

7. Анализ построения пирамиды «снизу вверх»

Описание алгоритма

1. Массив элементов рассматривается как бинарное дерево.
2. Начиная с последнего родительского узла, выполняется операция погружения для каждого узла.
3. Погружение продолжается до тех пор, пока не будет обработан корневой элемент.

Операция погружения

Операция погружения (*heapify-down*) восстанавливает свойства пирамиды:

- Текущий элемент сравнивается со своими дочерними узлами.
- Если свойства пирамиды нарушены, элемент обменивается местами с дочерним узлом с наибольшим (или наименьшим для мин-кучи) значением.
- Этот процесс повторяется для нового положения элемента, пока свойства пирамиды не будут восстановлены.

Анализ сложности

Для оценки сложности построения пирамиды «снизу вверх» рассмотрим количество операций погружения на каждом уровне дерева.

Высота дерева. Дерево высоты h содержит 2^h узлов на уровне h , начиная с уровня 0.

Количество узлов на каждом уровне. На уровне h находится 2^h узлов, но погружение на этом уровне выполняется за $O(h)$ операций.

Общая сложность. Суммарное число операций на всех уровнях дерева:

$$T(n) = \sum_{h=0}^{\log n} 2^h \cdot (\log n - h),$$

где n — общее количество элементов.

Упростим выражение:

$$T(n) \sim n \cdot \sum_{h=0}^{\log n} \frac{\log n - h}{n} = O(n).$$

Таким образом, сложность построения пирамиды «снизу вверх» равна $O(n)$.