Homework assignment 5:
Suggested due date: Friday, November 17 2017 at 03:30pm

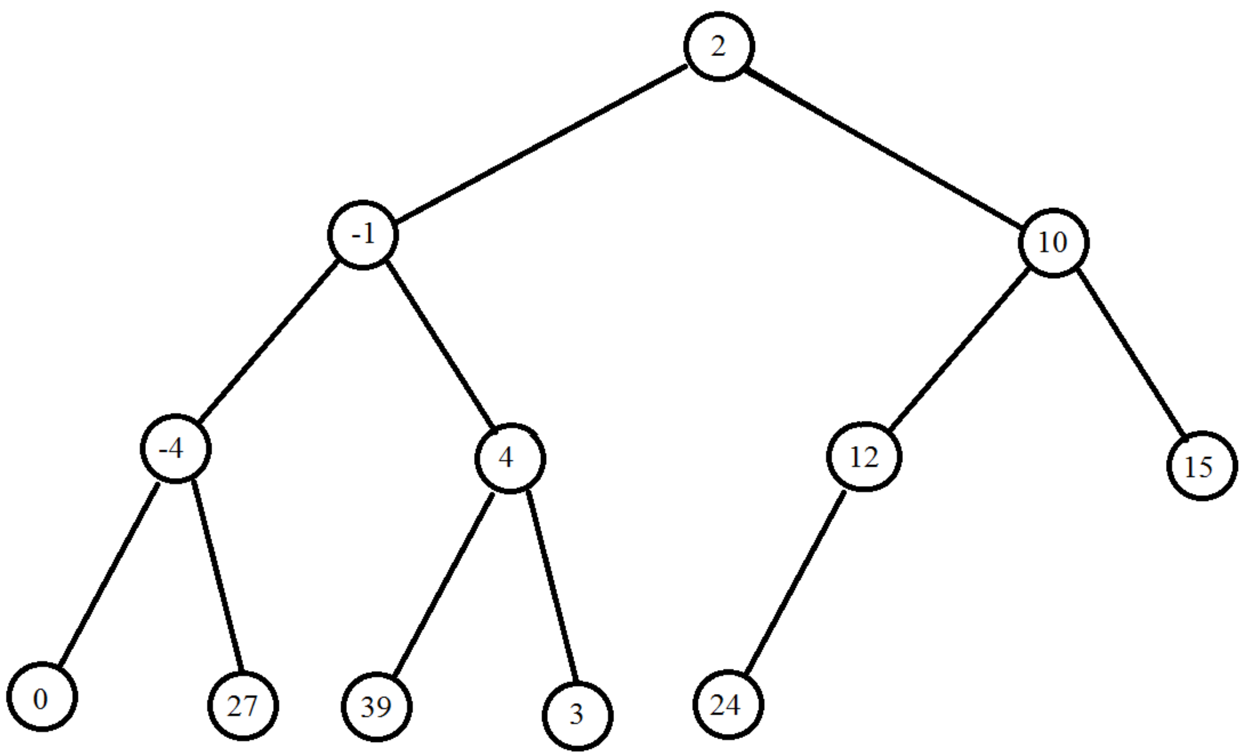
1. Where in a min heap the largest element resides? (Assume all elements are distinct)Explain.
2. Make the min heap by successive insertions into an initially-empty min heap. Re-draw the heap each time an insertion causes one or more swaps.

   -4, 2, 10, 12, -1, -3, 15, 76

3. Make the max heap by successive insertions into an initially-empty max heap. Re-draw the heap each time an insertion causes one or more swaps.

   3.1. -4, 2, 10, 12, -1, -3, 15, 76
   3.2. 61, 25, -12, 16, 20, 97, -1, 100

4. Is this a max-heap? If not, apply max-heapify function to change it to a max heap (Hint: remember the assumption for the max-heapify function: The left and the right sub-trees below node i are max-heap)

