



Обзор современного состояния области алгоритмов и структур данных

Калачёв Максим Александрович

Разработчик

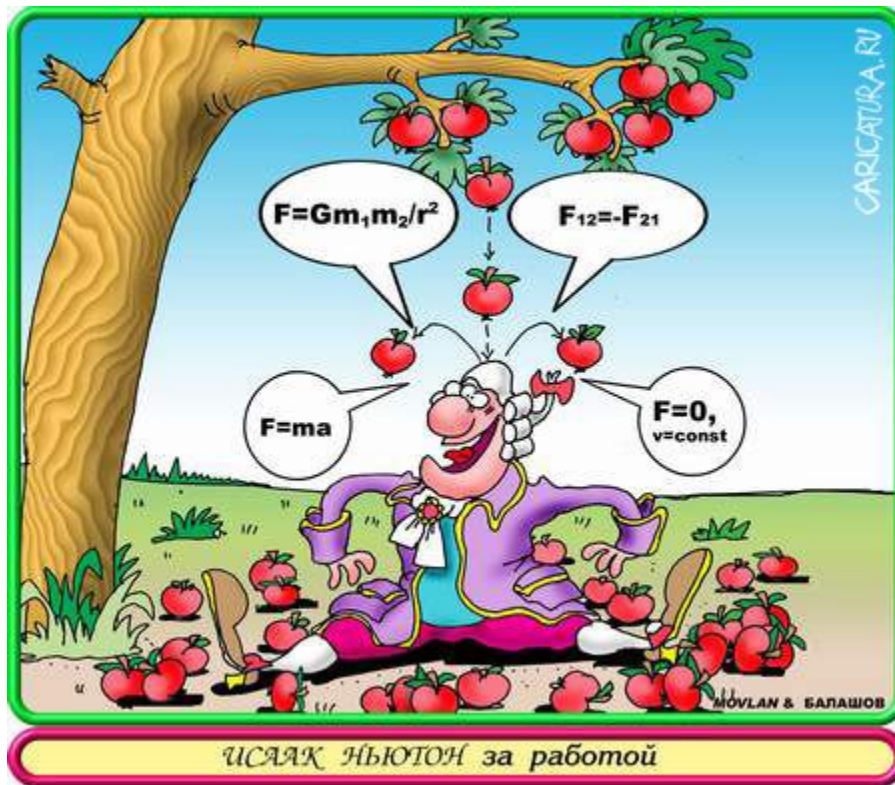
maxkalachev@yandex.ru

Cloud

Software

Services

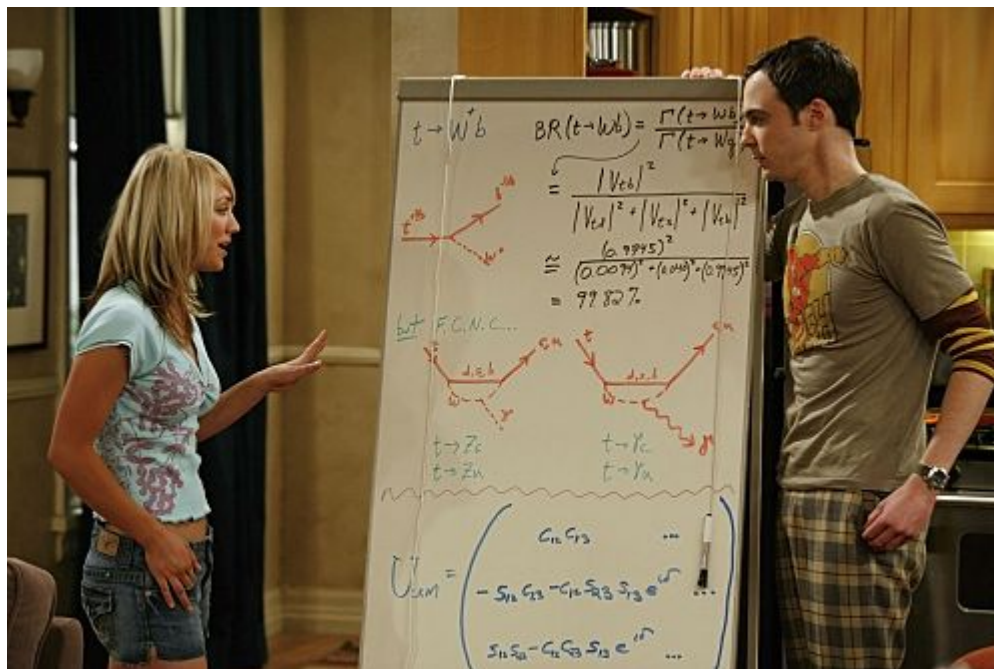
Идеи



План

- Computer Science
- Web-графы
- Случайные графы
- Highway dimension
- NP vs P
- Что осталось нерассмотренным
- Послесловие

