**Lab Assignment #1 – Using Fundamental Data Structures**

Due Date: Friday midnight, Week 3

Purpose: The purpose of this Lab assignment is to:

1. Design and develop Applications that incorporate fundamental data structures such as:

* Singly Linked Lists
* Doubly Linked Lists
* Circularly Linked Lists

References: Read the course’s text chapter 3 and the lecture slides. This material provides the necessary information that you need to complete the exercises.

Be sure to read the following general instructions carefully:

- This assignment must be completed individually by all the students.

- See the naming and submission rules at the end of this document

- You will have to provide a **demonstration video for your solution** and upload the video together with the solution on **eCentennial** through the assignment link. See the **video recording instructions** at the end of this document.

**Exercise 1**

**If your first name starts with a letter from A-J inclusively:**

Add a method *swapTwoNodes* to *SinglyLinkedList* class from week 2 lecture examples. This method should swap two nodes *node1* and *node2* (and not just their contents) given references only to *node1* and *node2*. The new method should check if *node1* and *node2* are the same node, etc. Write the main method to test the *swapTwoNodes* method. **Hint**: You may need to traverse the list.

**If your first name starts with a letter from K-Z inclusively:**

Add a method *swapTwoNodes* to *DoublyLinkedList* class from week 2 lecture examples. This method should swap two nodes *node1* and *node2* (and not just their contents) given references only to *node1* and *node2*. The new method should check if *node1* and *node2* are the same node, etc. Write the main method to test the *swapTwoNodes* method. **Hint**: You may need to traverse the list.

(4 marks)

**Exercise 2**

**If your first name starts with a letter from A-J inclusively:**

Use the *SinglyLinkedList* implementation of the textbook (week 2 lecture examples. Write a method for **concatenating two singly linked lists L1 and L2**, into a single list L that contains all the nodes of L1 followed by all the nodes of L2. Write a main method to test the new method. **Hint**: Connect the end of L1 into the beginning of L2.

**If your first name starts with a letter from K-Z inclusively:**

Use the *DoublyLinkedList* implementation of the textbook (week 2 lecture examples. Write a method for **concatenating two doubly linked lists L1 and L2**, into a single list L that contains all the nodes of L1 followed by all the nodes of L2. Write a main method to test the new method. **Hint**: Connect the end of L1 into the beginning of L2.

(3 marks)

**Exercise 3**

**If your first name starts with a letter from A-J inclusively:**

Implement the **clone() method** for the CircularlyLinkedList class. Make sure to properly link the new chain of nodes.

**If your first name starts with a letter from K-Z inclusively:**

Let L1 and L2 be two circularly linked lists created as objects of CircularlyLinkedList class from Lesson. Write a method that returns *true* if L1 and L2 store the same sequence of elements (but perhaps with different starting points). Write the main method to test the new method. **Hint**: Try to find a matching alignment for the first node of one list.

(3 marks)

**Evaluation:**

|  |  |
| --- | --- |
| **Functionality:**   * Correct implementation of requirements * Code demonstration and brief explanation in a short video | 70%  10% |
| **Object-Oriented design**:   * Correct design of classes and methods similarly to chapter 3 examples. * Correct use of generics * Correct use of naming guidelines for classes, variables, methods. | 15%  5% |
| **Total** | 100% |

**Naming and Submission Rules:**

You must **name your Eclipse project** according to the following rule:

**YourFullname\_COMP254Labnumber**. Example: **JohnSmith\_COMP254Lab1**

You must name package names **com.exercisenumber.yourfirstname.yourlastname**, for example: com.exercise1.john.smith

Provide your **student number and full name as a comment** at the top of main method for each exercise.

**Archive your project in a zip file** named according to the following rule:

**YourFullname\_COMP254Labnumber.zip**

Example: **JohnSmith\_COMP254Lab1.zip**

Upload the zip file on eCentennial using the Assignment link.