

Algorithms and data structures

lecture #1 Introduction, simple examples.

Mentor: Rustam Khakov

lecture #1 Introduction, simple examples.

- What is algorithm, introduction
- Characteristics of an algorithm
- Properties of an algorithm
- Type of an algorithm
- How to design an algorithm
- Example
 - add three numbers and print the sum
- Why analysis is important
 - Algorithmic analysis
 - Types of algorithmic analysis



Тимлид в компании, предоставляющий резервное копирование данных.

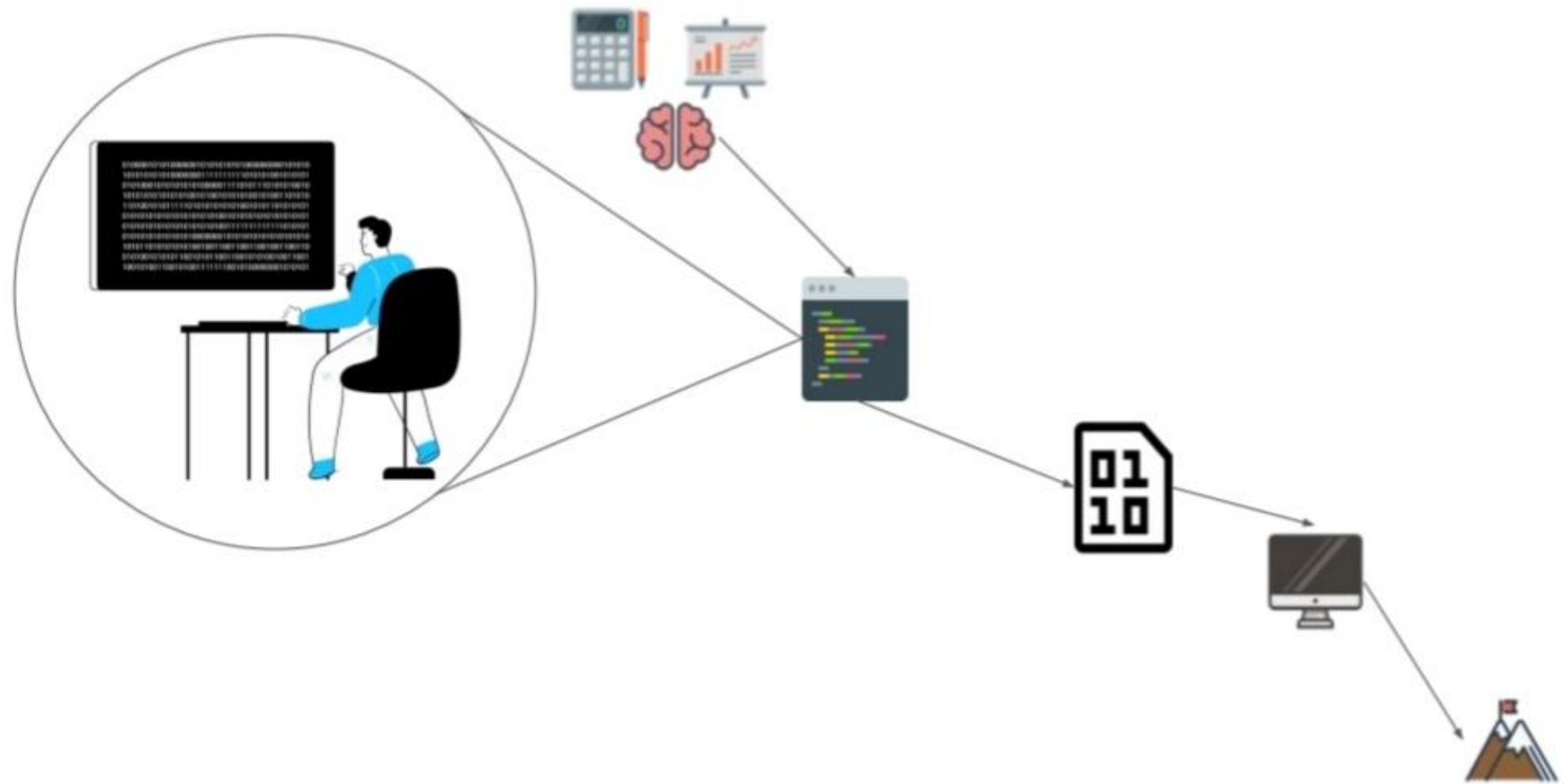
Учился в Казани и в Германии.

