

## Билет 21. Пирамидальная сортировка

**Основная идея:** Строим кучу, вытаскиваем максимумы, ставим их в конец. Profit!

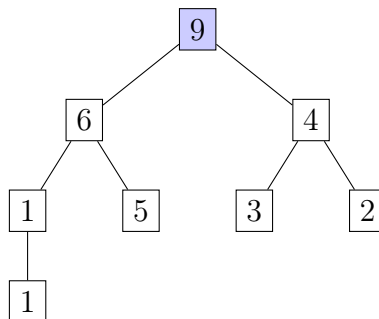
### Алгоритм пирамидальной сортировки

1. Построить кучу из исходного массива (in-place)
2. Для  $i$  от  $n-1$  до  $1$ :
  - Поменять местами корень (максимум) и последний элемент
  - Уменьшить размер кучи на 1
  - Выполнить **SiftDown(0)** для восстановления кучи
3. Результат: Отсортированный массив

### Пример сортировки шаг за шагом

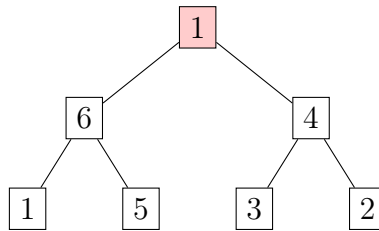
Исходный массив: [3, 1, 4, 1, 5, 9, 2, 6]

Шаг 1: Строим кучу (за  $O(n)$ )



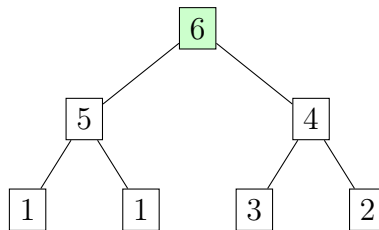
Массив: [9, 6, 4, 1, 5, 3, 2, 1]

## Шаг 2: Извлекаем максимум (9)



Меняем 9 и 1: [1, 6, 4, 1, 5, 3, 2, 9]

**SiftDown(1):** 1 → 6 → 5

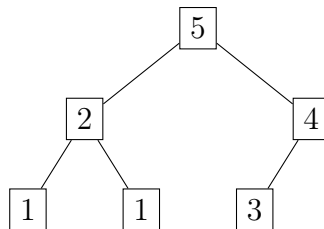


Массив: [6, 5, 4, 1, 1, 3, 2, 9]

## Шаг 3: Извлекаем максимум (6)

Меняем 6 и 2: [2, 5, 4, 1, 1, 3, 6, 9]

**SiftDown(2):** 2 → 5 → 1



Массив: [5, 2, 4, 1, 1, 3, 6, 9]

Продолжаем процесс...

После извлечения 5: [4, 2, 3, 1, 1, 5, 6, 9]

После извлечения 4: [3, 2, 1, 1, 4, 5, 6, 9]

После извлечения 3: [2, 1, 1, 3, 4, 5, 6, 9]

После извлечения 2: [1, 1, 2, 3, 4, 5, 6, 9]

После извлечения 1: [1, 1, 2, 3, 4, 5, 6, 9]

## Финальный результат

Отсортированный массив: [1, 1, 2, 3, 4, 5, 6, 9]

## Анализ сложности

- Построение кучи:  $O(n)$
- $n$  извлечений:  $n \times O(\log n) = O(n \log n)$
- Общее время:  $O(n) + O(n \log n) = O(n \log n)$
- Память:  $O(1)$  (in-place)

## Оптимизация для дорогих сравнений

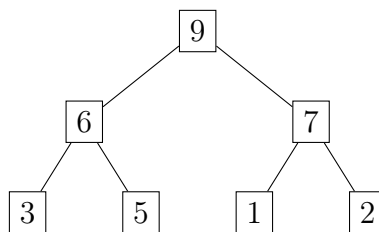
**Проблема:** При SiftDown приходится сравнивать элемент с обоими детьми

**Решение:**

1. Ставим в корень  $-\infty$
2. Просеиваем  $-\infty$  вниз (сравниваем только детей между собой)
3. Заменяем  $-\infty$  на последний элемент
4. Выполняем **SiftUp** для этого элемента

## Пример оптимизации

Исходная куча:



#### Обычный способ:

- Сравниваем 2 с 6 и 7  $\rightarrow$  3 сравнения
- Сравниваем 2 с 3 и 5  $\rightarrow$  ещё 2 сравнения
- **Итого:** 5 сравнений

#### Оптимизированный способ:

- Ставим  $-\infty$  в корень
- Сравниваем 6 и 7  $\rightarrow$  1 сравнение
- Сравниваем 3 и 5  $\rightarrow$  1 сравнение
- Заменяем  $-\infty$  на 2
- SiftUp(2): 2 сравнения
- **Итого:** 4 сравнения

### Сравнение с другими сортировками

Алгоритм	Время	Память	Устойчивость
Пирамидальная	$O(n \log n)$	$O(1)$	Нет
Быстрая	$O(n \log n)$	$O(\log n)$	Нет
Слиянием	$O(n \log n)$	$O(n)$	Да
Пузырьковая	$O(n^2)$	$O(1)$	Да