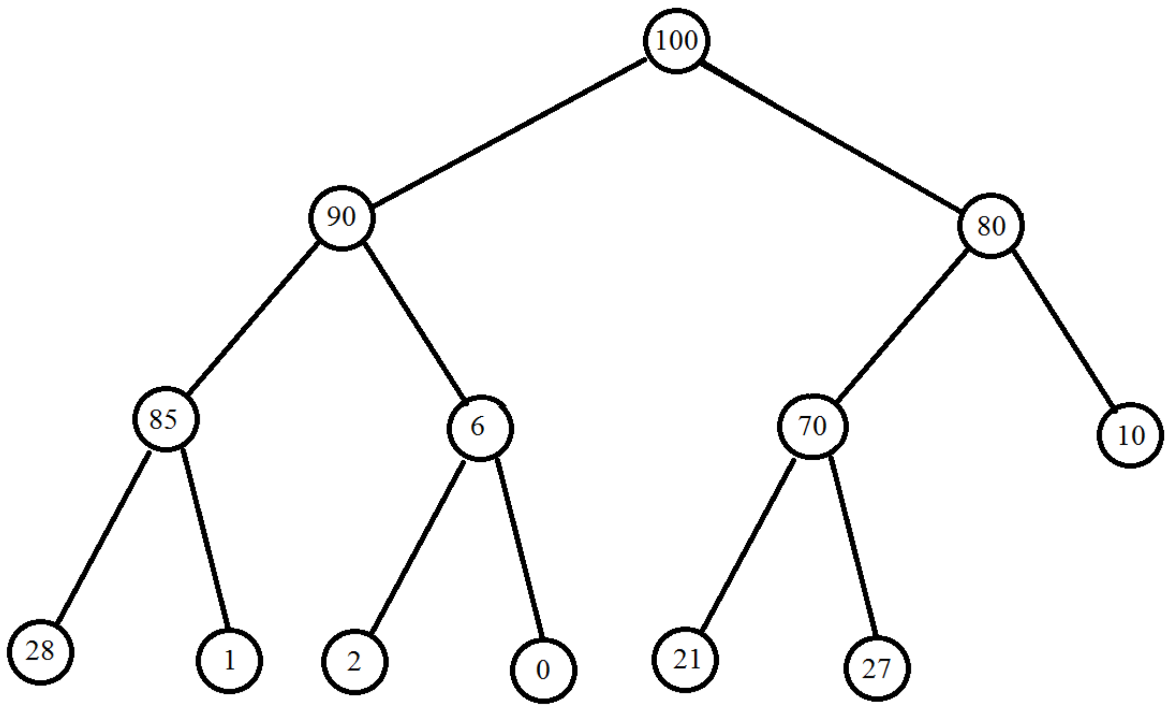


5. Is this array max-heap? If not, change it to the max heap by putting the elements on a binary tree and then applying the max-heapify function.

