе

Вождение автомобиля

Ответ у доски

Приготовление пищи

Печатание документа  
на принтере

**Результативность** – получение результата за конечное количество шагов

**Понятность** – алгоритм состоит из команд, входящих в систему команд исполнителя.

**Дискретность** –  
Переход к следующему действию возможен только после выполнения предыдущего

**Массовость** – использование алгоритма для решения однотипных задач

Свойства  
АЛГОРИТМА

**Определенность** – означает, что в алгоритме нет команд, смысл которых может быть истолкован исполнителем неоднозначно.

# КЛАССИФИКАЦИЯ АЛГОРИТМОВ ПО ФОРМЕ ПРЕДСТАВЛЕНИЯ:

Словесные

Графическ

ие (блок-

схемы)

Программн

ые



# ЗАДАНИЕ: СОСТАВЬ АЛГОРИТМ СБОРА ПОРТФЕЛЯ. ПРОДУМАЙ СКИ.

1. Возьми портфель
2. Открой дневник
3. Посмотри расписание
4. Сложи школьные принадлежности в портфель
5. Закрой портфель



Способ описания алгоритма Словесный

Число шагов 5 (пять)

Исполнитель Человек

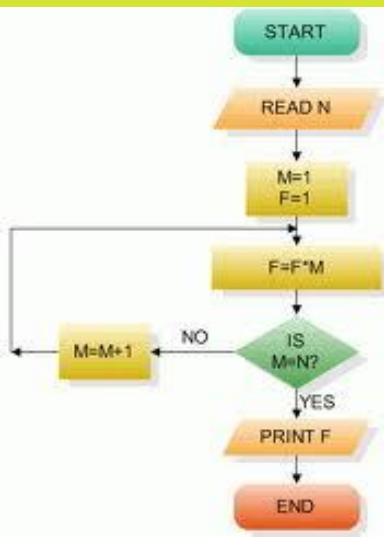
Среда исполнителя Квартира

# Средства представления и




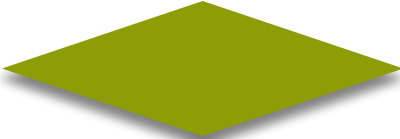

## записи алгоритмов

*Блок-схема* *графическое*

*представление* алгоритма в виде последовательности связанных между собой функциональных блоков (*стандартных графических элементов*), каждый из которых соответствует выполнению одного или нескольких действий.



# Основные условные обозначения в блок-схемах

Условное обозначение	Назначение блока
	<b>Начало или конец алгоритма</b>
	<b>Ввод или вывод данных.</b> Внутри блока перечисляются данные через запятую.
	<b>Процесс.</b> Внутри блока записываются матем. формулы и операции для обработки данных.
	<b>Проверка условия.</b> Внутри блока записываются логические условия. Имеет два выхода Да(+) и Нет(-).
	<b>Направление.</b>

# Классификация алгоритмов по структуре:

- Линейный (следование)
- Разветвленный (ветвление, выбор, альтернатива)
- Циклический (повтор)
- Вспомогательный
- Комбинированный

# ЛИНЕЙНЫЙ АЛГОРИТМ

Линейный алгоритм - это алгоритм, шаги которого выполняются последовательно друг за другом.

(Пример: алгоритм сбора портфеля).

## *Задача*

Вычислить периметр произвольного треугольника по его трем сторонам.

