

# Алгоритм. Свойства. Способы записи.

Автор: Э.О. Качанова,  
учитель информатики



# Понятие алгоритма

- Алгоритм – это строгая и четкая последовательность действий, выполнение которых приводит к определенному результату.
- **Требования к алгоритмам**
  - 1) Ориентированность на конкретного исполнителя.
  - 2) Понятность для исполнителя (алгоритм составляется в соответствии с системой команд исполнителя).
  - 3) Точность (каждая команда должна определять однозначное действие исполнителя).
  - 4) Конечность (наличие конца алгоритма через конечное число шагов).
  - 5) Результативность (получение нужного результата по окончании алгоритма).
  - 6) Массовость (применимость для широкого класса задач).
  - 7) Формальность исполнения (во время исполнения алгоритма исполнитель не должен задумываться над сутью выполняемых действий).



# Свойства алгоритма

- **дискретность**: состоит из отдельных шагов (команд)
- **понятность**: должен включать только команды, известные исполнителю (входящие в СКИ)
- **определенность**: при одинаковых исходных данных всегда выдает один и тот же результат
- **конечность**: заканчивается за конечное число шагов
- **массовость**: может применяться многократно при различных исходных данных
- **корректность**: дает верное решение при любых допустимых исходных данных

## Свойства алгоритма

**Дискретность**

Путь решения задачи разделён на отдельные шаги

**Понятность**

Алгоритм состоит из команд, входящих в СКИ

**Определённость**

Команды понимаются однозначно

**Результативность**

Обеспечивается получение ожидаемого результата

**Массовость**

Обеспечивается решение задач с различными исходными данными

# Виды алгоритмических конструкций

**Линейный алгоритм** – алгоритм, в котором команды исполняются последовательно, одна за другой.

**Разветвляющийся алгоритм** – алгоритм, в котором после проверки условия в разных ситуациях исполняются разные наборы команд.

**Циклический алгоритм** – алгоритм, в котором содержится команда повторения, т.е. цикл.



# Способы записи алгоритмов

- Словесный
- Запись на алгоритмическом языке
- Блок-схема (Графическое представление алгоритма)
- Программа (запись алгоритма на языке программирования)



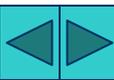
# Словесный способ записи

(описание алгоритма с помощью слов русского языка).

Пример. Алгоритм включения компьютера.

- Подойти к компьютеру.
- Включить монитор.
- Включить системный блок.

Задание: придумайте свой пример. Запишите



# Запись на алгоритмическом языке

Пример. Алгоритм нахождения максимального из двух введенных чисел.

- Начало

Ввод числа  $x$

Ввод числа  $y$

Если  $x > y$

То Вывод  $x$

Иначе Вывод  $y$

Все

- Конец

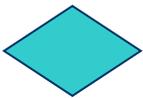


# Блок-схема (графическое представление алгоритма)

- Блок-схемы являются одним из графических способов представления алгоритмов.
- Блок-схема состоит из блоков, соединенных линиями. Чаще всего используются блоки следующих типов:



- выполнение операции;



- выбор направления выполнения алгоритма в зависимости от выполнения условия;



- ввод/вывод данных;



- начало и конец алгоритма.

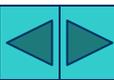


# Программа (запись алгоритма на языке программирования)

Пример. Определение четности введенного числа.

На языке BASIC:

```
INPUT "Введите целое число"; X
A$="четное"
IF X MOD 2<>0 THEN A$="не"+A$
PRINT "Введенное число ", A$
```



Исполнитель это некоторый объект (человек, животное, техническое устройство), способный выполнять некоторый набор команд



**Формальный**

**Неформальный**

**Одну и ту же команду  
всегда выполняет  
одинаково**



Исполнителя

местность, обстановка,  
условия

**СКИ**  
система команд  
исполнителя

Набо  
вь

**Может выполнить  
команду по-разному**

Режим работы

Непосредственное  
управление и программное  
управление



# Система команд исполнителя



Команды, которые может выполнять конкретный исполнитель, образуют **систему команд исполнителя** (СКИ).

Исполнитель **формально** выполняет алгоритмы, составленные из команд, входящих в его СКИ.

# Самое главное



**Алгоритм** - это предназначенное для конкретного исполнителя описание последовательности действий, приводящих от исходных данных к требуемому результату, которое обладает свойствами:

- ✓ **дискретности**
- ✓ **понятности**
- ✓ **определённости**
- ✓ **результативности**
- ✓ **массовости**

# Самое главное



**Исполнитель** - некоторый объект (человек, животное, техническое устройство), способный выполнять определённый набор команд.



**Формальный исполнитель** одну и ту же команду всегда выполняет одинаково. Для каждого формального исполнителя можно указать: круг решаемых задач, среду, систему команд и режим работы.



Способность исполнителя действовать формально обеспечивает **возможность автоматизации деятельности человека.**



# Домашнее задание

§2.1, 2.2;

№6, стр.54, №4, стр.62

(учебник «информатика 8» ФГОС, Л.Л.Босова)



**Спасибо за внимание!**