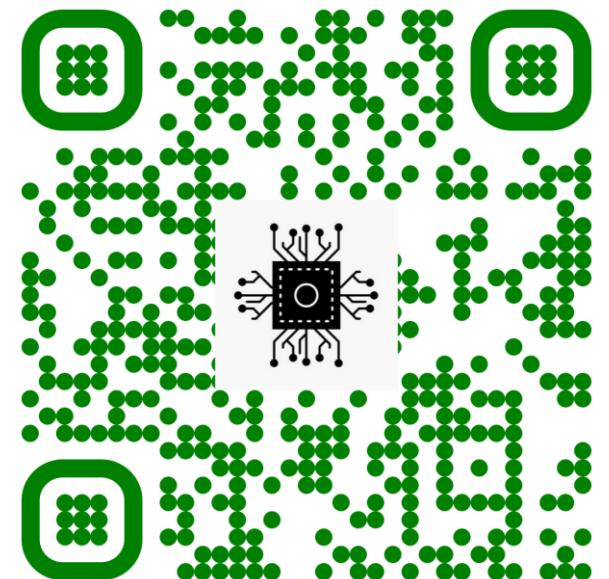


Алгоритмы и структуры данных

Деев Богдан Юльевич
Почта: deevbogdanyi@yandex.ru
Телеграм: @BogdanDeev



Программа курса. 1 семестр

Свойства и особенности алгоритмов

Базовые структуры данных

Сортировки

Деревья поиска

Хеш-таблицы

Программа курса. 2 семестр

Обходы графа

Кратчайшие пути во взвешенном графе

Жадные алгоритмы

Динамическое программирование

Остовные деревья

Потоки в сетях

RMQ, LCA, Sparse-table

Программа курса. 2 семестр

10 лекций

4 конкурса

Алгоритмы. Зачем?

Решать

Объяснять

Доказывать правильность

Объяснять эффективность

Алгоритмы. Зачем?



A
—

kaspersky

Яндекс



SBER BANK



OZON



Оценка сложности алгоритма. Как считать?



Λ √ ?



