CSC 228-01 Data Structures and Algorithms, Spring 2020
Instructor: Dr. Natarajan Meghanathan

Assignment 3 (Programming): Design and Implementation of a Θ(n) Algorithm to Delete all the Occurrences of an Element in a List ADT

Due by: Feb. 13th, 11.59 PM

In this programming assignment, you will design and implement algorithms to delete all the occurrences of an element (integer data) in a List ADT: implemented using a dynamic array as well as using a singly linked list.

You are given the code for both dynamic array and the singly linked list-based implementations of a List ADT. The main function in these two programs are already setup to generate random integers and fill up the contents of the list. The deleteALL(int deleteData) function is called on the list to delete all the occurrences of the 'deleteData' in the list.

Your task is to implement the deleteALL(...) function in both the dynamic array and singly linked list-based implementations of the List ADT. Your algorithms/implementations should run in Θ(n) time, where *n* is the number of elements in the list before the deletions.

The main function is written in such a way that it will print the contents of the list before and after the deletions.

Testing Procedure:
You would test your code by entering 15 as the list size and 5 as the maximum value for any element. The list will be then filled with random integers in the range [1...5]. As there are going to be 15 integers in the list, it will be definitely the case that there will be more than one occurrence for one or more integers in the list. Enter one of such integers as the input for the 'deleteData' and the deleteALL(...) function is called with the deleteData to delete all the occurrences of deleteData in the list.

Submission:
Submit Items 1 and 2 as separate .cpp files.
1) The entire .cpp file for the dynamic array-based implementation of the List ADT.
2) The entire .cpp file for the singly linked list-based implementation of the List ADT.
3) Screenshots of the execution of both the implementations (using the testing procedure described above) submitted together in a word document.