

Download and Extract

An initial setup of files is provided to you via a shell script: [Download potd-q46](#)

Using a terminal, extract the initial files by running the shell script you just downloaded (you will need to navigate to the directory where you saved the file):

```
sh potd-q46.sh
```

Your files for this problem will be in the `potd-q46` directory.

POTD

In this problem you are given a max-heap, which means the keys of parent nodes are always greater than or equal to those of the children and the highest key is in the root node (from Wikipedia). Note that the priority queue starts from index 1 of the vector (the number in index 0 can be considered as NULL), so think carefully about the index-relation between parent and child.

You are given an implementation of `enqueue()` function and required to implement the `procolateDown()` function, which is the helper function of `dequeue()`. You can implement it recursively like what you are introduced in the lecture, but you are encouraged to try the non-recursive version like the given `enqueue()` function for practice.

Example Output

```
Current Priority Queue is: 15 12 13 7 10 8 3 1 4 -1 6 5
15 Pop from Priority Queue
13 Pop from Priority Queue
12 Pop from Priority Queue
10 Pop from Priority Queue
8 Pop from Priority Queue
7 Pop from Priority Queue
6 Pop from Priority Queue
5 Pop from Priority Queue
4 Pop from Priority Queue
3 Pop from Priority Queue
1 Pop from Priority Queue
-1 Pop from Priority Queue
```

Upload Solution

Drop files here or click to upload.

Only the files listed below will be accepted—others will be ignored.

Files

☐ Heap.cpp
not uploaded

POTD 46

Total points: 0/1

Score: 0%

Question

Value: 1

History:

Awarded points: 0/1

[Report an error in this question](#)

[Previous question](#)

[Next question](#)

Save & Grade

Save only