Теоретики



11001101010001011110 SERVICES SOFTWARE 100110101 VIRTUALIZATION 01010010 SECURITY DATA LEAK
PREVENTION PROTECTION 101000 CONSULTING HARDWARE CAD/CAM/CAE 11101001000 OPEN SOURCE 10100101 LINUX
101010010 TECHNICAL SUPPORT SAM LEGAL SUPPORT 111010101010 ERP & CRM EDUCATION SERVICES

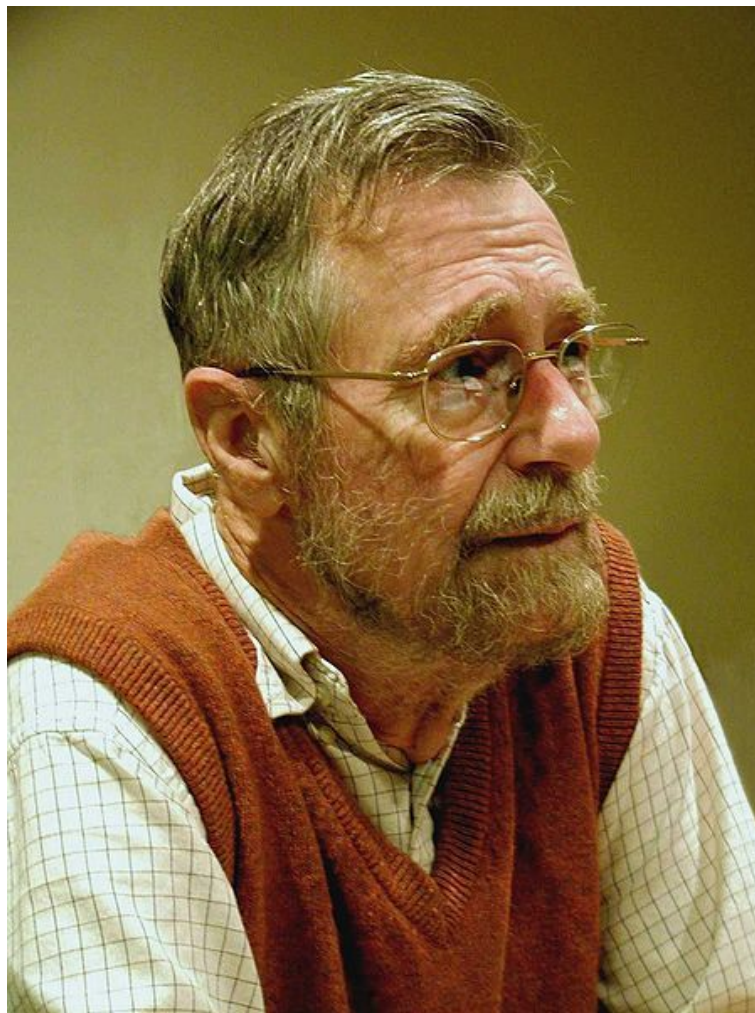
Практики



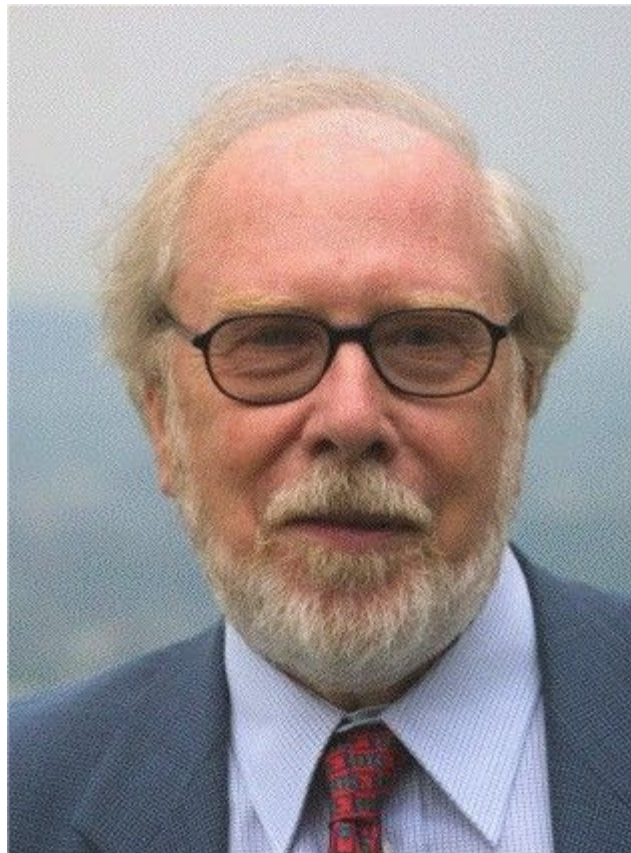
Программисты



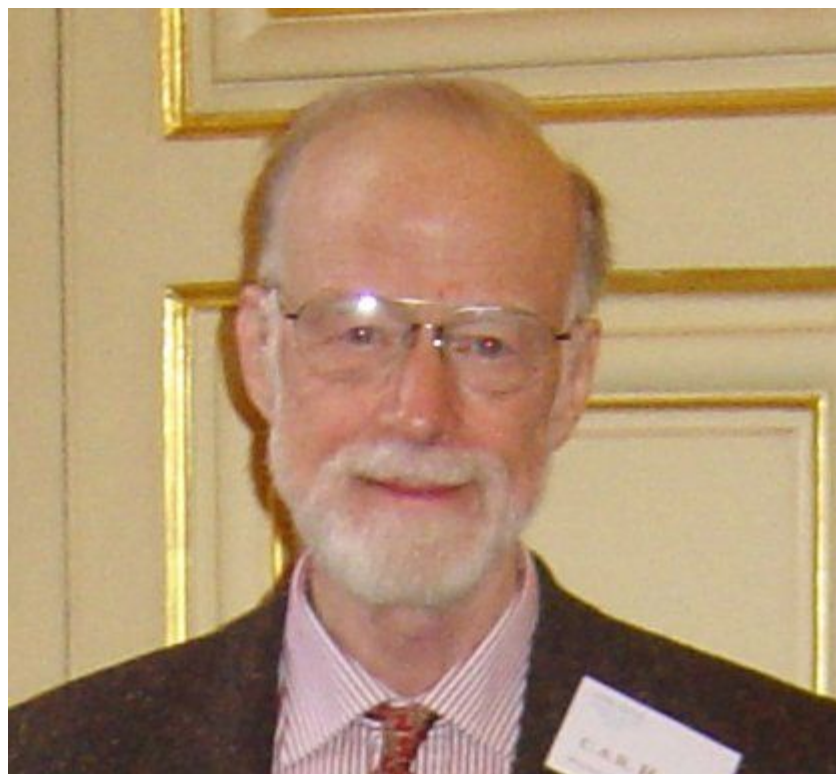
Эдгар Дейкстра



Никлаус Вирт



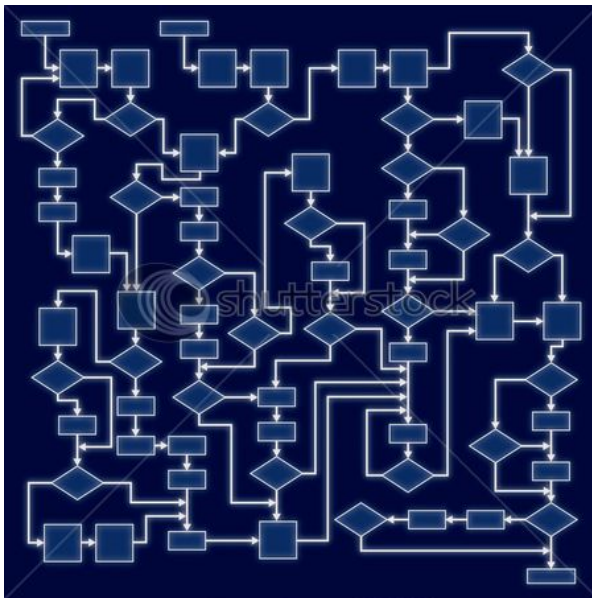
Чарльз Хоар



Дональд Кнут

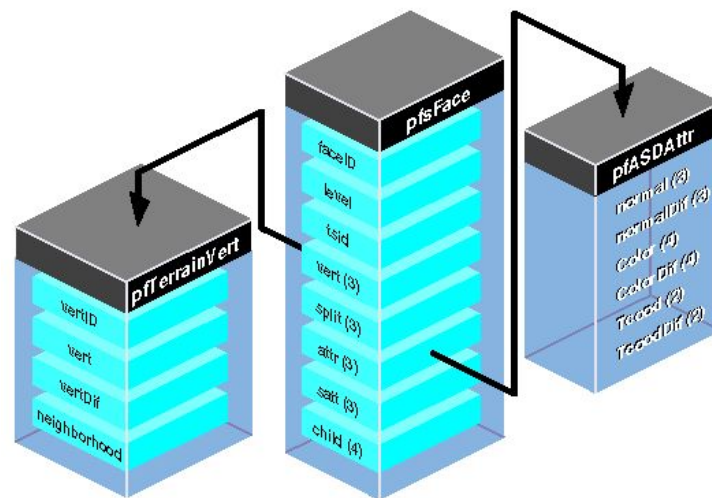


