**Project 2**

**Lists and Object-oriented Programming**

Project 2 is an individual assignment that focuses on object-oriented programming and lists. The list will be a doubly-linked list and it will be implemented using nodes and pointers. The data contained in the list will objects from a class.

1. Project data: You will choose one of the following classes to define: Tractor, Hobby, Park, Country, RadioStation, Clothing, Food, ClubMember, Animal, Tree, and ProgrammingLanguage. You may also choose a sub-class of one of the previous choices. Your class code will include a .h class interface file and a .cpp class definition file. You will choose the data attributes of your class.
2. Project data structure: Additionally, you will create classes for a doubly-linked list that will manage a collection of the objects from section (a). The list will use dynamically allocated nodes.
3. For a user to work with your data and list, you should write a fully functional menu with the following options:
   1. Adding to the front of the list
   2. Adding to the back of the list
   3. Searching and displaying a specific item in the list
   4. Edit a specified item from the list
   5. Removing an identified item from the list
   6. Displaying the entire list
   7. Ending the program

The following files are required:

The class driver similar to WorkingWithBooks.cpp (c).

The class declaration similar to Book.h (a).

The class implementation file similar to Book.cpp (a).

A header file for the node class similar to Node.h (b).

A header file for the list similar to SinglyLinkedList.h (b).

A class implementation file for the list class similar to SinglyLinkedList.cpp (b).

Additionally follow all these project instructions:

* Your project folder name should contain your last name in it.
* Your node file should contain your class name in it (*e.g.*, TractorNode.h)
* Your list file name should contain your class name in it (*e.g.*, TractorList.h and TractorList.cpp)
* Your program should run. Comment out any part of your file that does not work and briefly explain why you think it does not work.
* Submit a .zip zipped project folder (not a .rar folder) which will include all the project’s files.

Grading [100 points total]:

Code should be documented and use meaningful identifiers 10 points

Code should be well-structured and easy to read 10 points

Files named properly and project folder zipped up properly 10 points

Adding to the doubly-linked list 15 points

Removing from the list 15 points

Searching the list 15 points

Editing the list 15 points

Overall running of the program 10 points