

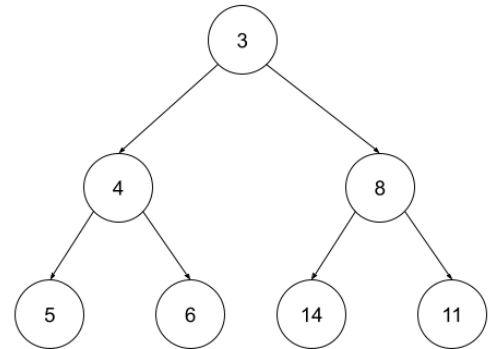
COMSC 205 Worksheet 20: Heaps

If you are viewing this from gDocs on your computer, go to “File -> Make a copy” to edit

Question 1. (Min) heaps are **complete** binary trees that have this **invariant** for each node:

- The value at a node is smaller than every value in its left and right subtree.

1a. Let's look at the heap on the right. What would peek() return?



1b. Let's now insert the value 2. Describe in words where we would insert 2 as a leaf node to start.

1c. Next, let's reconstruct the heap. Using paper or a tablet, draw what the heap will look like after swapping 2 into the right place. Note: you **do not** need to turn in your drawing as part of the Gradescope submission.

1d. Starting with the **original heap** (without 2 inserted) shown above, draw what the heap looks like after 3. Again, you **do not** need to turn in this drawing. The algorithm for reconstructing a heap after deletion is:

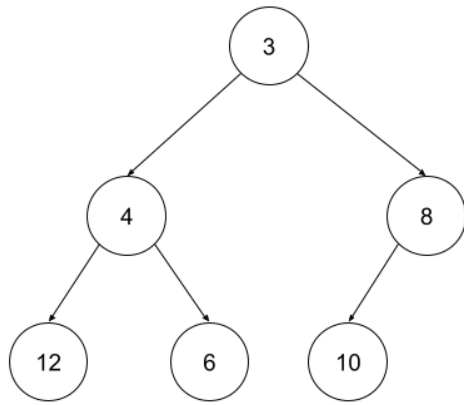
1. Move the rightmost (“last”) leaf to the root.
2. If the new root is larger than either child, repeatedly swap it with the **smallest** child until we have a heap again.

--	--

Original heap with 2 inserted

Original heap with 3 removed

Question 2. Consider the following heap:



0	1	2	3	4	5	6

2a. Fill out the array representation of the heap in the boxes above.

2b. Next, fill out the array representation of the heap after `insert(2)`. You may find it easier to first percolate up the value visually, then fill out the array.

0	1	2	3	4	5	6

2c. What is the updated index of the value 3?