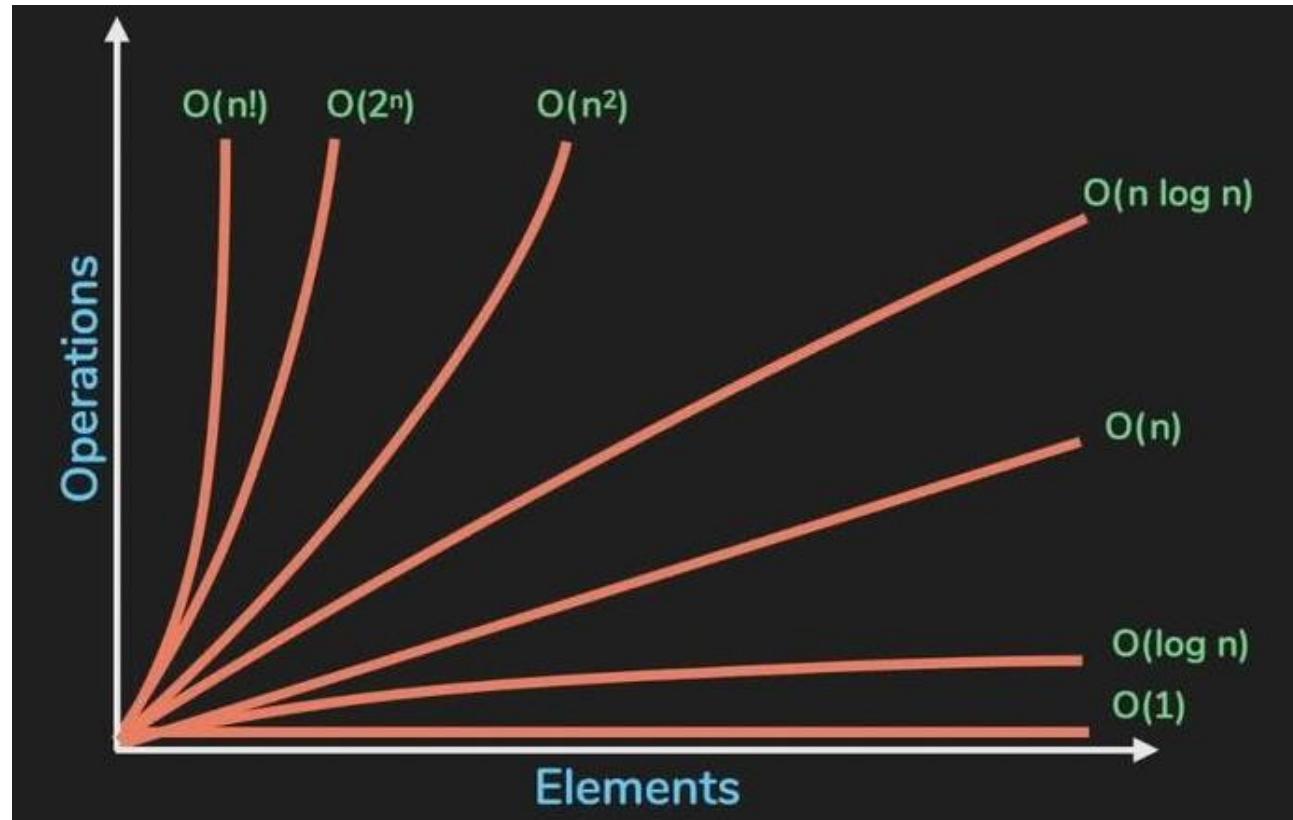
Время – не подходит

Оценка сложности алгоритма. Что подходит?

Фиксированные операции компьютера!

Ответ на вопрос: «Сколько нужно сделать таких операций?» - Асимптотический анализ сложности алгоритма.

Асимптотический анализ



input	constant	logarithmic	linear	log-linear	quadratic	polynomial	exponential
n	$\Theta(1)$	$\Theta(\log n)$	$\Theta(n)$	$\Theta(n \log n)$	$\Theta(n^2)$	$\Theta(n^c)$	$2^{\Theta(n^c)}$
1000	1	≈ 10	1000	$\approx 10,000$	1,000,000	1000^c	$2^{1000} \approx 10^{301}$