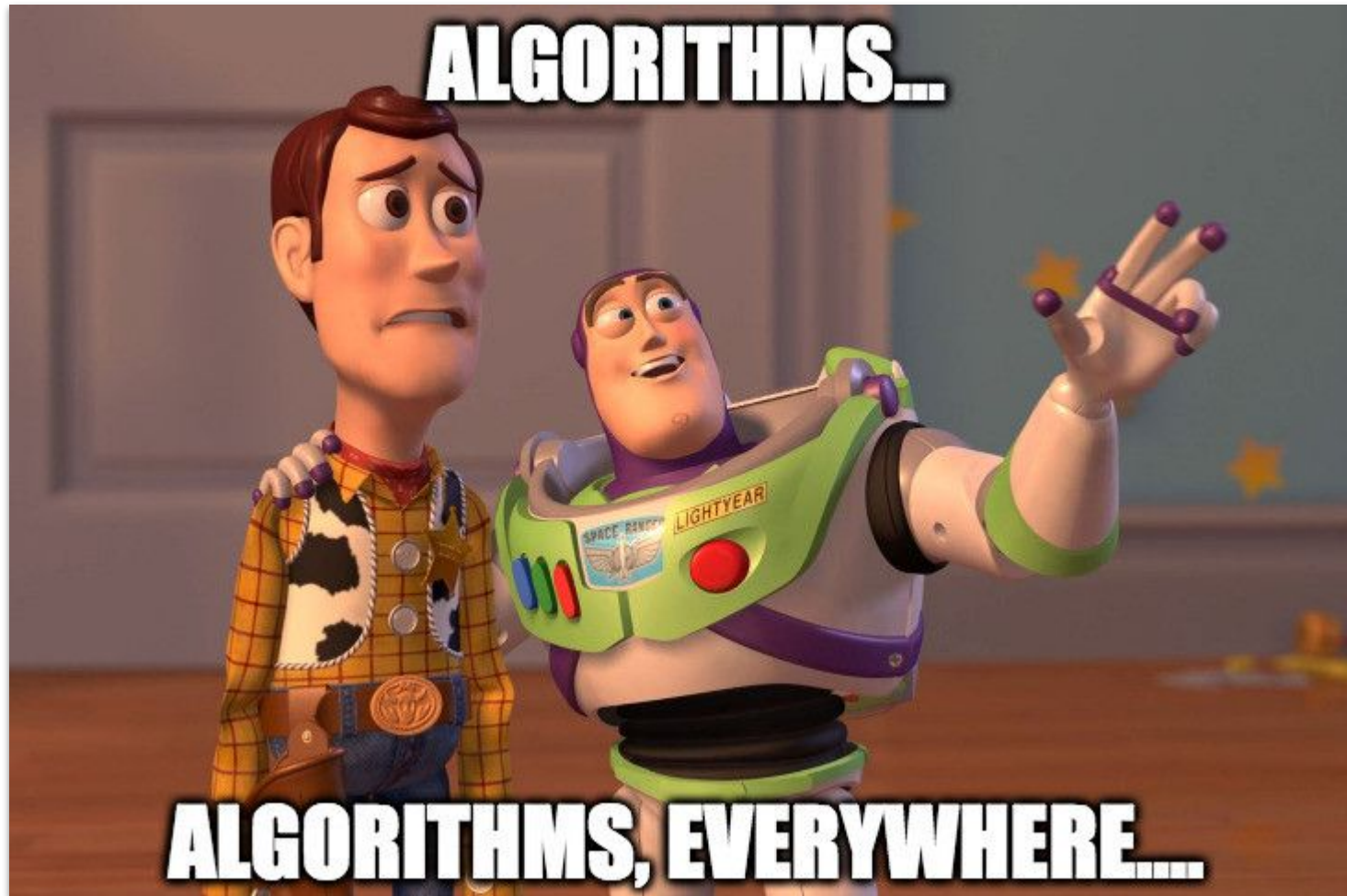
- изучал различные языки программирования и технологии
- были проекты с дронами
- прошел путь от Java Junior до Team Lead
- играю на гитаре
- увлекаюсь бегом

What is algorithm, introduction

- Алгоритм - совокупность точно заданных правил решения некоторого класса задач или набор инструкций, описывающих порядок действий исполнителя для решения определённой задачи. (wikipedia)
- Алгоритм - последовательность конечных шагов для решения конкретной проблемы (могут быть как простыми, так и сложными).