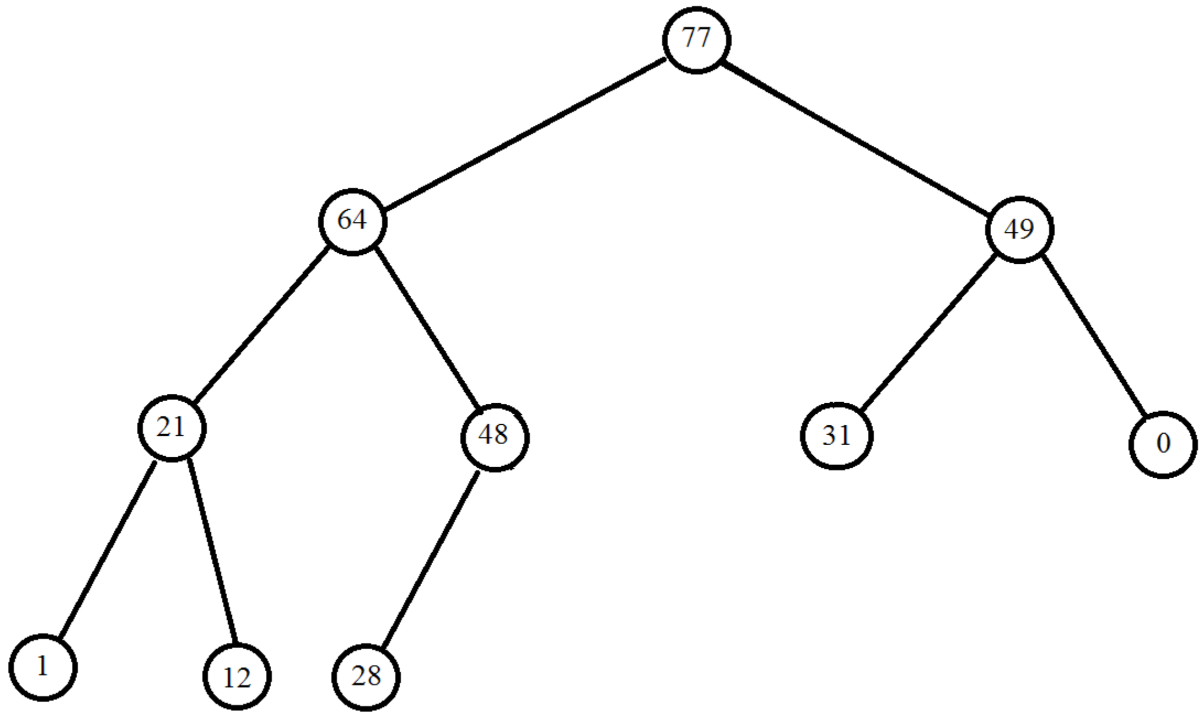
0, 20, 11, 10, 2, -2, 16, 5, 18, 19

6. Insert
   6.1. 95
   6.2. 160
   6.3. 37

   into the below max heap.



7. Delete the root of the below max-heap



8. Sort the below array using heapsort.

    8.1. 21, 0, 1, -1, -3, 10, 21, 25, 31
    8.2. 87, 21, 0, -1, -22, 10, 17, 11, -3, 10

9. Suppose that instead of binary heaps, we wanted to work with ternary heaps. Suggest an appropriate indexing scheme so that a complete tree will yield a contiguous sequence. (Hint: Let the root have an index of 0. Demonstrate your indexing scheme for a complete ternary tree of size 12.)
10. Use induction to prove that 1 + 2 + 4 + …+ $2^h = 2^{h+1} - 1$.
11. What is the minimum and maximum number of leaves in a binary heap that has height h. Explain.
12. Prove that a binary heap with n elements has height $\lfloor \log n \rfloor$ (Hint: Use the result of the above question)
13. Prove that a binary heap with n nodes has exactly $\lceil n/2 \rceil$ leaves. (Hint: It has two solutions:
    1- If a binary heap has **n nodes**, then the **last internal node** has index $\lfloor n/2 \rfloor$
    2- Use induction:

2.1. **base case:** n=1 $\Rightarrow \lceil n/2 \rceil = \lceil 1/2 \rceil = 1 \Rightarrow$ One node, one leaf ➔ Proved!

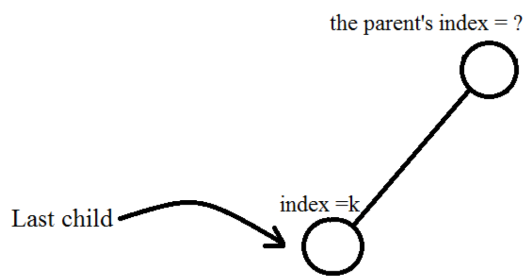
2.2. **Inductive step**: Assumption: The statement is correct for n=k,

2.3. **Inductive step**: Now prove that it is correct when n=k+1. In order to prove the statement, you can use the below information:

a. We know that the parent node in the max-heap can have either one or two children, and

node i ➔ left child index= 2i, right child index= 2i+1

b. Assume the parent (index i) of the last node/leaf (index k) has only one child (look at the below figure). So, is the index of the last child (=k which also shows the number of the nodes in the whole max-heap) **even or odd?** (look at part a)



c. Now add one more child to the tree. (where do you add it? To the right side of the last child!) ➔ Now you have k+1 nodes.

d. Based on the assumption you had $\left\lceil \frac{k}{2} \right\rceil$ leaves, now you have $\left\lceil \frac{k}{2} \right\rceil$+1 leaves and it is equal to $\left\lceil \frac{k}{2} \right\rceil + \frac{2}{2} = \left\lceil \frac{k}{2} + \frac{2}{2} \right\rceil = \left\lceil \frac{k+2}{2} \right\rceil$ and from **part c** we can say $\left\lceil \frac{k+2}{2} \right\rceil = \left\lceil \frac{k+1}{2} \right\rceil$ (Explain why?) ➔ we proved that the statement is correct for n=k+1 when the parent of the last node has only one child. :D

e. Now we need to prove that the statement is correct when n=k+1 and the parent of the last node has 2 children. (Just repeat steps b to d)

14. Show that there are at most $\left\lceil \frac{n}{2^{h+1}} \right\rceil$ nodes of height h in any n-element heap. (Hint: Use the result of the above question, and induction (on *height*))