Due : Oct. 10 (10/19), 13:00
Late submission due: Oct. 11 (10/20), 13:00

Overview

This assignment consists of the contents of the scope of the mituem exam. Please read the problems carefully and answer the questions.

General Notes

- Please don't forget to write down your student ID and name.
- Your answers must be in English.
- You must hand your assignment in one pdf file. *Do not* submit in other formats, like doc/docx (MS Word), hwp (HWP).
- You can submit handwritten answers; Please write clearly. If we cannot recognize what you wrote, you will get a 0.
- When we say 'describe an algorithm', you must give one of a high-level description, pseudo code, or a source code for Java. We do not accept any literal descriptions.

Problems

Answer the following problems.

1. Write its simplest tight Big-O notation for each function and program. (1 point each)

   (a) $2n^2 + 5n + 3$

   (b) $2^n + n!$

   (c) $n^2 + n \log n + 1$

   (d) $2020x^2 + 2^x$

   (e)
   ```
   for i = 0 to n
       for j = 0 to n
           k++;
   ```

   (f) The time complexity of the function $f(n)$ is $O(n)$.
   ```
   i = 1
   while i < n
       f(i);
       i = i * 2;
   end while
   ```

   (g)
   ```
   for i = 0 to n
       for j = 0 to 2^i
           k++;
   ```

2. What is the number of # printed when calling hashBrown(5) with the function below. (3 points)

   (a)
   ```
   public void hashBrown(int i){
       if(i>1){
           hashBrown(i/2);
           hashBrown(i/2);
       }
       System.out.print("#");
   }
   ```

3. The following is a heap implemented using an array. Answer the following questions.

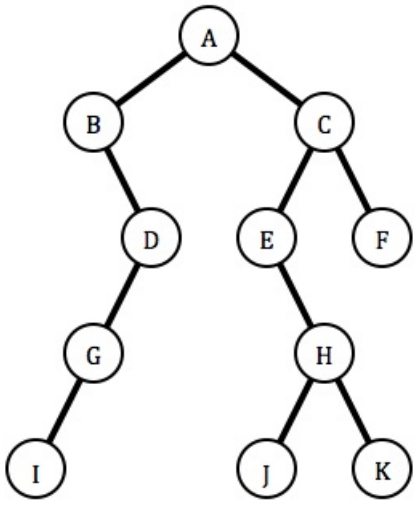
Index	0	1	2	3	4	5	6
Data	0	2	5	6	8	9	10

   (a) Show the array after deleting the node with the smallest data in the above heap. (3 points)

   (b) Show the array after inserting the node with data 7 in the above heap. (3 points)

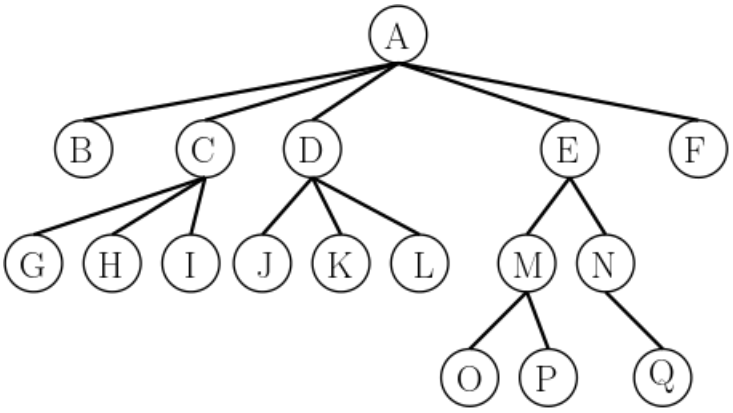
4. Explain how to implement a queue using stacks. (10 points)

5. Give the pre-order (4 points) and post-order (4 points) traversals of the following tree, rooted at A.



6. Describe in detail an implementation (4 points) of a priority queue based on a sorted array. Show that this implementation achieves O(1) time for operations min and removeMin and O(n) time for operation insert. (6 points)

7. The following questions refer to the given tree, write the answers to each question.



(a) Which node is the root? (1 point)

(b) What is the height of the tree? (1 point)

(c) What are the external nodes? (1 point)

(d) How many descendents does node D have? (1 point)

(e) How many ancestors does node P have? (1 point)

(f) Which nodes are in the subtree rooted at node E? (1 point)

8. Suppose that there is a doubly linked list maintaining a reference to the first and last node in the list, along with its size. Write the worst-case running time of each operation below with respect to n using Big-O notation.

   (a) addFirst(e) //insert the element e to the beginning of the list (2 points)

   (b) removeLast() //delete and return the element at the end of the list (2 points)

   (c) prev(N)//return the previous node of node N (2 points)

   (d) search(e) //is the element e is in the list? (2 points)

   (e) set(i, e) // replace the ith position of the list with the element e (2 points)