Алгоритмизация – процесс разработки алгоритма для решения какой-либо задачи





Algorithm's characteristics

- Ясный и недвусмысленный
- Четко определенные входные данные
- Четко определенные результаты
- Конечность
- Выполнимый
- Независимый от языка

Properties of an algorithm

Алгоритм должен:

- Завершиться через конечное время.
- Производить хотя бы один вывод.
- Принимать ноль или более входных данных.
- Консистентный - давать один и тот же результат для одного и того же входного случая.
- В идеале каждый шаг в алгоритме должен быть эффективным.

Types of an algorithm

1. Алгоритм грубой силы.
2. Рекурсивный алгоритм.
3. Алгоритм поиска с возвратом.
4. Алгоритм поиска.
5. Алгоритм сортировки.
6. Алгоритм хеширования.
7. Алгоритм «разделяй и властвуй».
8. Жадный алгоритм.
9. Алгоритм динамического программирования.
10. Рандомизированный алгоритм.

How to design an algorithm

1. Четкое определение задачи.
2. Учитывать ограничения.
3. Входные данные, которые необходимо принять для решения задачи.
4. Ожидаемый результат после решения задачи.
5. Решение задачи находится в рамках заданных ограничений.

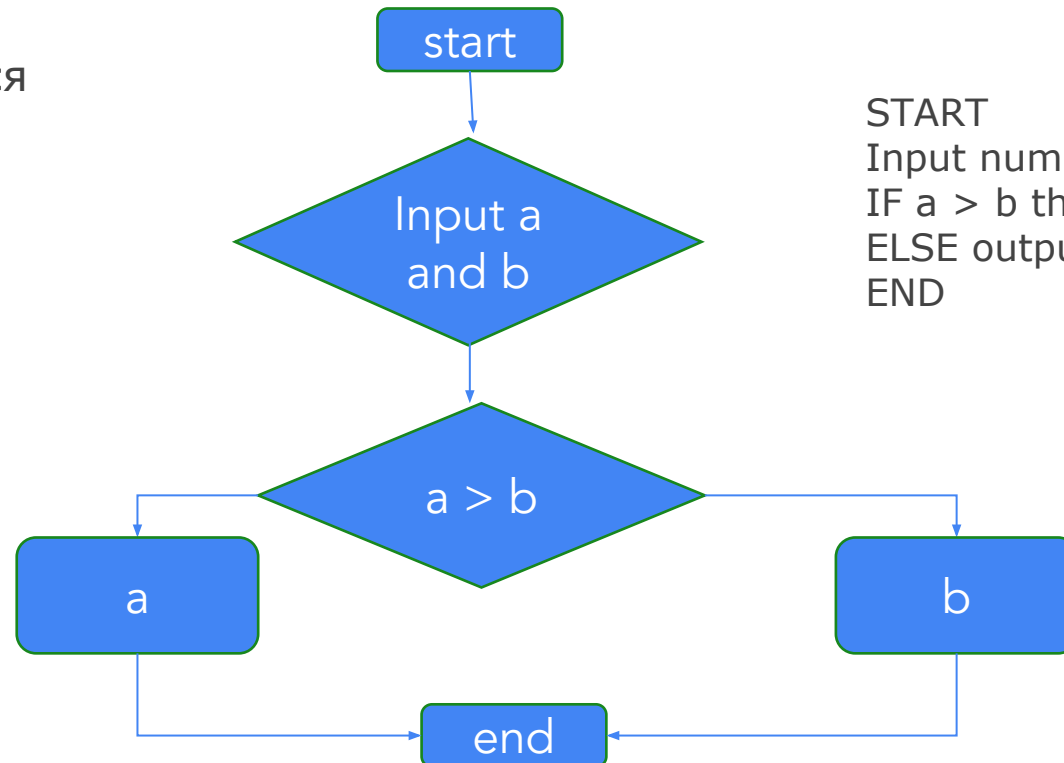
How to design an algorithm

Существует три основных способа описания алгоритма:

- Текстовой – расписываете шаги алгоритма последовательно в тексте
- Алгоритмический язык – псевдокод
- Графический способ – изображается графическим в виде блок-схем.