Time	1 ns	10 ns	$1\text{ }\mu\text{s}$	$10\text{ }\mu\text{s}$	1 ms	10^{3c-9} s	10^{281} millenia

Как решить алгоритмическую задачу?

Упростить задачу до существующего решения

Разработать свой алгоритм

Упростить задачу до существующего решения

Поиск

Сортировка

Кратчайший путь

...

Разработать свой алгоритм

Рекурсия

Жадные алгоритмы

Динамическое программирование

...

Алгоритмы на графах

$$G = (V, E)$$

Алгоритмы на графах. Терминология

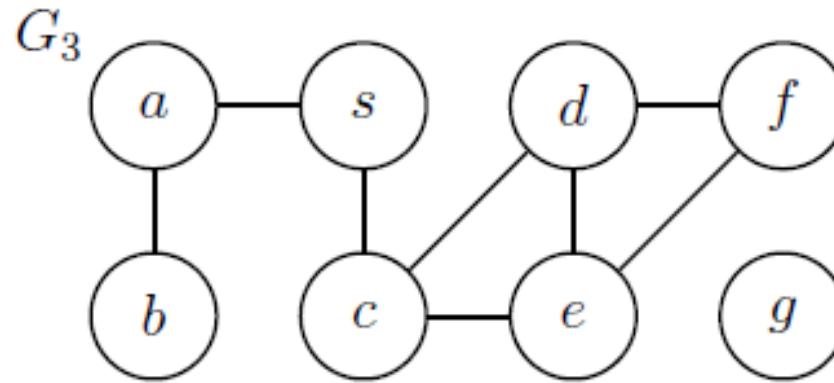
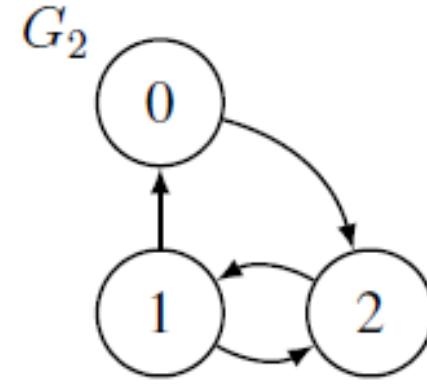
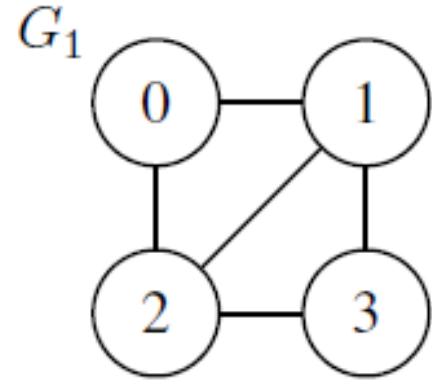
$$G = (V, E)$$

