**Lab 11**

**HEAPS**

**INTRODUCTION:**

In computer science, a heap is a specialized tree-based data structure which is essentially an almost completetree that satisfies the heap property: in a max heap, for any given node C, if P is a parent node of C, then the key (the value) of P is greater than or equal to the key of C. In a min heap, the key of P is less than or equal to the key of C. The node at the "top" of the heap (with no parents) is called the **root node.**

* Shape property: A binary heap is a complete binary tree; that is, all levels of the tree, except possibly the last one (deepest) are fully filled, and, if the last level of the tree is not complete, the nodes of that level are filled from left to right.
* ii. Heap property: All nodes are either greater than or equal to (Max-Heaps) or less than or equal to (MinHeaps) each of its children, according to a comparison predicate defined for the heap.



**HEAP OPERATIONS:**

**i. Insert:**

1. Add the element to the bottom level of the heap.

2. Compare the added element with its parent; if they are in the correct order, stop.

3. if not, swap the element with its parent and return to the previous step. As an example of binary heap insertion, say we have a max-heap as shown in Figure.



**ii. Delete:**

1. Replace the root of the heap with the last element on the last level.

2. Compare the new root with its children; if they are in the correct order, stop.

3. if not, swap the element with one of its children and return to the previous step.



**OBJECTIVE:**

* The objective of this experiment is to implement a Max Heap as well as a Min Heap using array representation of binary trees.
* Learn deletion and insertion of heap using array.

**APPLICATION:**

**Heaps** are used in many famous algorithms such as:

* Dijkstra's algorithm for finding the shortest path,
* the **heap** sort sorting algorithm,
* Implementing priority queues, and more.
* Essentially, **heaps** are the data structure you want to use when you want to be able to access the maximum or minimum element very quickly.

**ISSUE:**

As such did not face any complicated issue while making its codes. Still have an issue in minimum heap.

**CONCLUSION:**

To conclude, I understand how to insert a node/value into a binary heap while maintaining the parent-child relationship and how to update a binary heap after extracting its root.