

Большое O

Вычислительная сложность

Числа Фибоначчи

- Рекурсия – наивный алгоритм
- Массив / список
- Два последних значения
- Кэширование вызовов
- Большие числа
- Дерево вызовов

Числа Фибоначчи

- https://ru.wikipedia.org/wiki/Числа_Фибоначчи
- $F_0 = 0$
- $F_1 = 1$
- $F_n = F_{n-1} + F_{n-2}$
- <https://www.cs.usfca.edu/~galles/visualization/DPFib.html>

Оценка времени исполнения программы

- Асимптотический анализ
 - Число строк кода
 - Число операций
 - Число вызовов
- Экспериментальное измерение
 - Cell magic (Jupyter Notebook): %%timeit
 - Системное время
 - Многократный прогон

Пример

- $T_0 =$

- $T_1 =$

- $T_2 =$

- $T_3 =$

Понятие сложности алгоритма

- Computational complexity of an algorithm
- Вычислительная сложность алгоритма
 - Скорость возрастания необходимых ресурсов
 - Временная – время исполнения
 - Пространственная – объём памяти
- Зависимость от размера входных данных
 - Асимптотическая (на бесконечности)
 - Наилучшая, наихудшая, средняя...
 - Worst-case, best-case, average-case performance of an algorithm

O-нотация

- Big O notation – нотация «О большое»
 - Ordnung / Order of approximation – порядок аппроксимации
- $f(x) = O(g(x))$
 - $f(x) \leq C \cdot g(x)$ – функция ограничена сверху
 - $x \geq x_0$ – начиная с некоторого значения x_0
 - $C > 0$ – положительный коэффициент (константа)
 - Оценка «сверху»
 - Function $f(n)$ is upper-bounded by constant times $g(n)$, for all large n
- Нарисовать график
 - https://en.wikipedia.org/wiki/Big_O_notation
 - [https://ru.wikipedia.org/wiki/«O» большое и «o» малое](https://ru.wikipedia.org/wiki/«O»_большое_и_«o»_малое)

Порядок сложности: график + названия

- $O(1)$
- $O(\log n)$
- $O(\sqrt{n})$
- $O(n)$
- $O(n \log n)$
- $O(n^2)$
- $O(2^n)$
- $O(n!)$

О-нотации

- О большое
- О малое
- Омега большая
- Омега малая
- Тэта большая
 - Tight bound

Что такое «вычислительная сложность»?

- a) Время, необходимое для написания алгоритма
- b) Количество строк кода в алгоритме
- c) Скорость изменения времени выполнения алгоритма при увеличении размера задачи
- d) Объем памяти, занимаемый программой

Что такое «асимптотическая сложность алгоритма»?

- а) Оценка производительности алгоритма на маленьких значениях входных данных
- б) Приблизительный порядок роста времени выполнения алгоритма при стремлении размера задачи к бесконечности
- с) Максимальная возможная сложность алгоритма
- d) Минимальная возможная сложность алгоритма

Что описывает O-нотация (Big-O)?

- a) Нижнюю границу сложности алгоритма
- b) Верхнюю границу сложности алгоритма
- c) Среднюю сложность алгоритма
- d) Временную сложность алгоритма

Какие из перечисленных функций имеют наибольшую сложность?

- a) $O(2^n)$
- b) $O(n \log n)$
- c) $O(n^2)$
- d) $O(n!)$

Как соотносятся между собой O -нотация и Ω -нотация?

- а) O -нотация описывает верхнюю границу сложности, а Ω -нотация — нижнюю
- б) Обе нотации равнозначны и взаимозаменяемы
- в) O -нотация описывает нижнюю границу сложности, а Ω -нотация — верхнюю
- г) O -нотация всегда меньше или равна Ω -нотации