G – граф

V – узлы

$E \subseteq V \times V$ – ребра

Алгоритмы на графах. Терминология



Алгоритмы на графах. Зачем?

Компьютерные сети

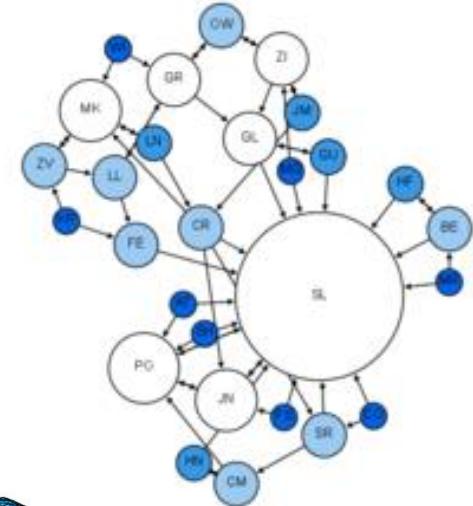
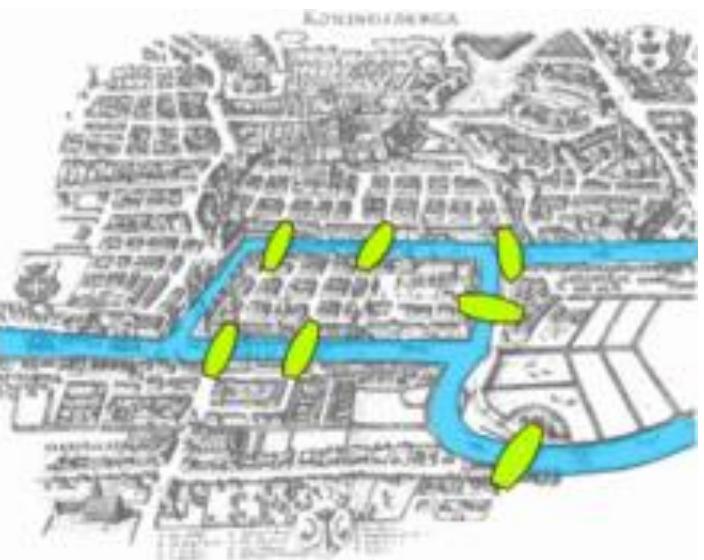
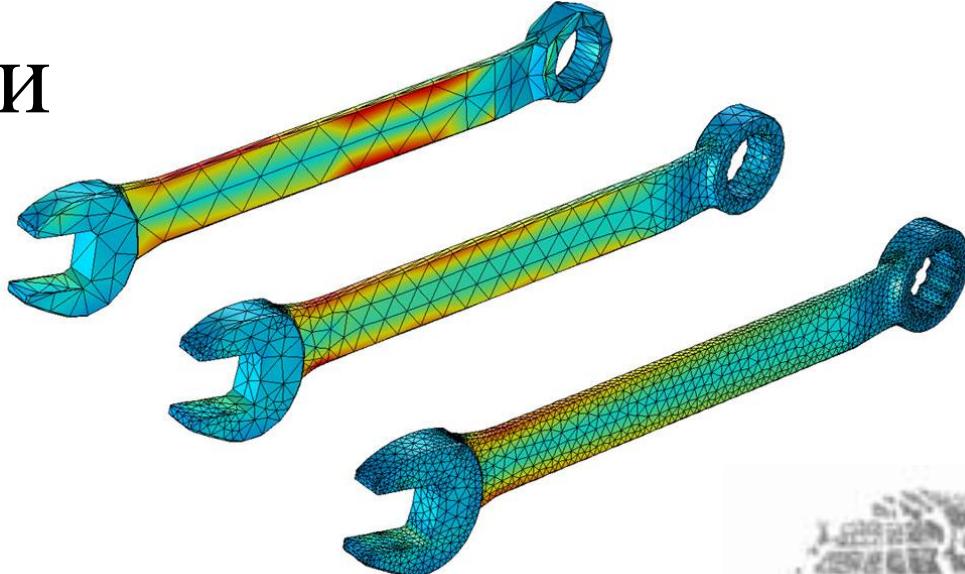
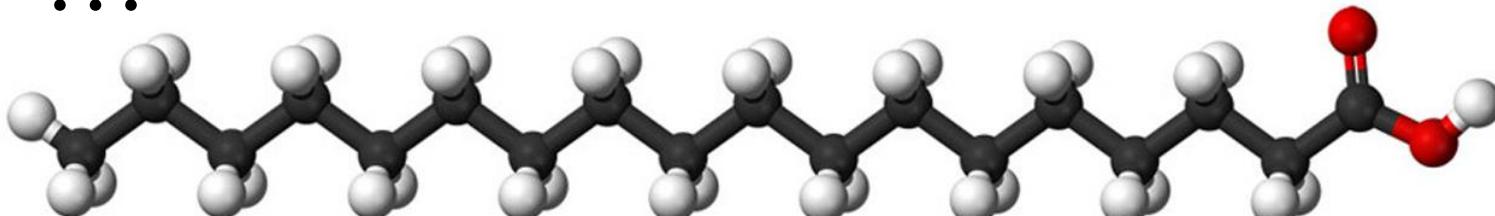
Социальные сети

