DATA STRUCTURES & ALGORITHMS 1

BATCH – B [FRIDAY APRIL 05, 2019: 3:30 PM – 6:30 PM]

LAB ASSIGNMENT – 9 CODE:assign09

NOTES:

1. Please carefully read all assignments and there is **no choice**.
2. Use the **template** for this assignment.
3. **Follow variable and function naming conventions**
   a. except for variables in for-loop, none of the other variables should be a single character.
   b. The variable names and function names should indicate what they are storing/computing. For this assignment, we have given you some of the variable names and function names to use. They are highlighted as function_name or variable_name
   c. All global variable should start with 'g_'
4. Indentation improves readability. Please pick an indentation style and **indent your code** appropriately.
5. Follow constants and type naming
   a. All constants should be defined using IFNDEF and DEFINE
   b. All structures should have a TYPEDEF to a simpler name
6. When in doubt about naming or style conventions, consult the following link: https://users.ece.cmu.edu/~eno/coding/CCodingStandard.html
7. If your project contains multiple files, add all the files into a folder, zip and submit. Only the zip file should follow (rollno based) naming convention.

PROBLEMS [Total Marks: 20]:

Today we are going to practice how to implement a priority queue using binary heap.

Exercises:

1. *[5 Marks]* Create a linked list to create and maintain student data. The student structure should have the following fields
   a. Name
   b. Roll no
   c. CGPA
   d. hackerRank

   (Hint: safely assume that no two people have the same rank)

   You should implement the following functions

a. addStudent (adds to the end of the linked list)
b. updateStudentByRollNo
c. deleteStudentByRollNo
d. printStudentByRollNo
e. printAllStudents

2. *[5 Marks]* Implement a (Min) Binary Heap index (create a structure as necessary)
   a. BuildHeap (uses hackerRank as the key)
   b. Insert (inserts into heap. Called when a node is inserted into the linked list)
   c. DeleteMin
   d. Percolateup (this is an internal function used by insert and/or deletemin)
   e. Percolatedown (this is an internal function used by insert and/or deletemin)
3. *[5 Marks]* Use the Binary Heap index to implement a function: getTopRankingStudent [Use DeleteMin appropriately]

   (Hint: This prints the details of the student and then removes him/her from both the heap and the linked list)
4. *[2 Marks]* find a way to pretty-print the heap (should look like a tree) call the function prettyPrintBHeap
5. *[3 Marks]* Split the code into multiple files and use ‘make’ to organize your project

Note: Those who doesn’t get Question-1 evaluated in lab gets a zero.