Программа



www.shutterstock.com · 49830424

+



Computer Science

$$P = \langle A_i, DS_i \rangle$$

$$A = \langle M_\rho, R_2 \rangle$$

M_ρ - множество процедур решения задачи

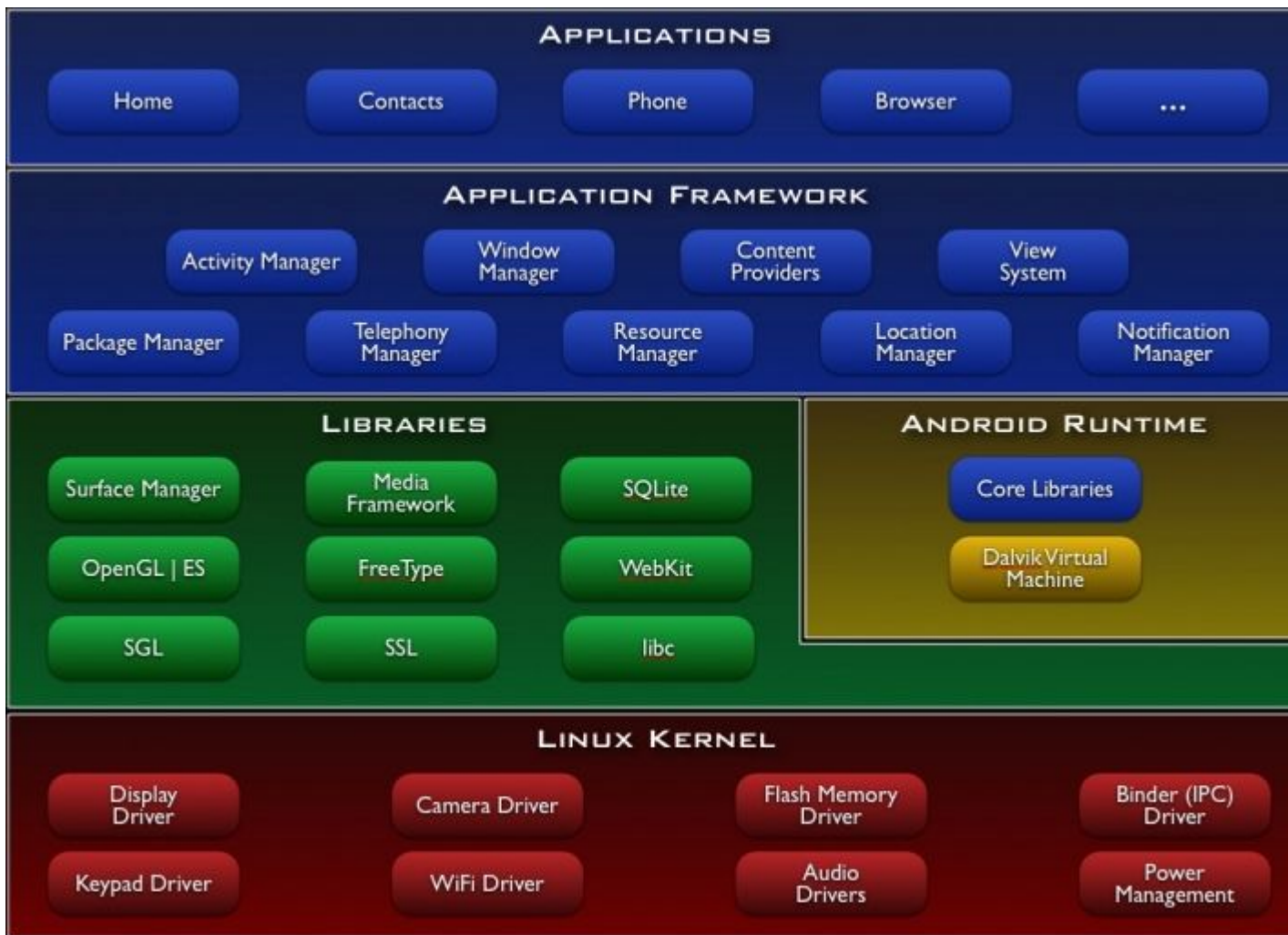
$R_2 \subset M_\rho^2$ - бинарное отношение на M_ρ

