Heap Data structure Implementation

There are two types of Heap

Root should be always maximum compared to left and right child and its true for all sub trees as well.

Diagram

Description automatically generated

Root should be always minimum and applicable for sub trees as well

A picture containing watch

Description automatically generated

* Quick access to lowest and highest element from N number of elements we achieve using O(1) operations
* Chart

  Description automatically generated
* Example : We have list of Jobs in Queue and each job assigned a priority number
* Max priority Job in root and find in O(1) time .Once job is completed hepify our heap in O(Log N) to get the next max priority Job

After inserting an element into the heap, it may not satisfy the heap property. In that case, we need to adjust the locations of the heap to make it heap again. This process is called **Heapifying.**

Implementation :

* Implemented using Priority Queue
* Prioirty Queue expects comparator to prioritize element
* Default ordering is increasing order of element
* Integers added to Priority Queue add in ascending order