

Алгоритмы и структуры данных на Python. Интерактивный курс

## Урок 1

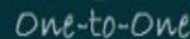


# Введение в алгоритмизацию и реализация алгоритмов на Python

↑  
right  
→



4 12 25 45 74  
8 13 20 29  
5 7 9  
2 2

 $B_4^9$ 

Find  $7 + 12 + 17 + 22 + \dots + 342$ .

$$2S_n = 349 + 349 + 349 + 349 + \dots + 349$$

$$2S_y = 349 \div 68$$

$$S_r = \frac{349 + 68}{2}$$

$$S_n = 11866$$



$$v - e + f = 2$$

$$a_n = a_0 + 4n$$
 $K_{33}$ 

Original:

$$\exists x \forall y (x \geq 2y \rightarrow x > y + 1)$$

CONVERSE:

$$\exists x \forall y (x > y + 1 \rightarrow x \geq 2y)$$

Negation:

$$\neg [\exists x \forall y (\neg (x \geq 2y) \vee x > y + 1)]$$

$$\forall x \exists y (x \geq 2y \wedge x \leq y + 1)$$

Contrapositive:

$$\exists x \forall y (x \leq y + 1 \rightarrow x < 2y)$$

☆☆☆ | ☆☆☆☆☆ | ☆ | ☆☆☆

$$(A \cup B \cup C) \cup (A \cap B \cap C)$$



### DLE Example

# План

- Что такое алгоритм
- Свойства алгоритмов
- Способы представления алгоритмов
- Пример псевдокода