**Решение:**

**1 этап: Постановка задачи.**

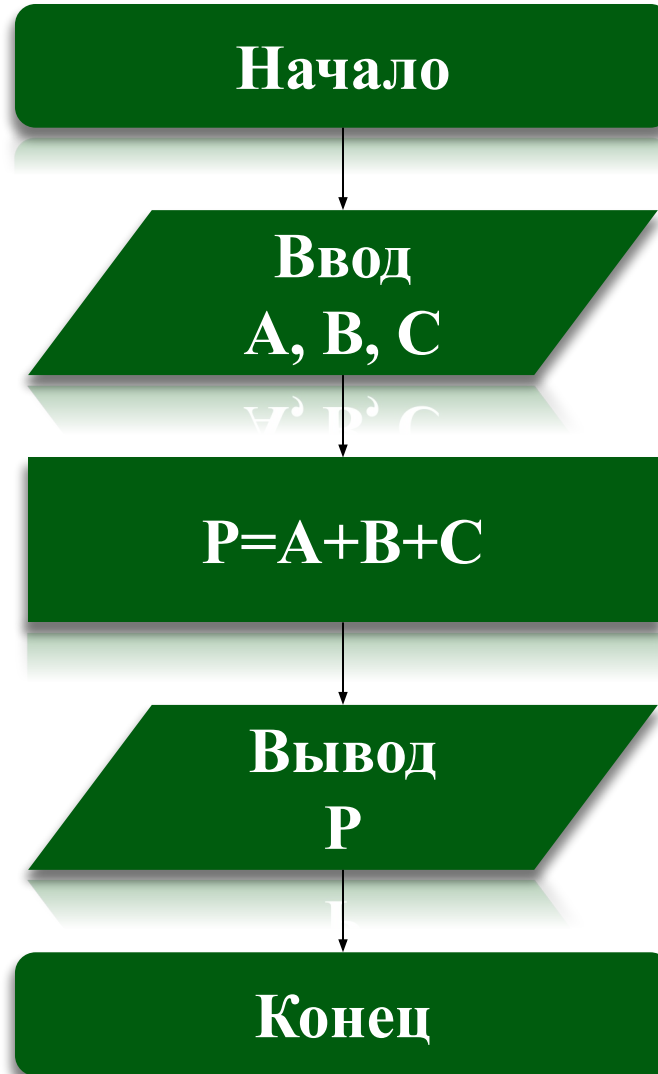
Исходные данные:  $A$ ,  $B$ ,  $C$  – стороны произвольного треугольника

Выходные данные:  $P$  – периметр треугольника.

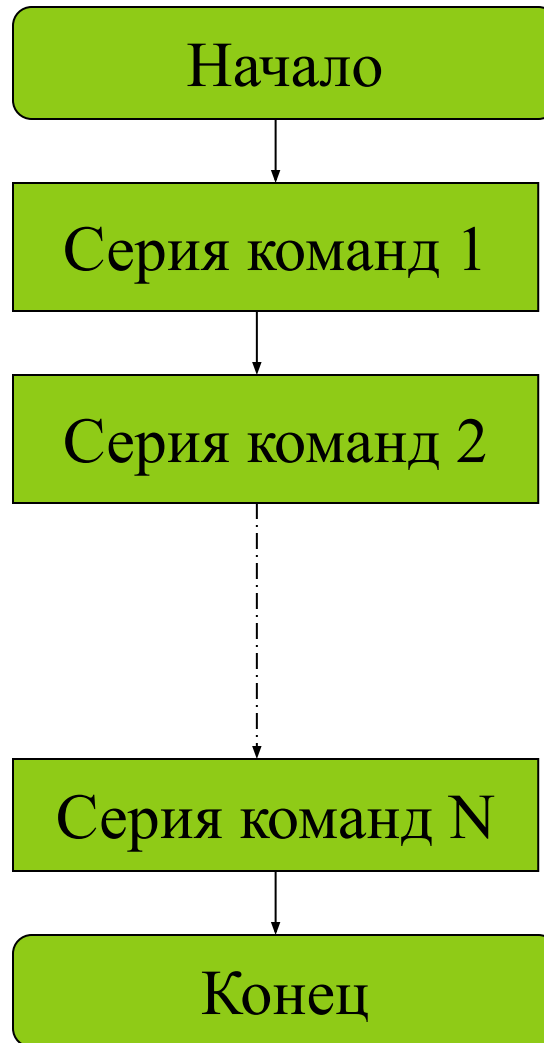
**2 этап: Математическая модель.**

$$P=A+B+C$$

### 3 этап: Составление алгоритма



## Базовая структура линейного алгоритма:





## № 1

Используя блок-схему алгоритма, вычислите значение функции  $Y$  при  $X=2$ ,

### РЕШЕНИЕ:

начало

1.  $X = 2$
2.  $Z = 8 * 2 = 16$
3.  $Z = \sqrt{16} = 4$
4.  $Z = 4 - 1 = 3$
5.  $Y = 3 * 2 = 6$
6.  $Y = 6 / 3 = 2$

конец

