

Доказательство сложности $O(N)$ для `makeheap`

Алгоритм `makeheap(arr)` строит мин-кучу, вызывая `sift_down` для всех внутр. узлов массива длины n — от индекса $n/2 - 1$ до 0.

Оценка времени работы

Высота кучи $H = \lfloor \log_2 n \rfloor$.

Узлов на уровне h от листьев: примерно $\frac{n}{2^{h+1}}$.

Для узла на высоте h выполнение `sift_down` требует не более $O(h)$ операций.

Время на уровень: $C \cdot \left(\frac{n}{2^{h+1}}\right) \cdot h$, где C — константа.

Суммарное время:

$$T \leq C \cdot n \cdot \sum_{h=0}^H \frac{h}{2^{h+1}}$$

Считаем сумму:

$$S = \sum_{h=0}^{\infty} \frac{h}{2^{h+1}}$$

Используем формулу:

$$\sum_{h=0}^{\infty} x^h = \frac{1}{1-x}$$

Дифференцируем:

$$\sum_{h=0}^{\infty} h x^h = \frac{x}{(1-x)^2}$$

Подставляем $x = \frac{1}{2}$:

$$\sum_{h=0}^{\infty} \frac{h}{2^h} = 2$$

Тогда:

$$S = \frac{1}{2} \cdot 2 = 1$$

Итог:

$$\sum_{h=0}^H \frac{h}{2^{h+1}} \leq 1$$

$$T \leq C \cdot n \cdot 1 = O(n)$$

Вывод: алгоритм работает за линейное время $O(n)$.