# Основы алгоритмизации



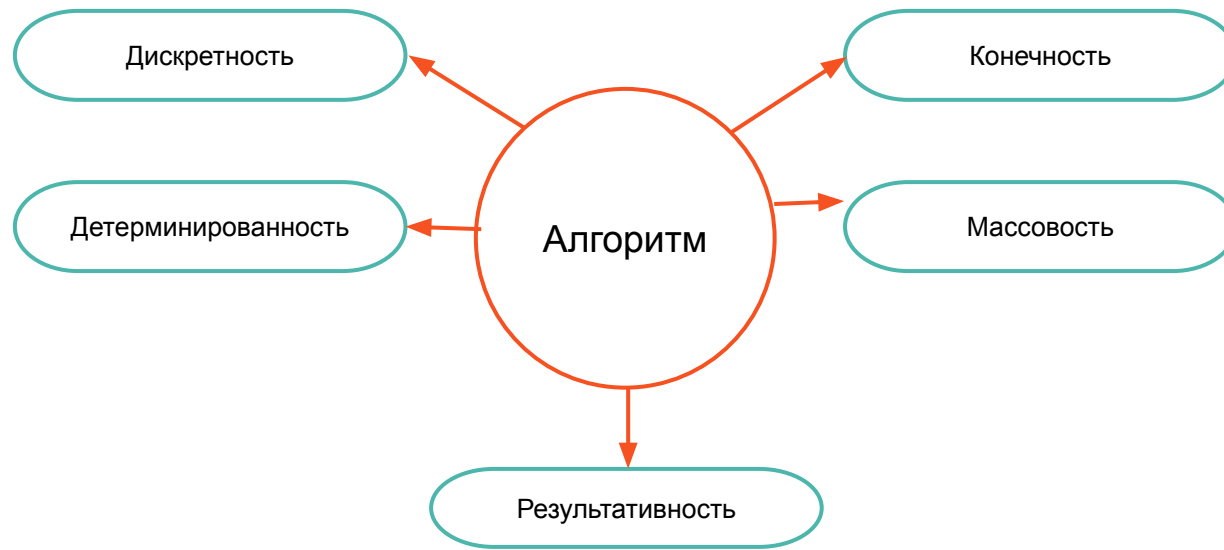
# Понятие алгоритма

совокупность

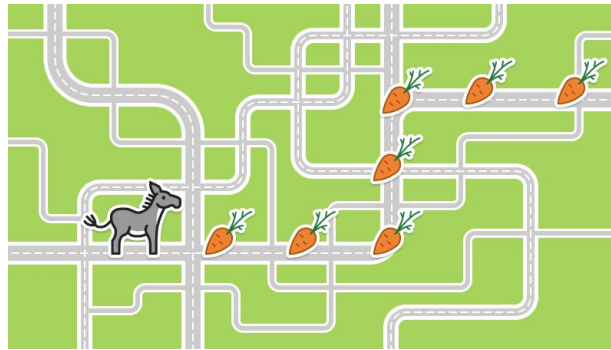
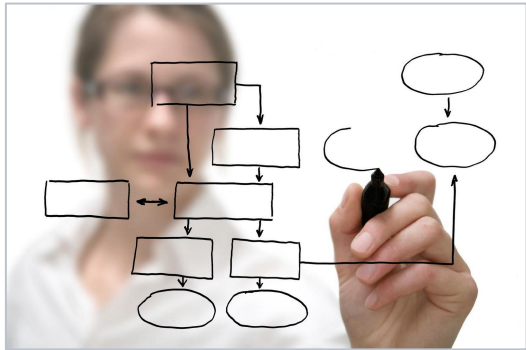
**Алгоритм** – это последовательность чётко определенных действий, выполнение которых ведёт к решению задачи.



# Каким должен быть алгоритм



# Понятие алгоритма



# Способы представления алгоритма

- Словесный
- Псевдокод
- Графический
- Программный





# Решим задачу - найдем частное двух чисел

делимое разделить на делитель равно частное



# Итоги:

## Теория

- Что такое алгоритм
- Свойства алгоритмов
- Способы представления алгоритмов

## Практика

- Пример псевдокода

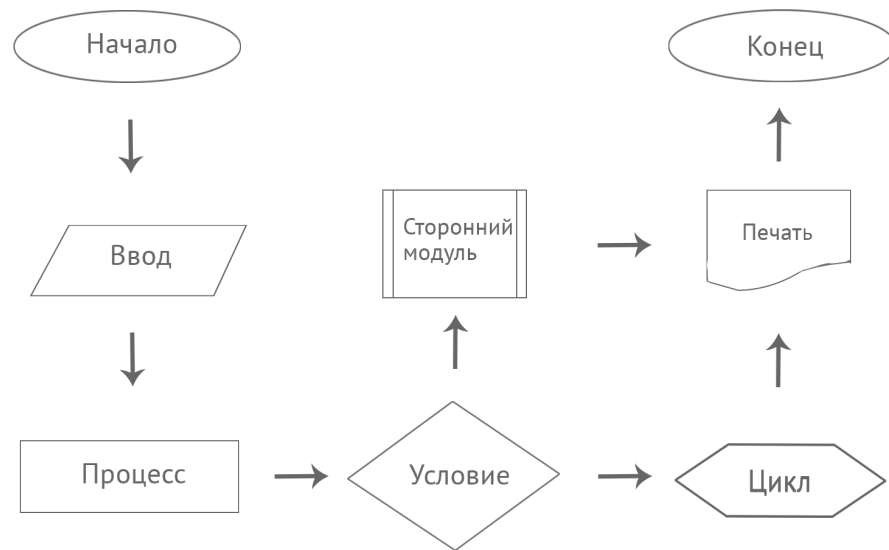


# План

- Способы представления алгоритмов (блок-схемы)
- Виды алгоритмов
- Пример блок-схемы


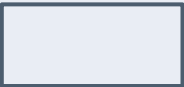
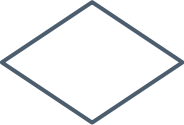


# Блок-схема




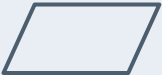


# Блок-схема

Таблица 1.2. Базовые элементы блок-схемы

Название	Обозначение	Функция
Терминатор		Отображает вход из внешней среды или выход из нее (наиболее частое применение – обозначение начала и конца алгоритма)
Процесс		Выполнение одной или нескольких операций, обработка данных любого вида, изменение данных. Внутри фигуры записывают операции над данными.
Решение		Отображает решение, влияющее на дальнейшее выполнение алгоритма. Используется для обозначения альтернативных ветвей выполнения алгоритма.



