Homework 5
CS 3385

1. In this problem you'll derive the minimum and maximum numbers of elements in a heap of height h.
   (a) What are the minimum number of nodes at level i? The root is at level 0, the children of root are at level 1, the grandchildren at level 2, etc. (*Note*: we're not asking for the total number of nodes – just the number of nodes at a given level of the tree. Also, we assume a level with no nodes does not exist.)
   (b) What are the maximum number of nodes at level i?
   (c) Derive (showing your work) the **minimum** total number of nodes in a tree of height h (meaning the 0-based index of the last layer in the tree is h) using the summation symbol ($\sum$).
   (d) Using properties in Appendix A of the textbook, give a closed-form solution (no $\sum$ symbol) of the minimum total number of nodes.
   (e) Derive the maximum total number of nodes in a tree of height h using the summation symbol ($\sum$).
   (f) Give a closed-form solution of the maximum total number of nodes.

2. Prove that in any subtree of a max-heap, the root of the subtree contains the largest value occurring anywhere in that subtree. Assume a function $parent^j(i)$ (recall functional iteration discussed in section 3.2 of the textbook). You will need to show that A[$parent^j(i)$] $\geq$ A[i] forall $j \geq 1$. You'll do this using mathematical induction on j.

3. Where in a max-heap might the smallest element reside, assuming that all elements are distinct?

4. Is an array that is in sorted order a min-heap?

5. Consider the array $\langle 23, 17, 14, 6, 13, 10, 1, 5, 7, 12 \rangle$.
   (a) Draw the tree associated with this array.
   (b) Is this a max-heap? If not, circle the offending piece(s) in your drawing of the tree.

6. Using 1-based indices, show that, with the array representation for storing an n-element heap, the leaves are the nodes indexed by $\lfloor n/2 \rfloor + 1, \lfloor n/2 \rfloor + 2, \ldots, n$. *Hint*: one approach is to use the left-child and right-child functions, and show for what values of i the child indices are not valid, i.e. greater than n.

7. Using figure 6.2 as a model (which uses 1-based indices), illustrate the operation of `max-heapify(A,3)` on the array $A = \langle 27, 17, 3, 16, 13, 10, 1, 5, 7, 12, 4, 8, 9, 0 \rangle$.

8. Using figure 6.3 as a model, illustrate the operation of `build-max-heap` on the array $A = \langle 5, 3, 17, 10, 84, 19, 16, 22, 9 \rangle$.