Пример

START
Input number: a, b
IF a > b then conclusion a
ELSE output b
END



What is effective algorithm?

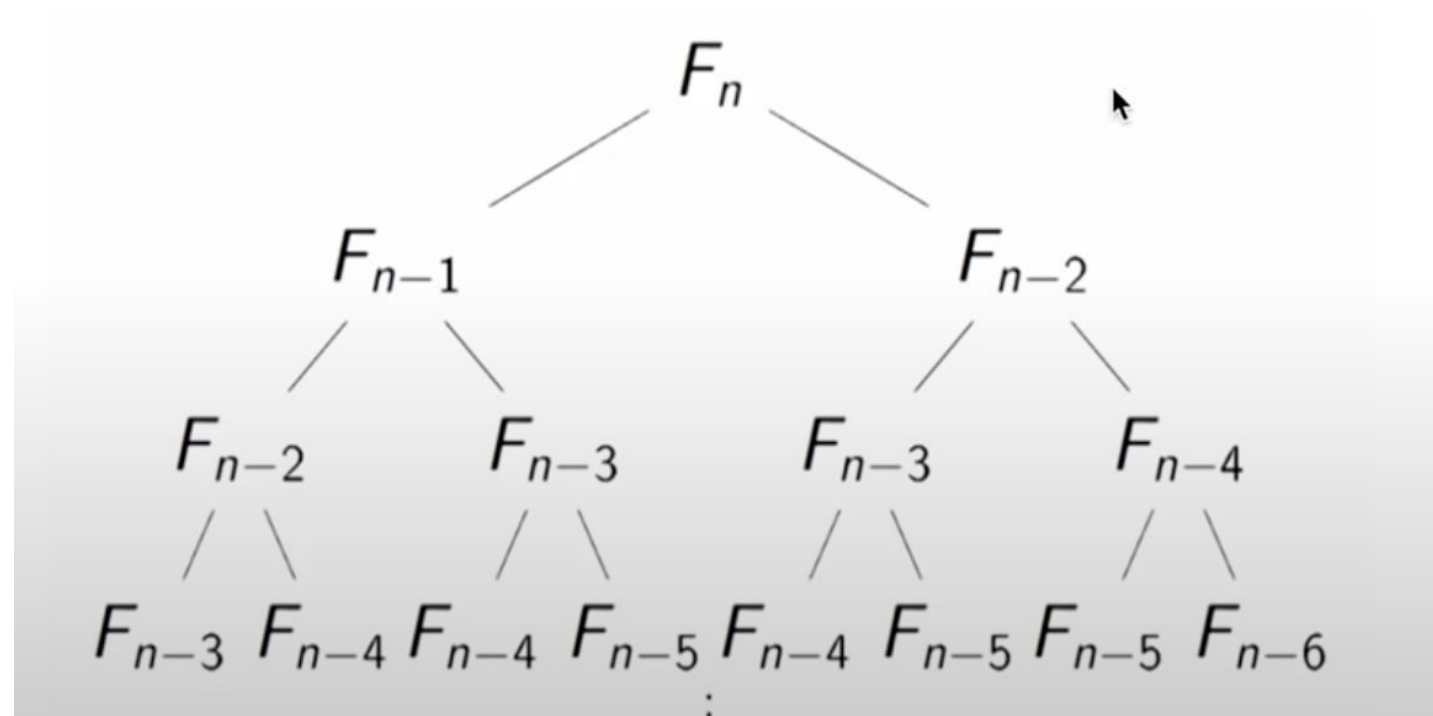
Фактор времени : время измеряется путем подсчета количества ключевых операций.

Фактор пространства : пространство измеряется путем подсчета максимального объема памяти, требуемого алгоритмом.

Space Complexity

Time Complexity

$$F_n = \begin{cases} 0, & n = 0 \\ 1, & n = 1 \\ F_{n-1} + F_{n-2}, & n > 1 \end{cases}$$



Преимущества:

- легко понять
- пошаговое представление решения поставленной задачи.

В алгоритме проблема разбивается на более мелкие части или шаги, поэтому программисту легче преобразовать ее в настоящую программу.

Недостатки:

- Написание алгоритма занимает много времени.
- Проблемы с пониманием сложной логики
- Операторы ветвления и цикла трудно показать в алгоритме.