# Блок-схема

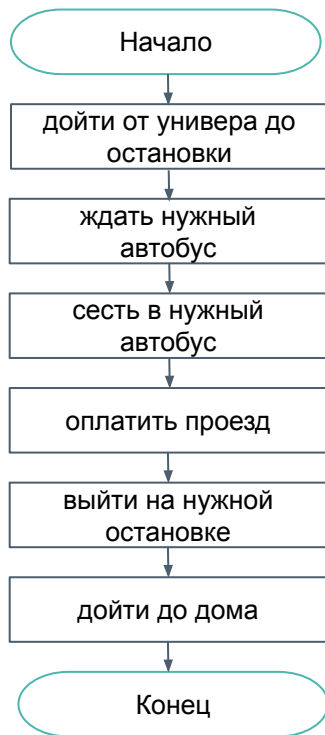
Предопределенные процесс		Символ отображает выполнение процесса, который определен в другом месте программы
Данные		Используется для обозначения операций ввода-вывода.
Соединитель		Символ обозначает выход в часть схемы и вход из другой части этой схемы. Используется для обрыва линии и продолжения ее в другом месте
Комментарий		Используется для более подробного описания шага, процесса или группы процессов



# Виды алгоритмов

- Линейный
- Разветвляющийся
- Циклический

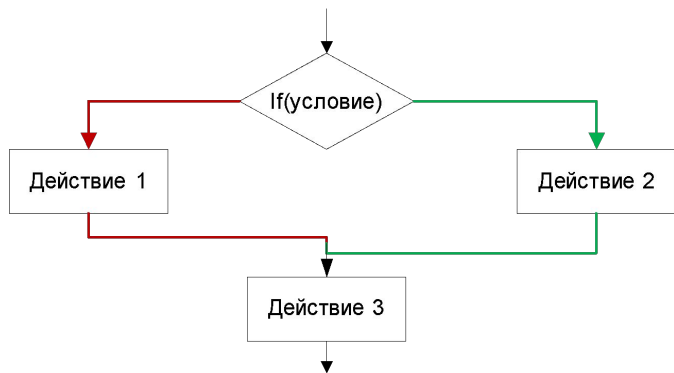




# Линейный алгоритм

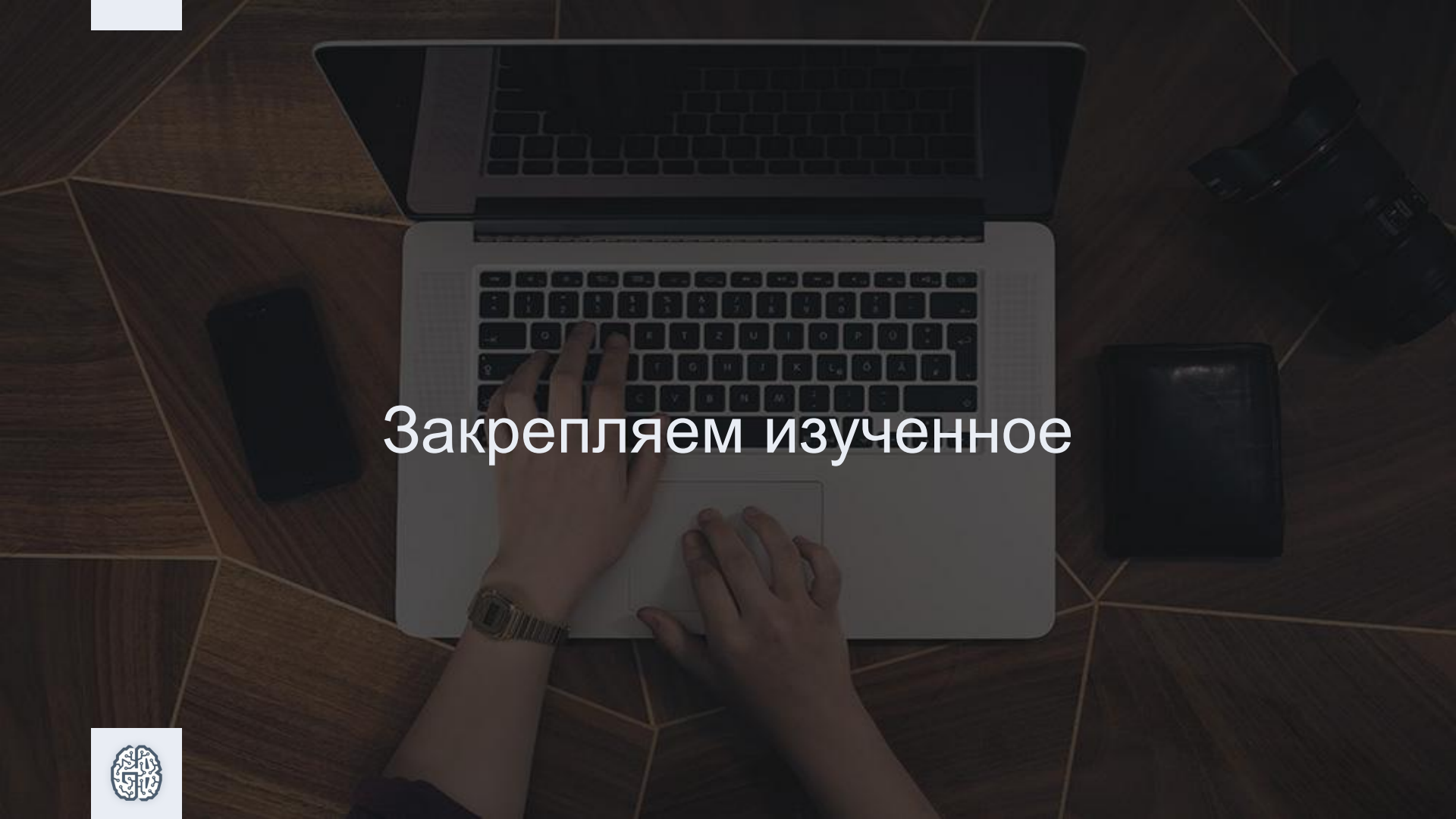






# Разветвляющийся алгоритм





Закрепляем изученное



