Name: ___________________________________  Name: ___________________________________

Name: ___________________________________

Priority Queue Group Exercises

By Ali Ebrahimpour

Please work in groups of 2 or 3 to work the following problems. Use additional paper as needed, and staple the sheets together before turning them in. **ONLY TURN IN 1 WORKSHEET/ANSWERS PER TEAM.**

1. For a data structure to be a *min-priority queue* we must have these functions:
   
i. **Insert:** To insert a new element to the priority queue
   
   ii. **FindMin:** To find the minimum element of the priority queue
   
   iii. **ExtractMin:** To find and extract the minimum element of the priority queue, then re-build it if necessary.

For any 2 of the data structures below, describe how we can use them to implement a priority queue and the time complexity of each of the above three functions when we use them.

   a) A simple array (list)
   
b) A linked list
   
c) A simple list and a pointer to the element of the list with minimum value
   
d) A sorted list
   
e) A sorted linked list
   
f) Heap
2. Suppose we are implementing a heap by using an array for which the \(i\)th element of the array is the parent node of the nodes \(2i\) and \(2i+1\) of the array (if not empty), in the tree representation of the heap. Suppose we have the array below:

\[
1 \ 5 \ 8 \ 11 \ 6 \ 10 \ 14 \ 13 \ 15 \ 8 \ 9 \ 12 \ 15 \ 14 \ 16 \ 8 \ 18
\]

a) Draw the tree of the above array.

b) Is this tree a min-heap? Why?

c) Write the steps that should be taken to fix the tree to become a min-heap.
3. If we had a sorted array in the previous question, was it a heap? Explain your answer.

4. Compute the total number of elements in a complete (i.e. full) heap with a tree of height $h$. 
5. Suppose that we want to use two heaps, one min-heap and one max-heap, to make a data structure that provides us with these functions, which have the complexities shown in parentheses:

   i. Find the median (in $O(1)$)
   ii. Extract median (in $O(\log n)$) – this function also rebuilds the heaps
   iii. Insert a new element (in $O(\log n)$)

Explain how we can use a max- and min-heap in each of the operations

*Hint:* Look at the max-heap and min-heaps below, and try to do each of the 3 operations on them. (The median for these numbers is 11)