$(\rho_i, \rho_j) \in R_2 \Leftrightarrow$ после процедуры ρ_i выполняется процедура ρ_j

Закон Вирта

Программы становятся медленнее, чем компьютеры становятся быстрее

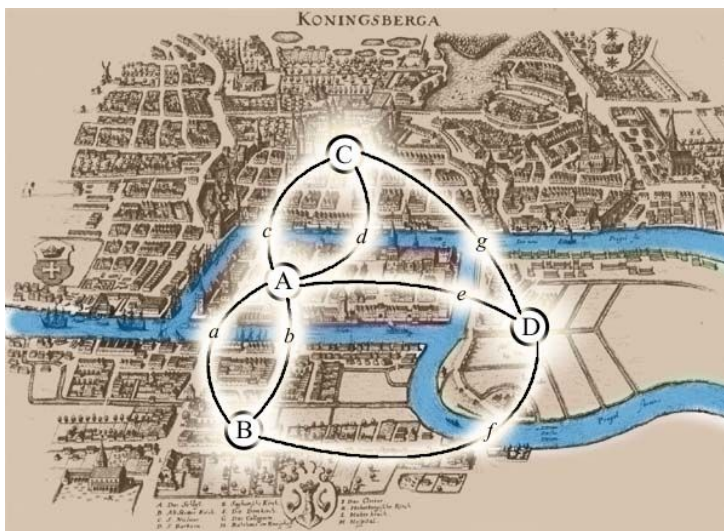
Абстракции



Математическое моделирование



Теория графов + Теория вероятностей = PROFIT

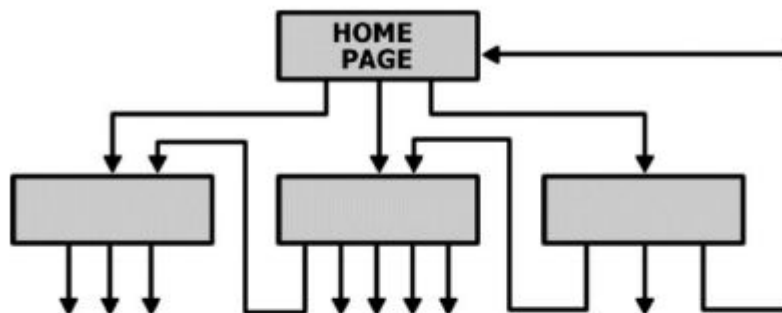


+

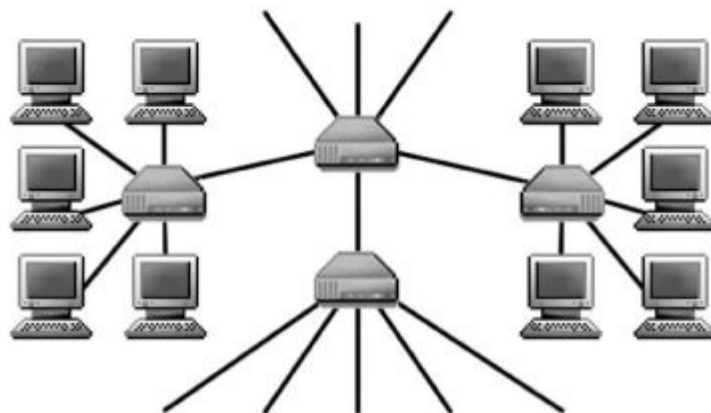


Веб-графы

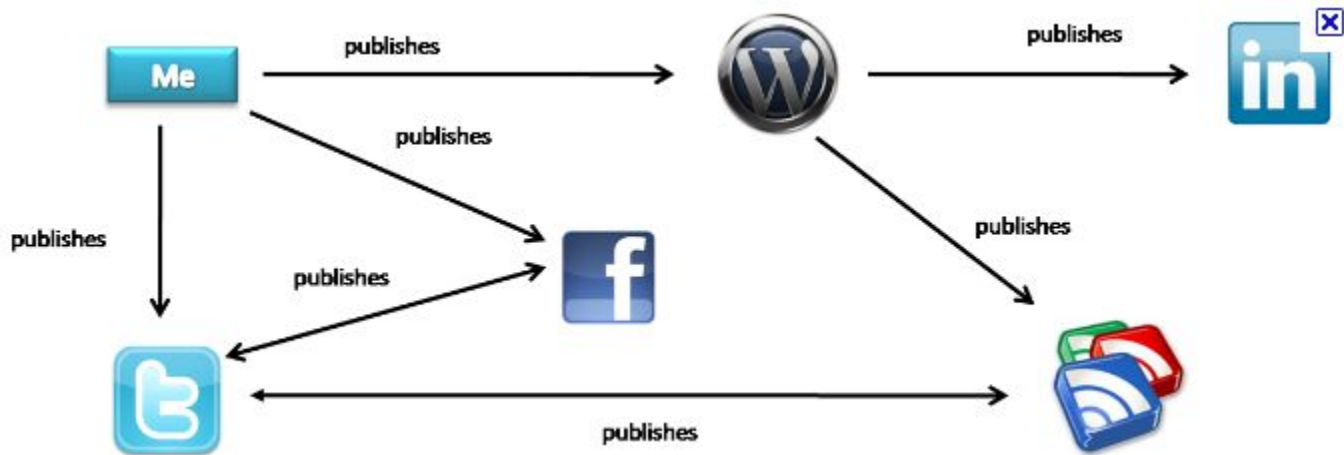
WORLD-WIDE WEB



INTERNET



Веб-графы



Случайные графы

Наблюдения Барабаши-Альберт

Как устроен web-граф?

Barabashi, Albert, 1999, 2000

5 млрд вершин, псевдомультиторграф

Ключевые свойства веб-графа:

- Разрежённость

на k вершин k^t рёбер, $k \geq 1$