# Итоги:

## Теория

- Способы представления алгоритмов (блок-схемы)
- Виды алгоритмов

## Практика

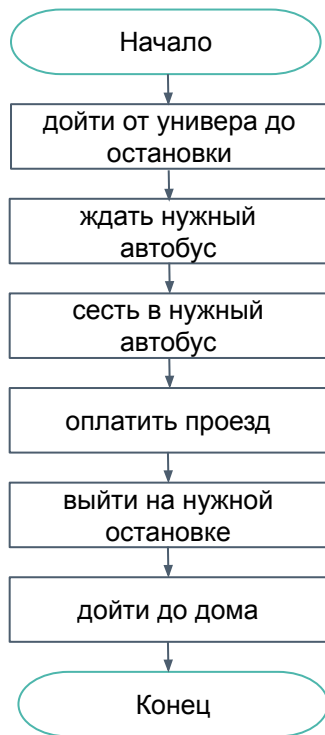
- Пример блок-схемы



# План

- Что такое линейный алгоритм
- Графическое представление линейного алгоритма
- Линейная программа





# Линейный алгоритм



# Закрепляем изученное

- **Задача 1.** Найти сумму и произведение цифр трехзначного числа, которое вводит пользователь.



# Итоги:

## Теория

- Линейный алгоритм - самый простой вид алгоритма
- Сложность линейного алгоритма практически всегда равна  $O(n)^*$

## Практика

- Блок-схема линейного алгоритма
- Код на Python



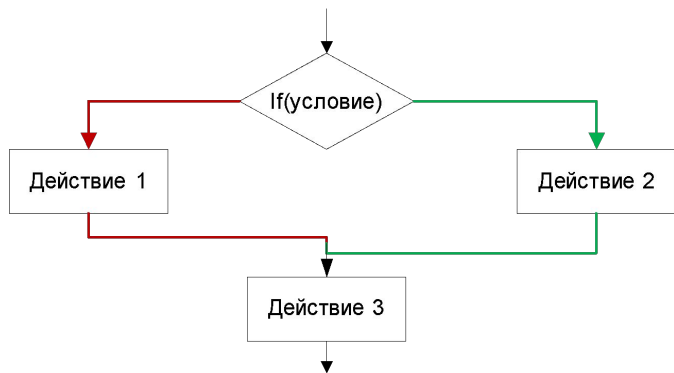
\*Что такое  $O(n)$ ??? Ответ мы узнаем через пару уроков.

