**Heaps and Priority Queues**

A picture containing text, clock

Description automatically generatedA heap is a **complete** binary tree with the following properties:

* The value in the root is a smallest item in the tree
* Every nonempty subtree is a heap

6 has the highest priority item.

In OS, each process have some priority. The jobs are kept in a queue. We always run the highest priority job. FIFO (as in the regular queue) is not okay bc we have some jobs whose priority is higher.

There are 2 operations for priority queue:

* offer
* poll

You can implement them using unsorted list (6 18 29 20 28 39 66 37 26 76 32 74 - Can you find children of 18? Yes, 1\*2+1 and 1\*2+2 - red ones are index of 18, results are indexes of children, left and right respectively). Add to the end with offer, find smallest in list and remove it with poll.

You can also use sorted list (smallest to largest). Find correct place and insert with offer, remove the first element (which is smallest) with poll. Sorting from largest to smallest would be easier for remove.

Heap is one of the implementations of the priority queue.

Algorithm for Inserting in a Heap

Insert the new item in the next position at the bottom of the heap

**while** new item is not at the root and new item is smaller than its parent

Swap the new item with its parent, moving the new item up the heap

Diagram

Description automatically generated with medium confidence

Inserting 89 is the best case. So running time : (1)

Inserting 1 is the worst case. Running time depends on height (always logarithmic since it is complete binary tree). So running time is : (logn)

A picture containing diagram

Description automatically generatedDiagram

Description automatically generated with low confidenceDiagram

Description automatically generated with low confidence

Algorithm for Removal from a Heap

Remove the item in the root node by replacing it with the last item in the heap (LIH)

**while** item LIH has children and item LIH is larger than either of its children

Swap item LIH with its smaller child, moving LIH down the heap

A picture containing diagram

Description automatically generated

------> You can only remove 6 (highest priority element)

If you want to remove 6, 66 must be moved to the place of 6 to preserve the complete binary tree structure.   
Then order must be fixed. You have to find smallest from children and change your node with it.

Running time of remove depends on height of the tree:

* In the worst case, the element we put to root circulates down until a leaf. 🡪 (logn)
* In the best case 🡪 (1)

Diagram

Description automatically generated with medium confidence

A picture containing diagram

Description automatically generatedA picture containing diagram

Description automatically generated

In heap, you can keep the same element more than once. Items with the same priority can be handled in different order but before the other jobs which has lower priority. Order same items pulled out is not guaranteed. You can make one of them a little bit larger or lower to guarantee the order.