- Диаметр графа $\in \{5, 6\}$

Теория о шести рукопожатиях

- Степенное распределение степеней вершин

$$P(d) \sim c / d^\lambda$$

$\lambda \approx 2.1$, c – нормирующий множитель

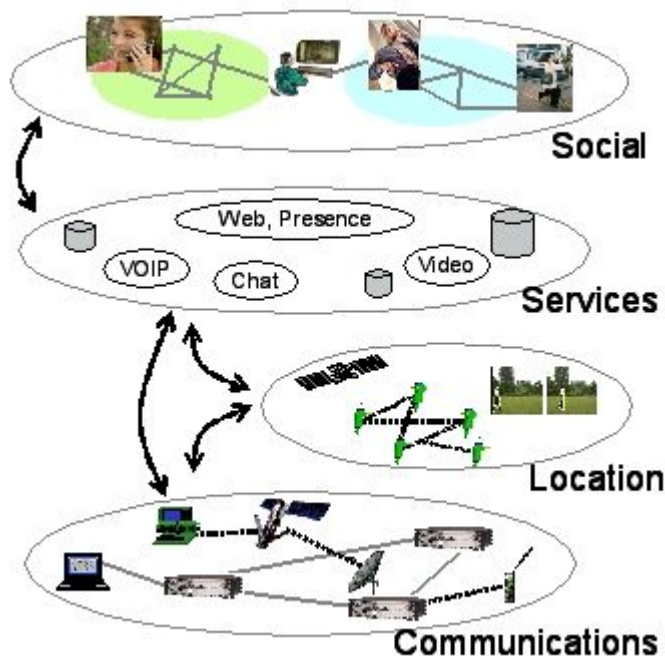
Случайные графы

Наблюдения Барабаши-Альберт

Веб-граф очень специфичен – разрежен и тесен

Степенной закон объединяет социальные, биологические и транспортные сети

Модели предпочтительного соединения



Случайные графы

Модель Эрдёша-Реньи

$G(n,p)$

$V = \{1, 2, \dots, n\}, E$

рёбра проводятся взаимно-независимо с вероятностью $p \in [0, 1]$ в соответствии со схемой Бернулли

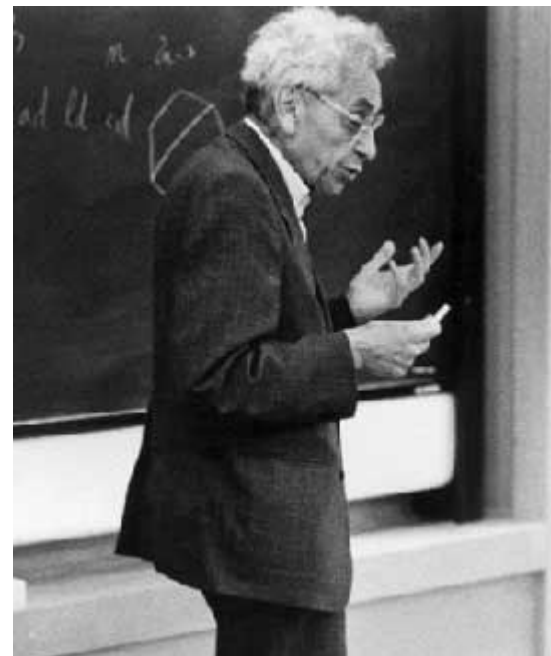
$e_1, \dots, e_m, m = C_n^2$ – количество всех испытаний