Дорожные сети

Сетки

Молекулярные структуры

...



Алгоритмы на графах. Простой граф

Нет кратных рёбер

Нет петель

$$|E| = O(|V|^2)$$

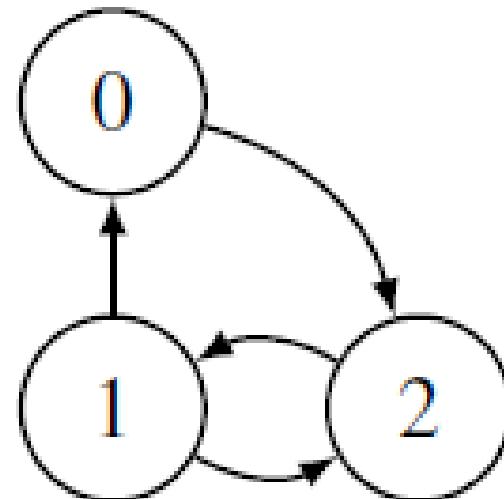
Алгоритмы на графах. Соседи

Множество исходящих соседей $u \in V$ is $Adj^+(u) = \{v \in V / (u, v) \in E\}$

Множество входящих соседей $u \in V$ is $Adj^-(u) = \{v \in V / (v, u) \in E\}$

Исходящая степень вершины $u \in V$ is $deg^+(u) = |Adj^+(u)|$

Входящая степень вершины $u \in V$ is $deg^-(u) = |Adj^-(u)|$



Алгоритмы на графах. Представление

Список ребер

Список соседей

Матрица смежности

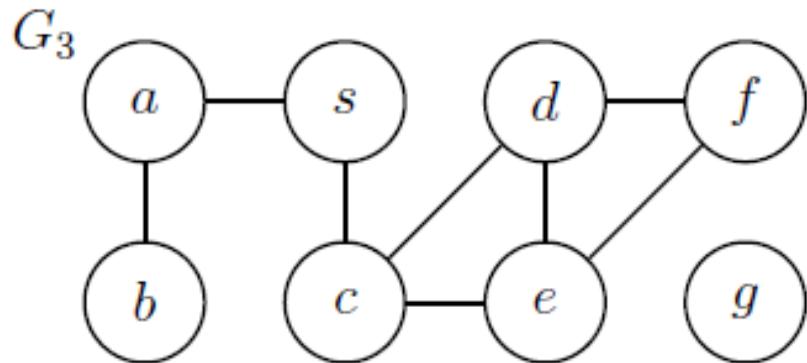
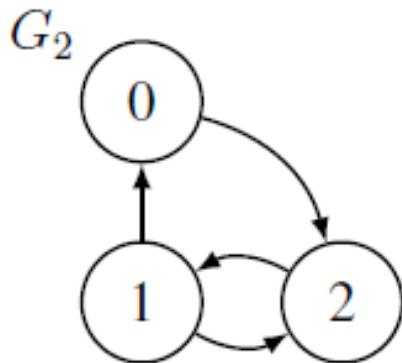
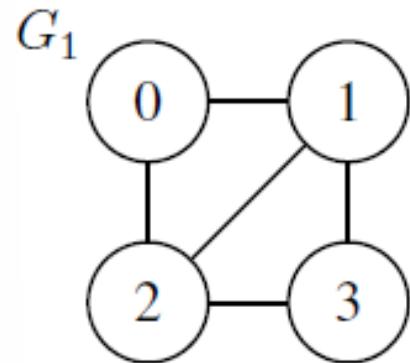
...

Space?

ExistEdge(v, w)?

Neighbors(v)?

Алгоритмы на графах. Представление



Ex 1 (Undirected)

```
G1 = [
    [2, 1],    # 0
    [2, 0, 3], # 1
    [1, 3, 0], # 2
    [1, 2],    # 3
]
```

Ex 2 (Directed)

```
G2 = [
    [2],      # 0
    [2, 0],   # 1
    [1],      # 2
]
```

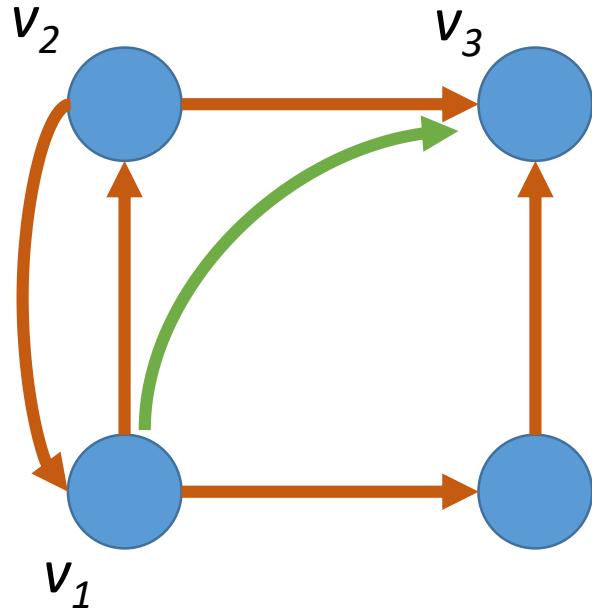
Ex 3 (Undirected)

```
G3 = {
    a: [s, b],
    b: [a],
    s: [a, c],
    c: [s, d, e],
    d: [c, e, f],
    e: [c, d, f],
    f: [d, e],
    g: []
}
```

Алгоритмы на графах. Пути

Путь – последовательность вершин:

$$p = (v_1, v_2, \dots, v_k), \text{ где } (v_i, v_{i+1}) \in E \text{ для всех } 1 \leq i < k.$$



$$p = (v_1, v_2, v_3)$$

$$l(p) = \text{length}$$

$$\delta(u, v) = \text{length of shortest path}$$

Алгоритмы на графах. Задачи по поиску пути

Single_Pair_Reachability (G, s, t);

Есть ли путь в G из s до t ?

Single_Pair_Shortest_Path(G, s, t);

Возвращает кратчайшее расстояние в G из s до t .

Single_Source_Shortest_Paths(G, s);

Возвращает кратчайшие расстояния из s до всех вершин и дерево кратчайших путей.

В порядке
увеличения сложности