

CS2023 - Data Structures and Algorithms

In-class Lab Exercise

Week 8

Name: A.A.H. Pramuditha

Index: 200476P

Question 1

- Terminal results for the test inputs.

```
"D:\Semester 4\CS2023 - Data Structures and Algorithms\In Class labs\Lab 8\heap.exe"
Input array
4 17 3 12 9 6
Sorted array
3 4 6 9 12 17

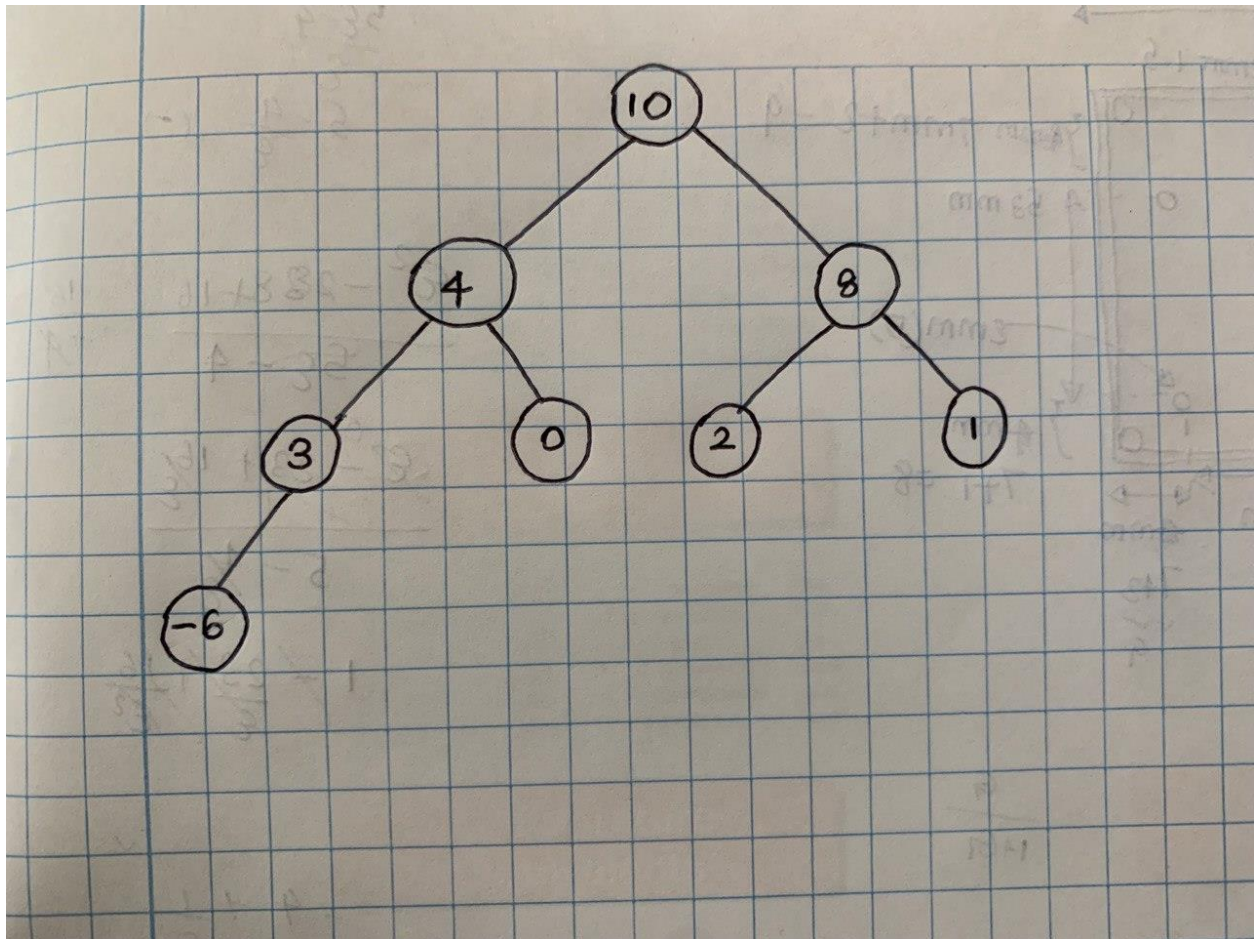
Process returned 0 (0x0)   execution time : 0.049 s
Press any key to continue.
```

- Terminal results for my inputs.

```
"D:\Semester 4\CS2023 - Data Structures and Algorithms\In Class labs\Lab 8\heap.exe"
Input array
3 -6 8 4 0 2 1 10
Sorted array
-6 0 1 2 3 4 8 10

Process returned 0 (0x0)   execution time : 0.013 s
Press any key to continue.
```

- Max-heap structure for the {3, -6, 8, 4, 0, 2, 1, 10} array.



- The time complexity of the heap sort for the three cases (best, worst and average) is $O(n \log n)$. As we have seen, to fully heapify an element whose subtrees are already max heaps, we need to keep comparing the element with its left and right children and pushing it downwards until it reaches a point where both its children are smaller than it. In the worst-case scenario, we will need to move an element from the root to the leaf node making a multiple of $\log(n)$ comparisons and swaps. During the building maximum heap stage, we do that for $n/2$ elements so the worst-case complexity of the building heap step is $n/2 * \log n$ which is nearly equal to $n \log n$.

Link for the GitHub repository - [hashirupramuditha/CS2023---Data-Structures-Algorithms \(github.com\)](https://github.com/hashirupramuditha/CS2023---Data-Structures-Algorithms)