Вероятностное пространство $\langle \Omega_n, F_n, P_{n,p} \rangle$

$\Omega_n = \{G = (V_n, E)\}$ – множество элементарных событий

$F_n = 2^{\Omega_n}$ – множество событий

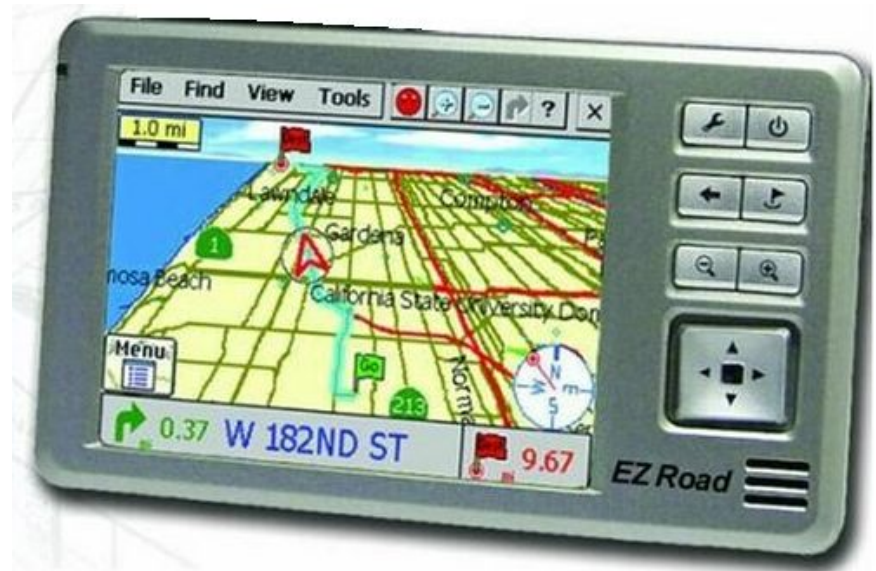
$P_{n,p}(G) = p^{|E|}(1-p)^{m-|E|}$ - вероятность появления конкретного графа



Транспортная интерпретация



Highway dimension



Highway dimension

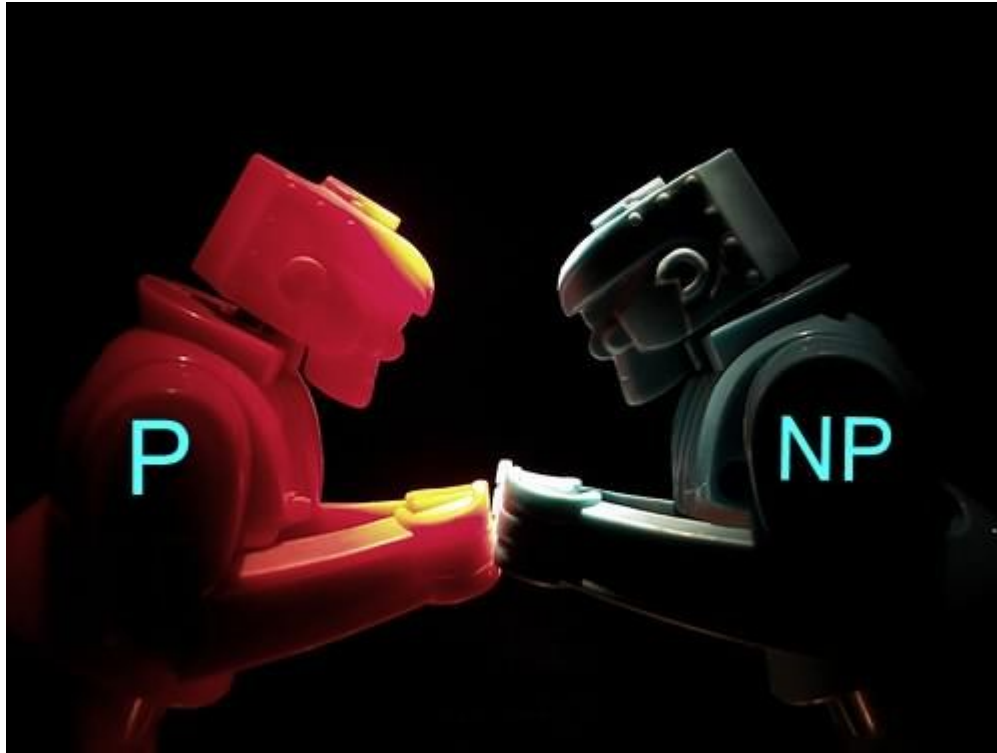
Почему современные алгоритмы на картах работают очень быстро

