Data Structures & Algorithms – Fall 2015

(BS-SE-F14 Morning & Afternoon)

Assignment # 2

Submission Deadline: Tuesday, 22nd December, 2015 (till 5:00 PM)

Submission Folders:

\\printsrv\Teacher Data\Ahmad Ghazali\DSA-F15\A2-Morning

\\printsrv\Teacher Data\Ahmad Ghazali\DSA-F15\A2-Afternoon

Instructions

- **This is an individual assignment. You are NOT allowed to work/submit in form of group. Absolutely NO collaboration is allowed. Any traces of plagiarism/cheating would result in an "F" grade in this course.**
- Do **NOT** copy even a single line of code from any other person or book or Internet or any other source.
- This assignment needs to be submitted in **Soft Form**. Write all the code in a **single CPP file**. See the **Submission Procedure** at the end.
- Late submissions will NOT be accepted, in any case.

In this assignment, you are required to implement the singly linked list class as explained in **Exercise 6 (a) (b)** on **Page 174-175** of your textbook (1st Edition). The linked list class will NOT have a head pointer; rather it will have two pointers ***left*** and ***right*** (***l*** and ***r***) as explained in the above-mentioned exercise.

Each node in the linked list will contain an integer. Your linked list class will maintain the nodes of the linked list **sorted in increasing order**.

Your linked list class should provide the following functions:

Constructor	To create an empty linked list.
Destructor	To destroy a linked list.
bool isEmpty()	To check whether the linked list is empty or not
void moveRight (int n)	Explained in Exercise 6(a) (Page 175)
void moveLeft (int n)	Explained in Exercise 6(b) (Page 175)
void printLeft ()	This function will print the data of all nodes which occur to the left of ***left*** pointer including the value pointed to by the ***left*** pointer. This function should NOT move/modify the ***left*** and ***right*** pointers.

void printRight ()	This function will print the data of all nodes which occur to the right of ***right*** pointer including the value pointed to by the ***right*** pointer. This function should NOT move/modify the ***left*** and ***right*** pointers.
bool insert (int val)	This function should insert **val** into the linked list maintaining the sorted order, and should return true. After insertion the pointer ***left*** should be pointing to the node which has just been added to the linked list.
bool delete (int val)	This function will delete the first node containing the value **val** from the linked list and return true. If the specified value is not found in the linked list, this function should not make any changes to the linked list and should return false. After removal the pointer ***left*** should be pointing to the node which occurred just before the node which has been deleted. If the first value of the linked list has been removed, the pointer ***left*** should be pointing to the new first node of the linked list.

You are also required to write a **menu-driven main function** which allows the user to use all of the functionalities of your linked list class. The menu should be **user-friendly** and should allow the user to use all the above-mentioned functionalities.

Note that: These *good programming practices* will also have their (significant) weightage in the marking of this assignment:

- There should be no memory leaks, dangling pointers, or any other type of runtime error in your program.
- Comment your code intelligently. *Uncommented code may not be given any credit.*
- Use meaningful variable and function names.
- Indent your code properly.
- Do NOT use any global or static variables.

Moreover, if your submitted program gives an error or a warning message at the time of compilation, you will get a ZERO in the assignment.

Submission Procedure

You are required to submit this assignment in **soft form**. Make sure that you follow these steps to submit your code:

1) You should submit only **ONE** CPP file containing all your code of this assignment.
2) Mention your **Name**, **Roll Number** and **Section** in comments at the top of your code.
3) The name of the CPP file MUST be your complete Roll Number (e.g. **BSEF14M789.cpp**).
4) Copy the **.CPP file** (e.g. **BSEF14A456.cpp**) in the submission folder of your section.

☺ GOOD LUCK! ☺