

Алгоритмы

Софья Копейкина

Август 2022

Содержание

1	Сортировки	3
1.1	Сортировки с асимптотикой $O(n^2)$	3
1.1.1	Сортировка выбором (selection sort)	3
1.1.2	Сортировка вставками (insertion sort)	3
1.2	Сортировки с асимптотикой $O(n \log n)$	3
1.2.1	Сортировка кучей (heap sort)	3
2	Куча (heap)	3
3	Бинарные деревья поиска	4
3.1	Декартово дерево, кучево, дуча (treap)	4
3.2	Красно-чёрное дерево (rb tree)	4

1 Сортировки

1.1 Сортировки с асимптотикой $O(n^2)$

Реализация: `algorithms/sorts/sorts_n2`

1.1.1 Сортировка выбором (selection sort)

На i шаге префикс длины i уже отсортирован. Чтобы продлить его на 1 элемент, найдем в оставшейся части минимум и поменяем его местами с i -ым элементом.

1.1.2 Сортировка вставками (insertion sort)

На i шаге префикс длины i уже отсортирован. Чтобы продлить его на 1 элемент, "тащим" i -ый элемент в сторону 0 до тех пор, пока его левый сосед больше. Таким образом, остановка произойдет тогда, когда все элементы слева не больше рассматриваемого и когда все элементы префикса справа больше - рассматриваемый элемент правильно вставлен.

1.2 Сортировки с асимптотикой $O(n \log n)$

Реализация: `algorithms/sorts/sorts_nlogn`

1.2.1 Сортировка кучей (heap sort)

Это оптимизация сортировки выбором - теперь минимум ищется за логарифм. И оптимизация эта происходит благодаря использованию кучи, которая обеспечивает такую асимптотику. Алгоритм заключается в следующем: строим кучу из этого массива и последовательно вытаскиваем из нее голову.

2 Куча (heap)

Реализация: `algorithms/heap`

Источник: `ru.algorithmica.org`

3 Бинарные деревья поиска

3.1 Декартово дерево, кучево, дуча (treap)

3.2 Красно-чёрное дерево (rb tree)