100000 млн вершин
Время работы 10^{-2} с

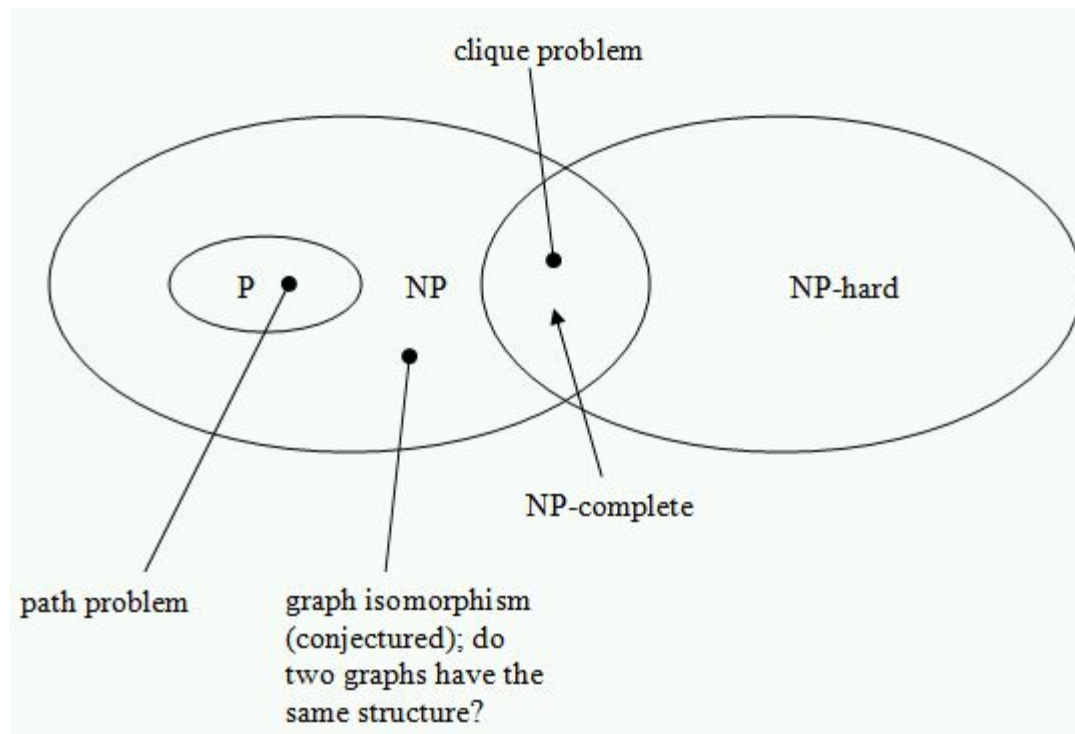
Интуитивные идеи:

- Указатели на дугах
- Поиск A^*
- Достижимость
- Шоссейная и желаемые иерархии
- Перевалочные пункты

P vs NP



Классы задач



P vs NP

Задача поиска задаётся алгоритмом C , который получает на вход условие I и кандидата на решение S и имеет полиномиальное, относительно I время работы.

S называется решением если и только если $C(S, I) = \text{true}$

NP – класс всех задач поиска, решение для которых может быть быстро проверено

P – класс задач поиска, решение для которых может быть быстро найдено

$P \neq NP$ – верно ли, что каждый раз, когда решение можно быстро проверить, его можно быстро найти

Задача о расписании

Задача о вершинном покрытии

$A \rightarrow B$



Андрей Михайлович Райгородский



Андрей Гольдберг



Что осталось нерассмотренным

- Параллельные алгоритмы
- Распознавание изображений
- Нейронные сети
- Генетические алгоритмы
- Нечёткие модели
- Строковые алгоритмы
- Комбинаторная оптимизация
- Численные алгоритмы
- Вычислительная геометрия
- Криптографические алгоритмы
- Компьютерная лингвистика
-

Так говорил Дейкстра

I think it wise, and only honest, to warn you that my goal is immodest. It is not my purpose to "transfer knowledge" to you that, subsequently, you can forget again. My purpose is no less than to effectuate in each of you a noticeable, irreversable change. I want you to see and absorb calculational arguments so effective that you will never be able to forget that exposure. I want you to gain, for the rest of your lives, the insight that beautiful proofs are not "found" by trial and error but are the result of a consciously applied design discipline. I want to inspire you to raise your quality standards. I mean, if 10 years from now, when you are doing something quick and dirty, you suddenly visualize that I am looking over your shoulders and say to yourself "Dijkstra would not have liked this.", well, that would be enough immortality for me.