# План

- Алгоритм с ветвлением
- Вспомним `if elif else`







# Разветвляющийся алгоритм



# Разветвляющийся алгоритм

```
if на улице идёт снег:  
    надеть шапку  
else:  
    надеть кепку
```



# Разветвляющийся алгоритм

```
if на улице идёт снег:  
    надеть шапку
```



# Разветвляющийся алгоритм

```
if на улице идёт снег:  
    надеть шапку  
    if сильный ветер:  
        надеть шарф
```



# Разветвляющийся алгоритм

```
if на улице идёт снег and сильный ветер:  
    надеть шапку  
    надеть шарф
```



# Закрепляем изученное

- **Задача.** Вычислить значение функции  $y = f(x)$

$$f(x) = \begin{cases} 2x-10, & x > 0 \\ 0, & x = 0 \\ 2|x| - 1, & x < 0 \end{cases}$$



# Итоги:

## Теория

- Если программа должны принимать решения, делать выбор - используем ветвящийся алгоритм

## Практика

- Блок-схема ветвящегося алгоритма
- Код на Python



# План

- Алгоритм с ветвлением
- Одна задача - разные алгоритмы
- Один алгоритм - одно решение





# Закрепляем изученное

- **Задача.** Сравниваем числа и находим максимум.

Пояснение: Из трёх целых чисел найти наибольшее



# Итоги:

## Теория

- Для решения одной задачи существуют разные алгоритмы
- Вывод: выбирайте самый эффективный алгоритм из возможных

## Практика

- Блок-схема ветвящегося алгоритма
- Код на Python



# План

- Практическое занятие
- Домашнее задание



# Закрепляем изученное

- **Задача.** Перевести байты в килобайты или наоборот



# Домашние задания

- Для каждого упражнения составьте графическое представление алгоритма и напишите программный код
- Сохраняйте каждую задачу (код на Python) в отдельный файл
- В начале файла вставьте текст задачи в виде комментария
- Графическое представление (блок-схему) присылайте в удобном для просмотра виде (jpg, pdf и т.п.)



# Домашнее задание

1. Выполнить логические побитовые операции "И", "ИЛИ" и др. над числами 5 и 6. Выполнить над числом 5 побитовый сдвиг вправо и влево на два знака.
2. По введенным пользователем координатам двух точек вывести уравнение прямой, которая проходит через эти точки.



# Домашнее задание

3. Написать программу, которая генерирует в указанном пользователем диапазоне:
  - a. случайное целое число
  - b. случайное вещественное число
  - c. случайный символ

Для каждого из трех случаев пользователь задает свои границы диапазона. Если надо получить случайный символ от 'a' до 'f', вводятся эти символы. Программа должна вывести на экран любой символ алфавита от 'a' до 'f' включительно.



# Домашнее задание

4. Пользователь вводит две буквы. Определить их порядковый номер в алфавите и рассчитать число букв между ними.
5. Пользователь вводит номер буквы в алфавите. Определить, какая это буква.





# Домашнее задание

6. По введенным пользователем длинам трех отрезков определить, можно ли составить из этих отрезков треугольник. Если да, определить, будет ли треугольник разносторонним, равнобедренным или равносторонним.
7. Определить, является введенный пользователем год високосным или нет.
8. Вводятся три разных числа. Найти, какое из них является средним (больше одного, но меньше другого).



# Итоги:

## Теория

- Требования к домашнему заданию

## Практика

- Блок-схема
- Код на Python



# План

- Разбор домашнего задания



# Домашнее задание

Выполнить логические побитовые операции "И", "ИЛИ" и др. над числами 5 и 6. Выполнить над числом 5 побитовый сдвиг вправо и влево на два знака.



# Домашнее задание

По введенным пользователем координатам двух точек вывести уравнение прямой, которая проходит через эти точки.



# Домашнее задание

Написать программу, которая генерирует в указанном пользователем диапазоне:

- а. случайное целое число
- б. случайное вещественное число
- с. случайный символ

Для каждого из трех случаев пользователь задает свои границы диапазона. Если надо получить случайный символ от 'a' до 'f', вводятся эти символы. Программа должна вывести на экран любой символ алфавита от 'a' до 'f' включительно.



# Домашнее задание

Пользователь вводит две буквы. Определить их порядковый номер в алфавите и рассчитать число букв между ними.



# Домашнее задание

Пользователь вводит номер буквы в алфавите. Определить, какая это буква.





# Домашнее задание

По введенным пользователем длинам трех отрезков определить, можно ли составить из этих отрезков треугольник. Если да, определить, будет ли треугольник разносторонним, равнобедренным или равносторонним.



# Домашнее задание

Определить, является введенный пользователем год високосным или нет.



# Домашнее задание

Вводятся три разных числа. Найти, какое из них является средним (больше одного, но меньше другого).

