University of Windsor

SCHOOL OF COMPUTER SCIENCE
03-60-212 – OOP USING JAVA
SUMMER 2014
LAB – 5

[To get the **full marks**, complete and show your works to the Lab Instructor before the end of the Lab period.]

Objective:

Linked Data Structures

1. Write a ***Generic*** class called ***Node<T>*** with at least the following members:
   a. An instance variable of type ***T*** called data.
   b. An intnstance variable of tyoe ***Node<T>*** called ***next***.
   c. A two-arguments constructor to initialize the instance variables.
   d. ***Getter*** and ***Setter*** methods for the instance variables.

2. Write a class ***SortedList*** with at least the following members:
   a. An instance variable of type ***Node<Integer>*** called ***head***. (This will serve as the entry point or handle to access the sorted list.)
   b. An instance variable ***name*** of type ***String*** to serve as the name of the list.
   c. A one-argument constructor that receives a ***String*** for the name. The constructor initializes ***head*** to ***null***.
   d. ***Accessor*** methods for the instance variables.
   e. A method ***insert*** that accepts one argument of type ***int***, creates an node object of type ***Node<Integer>*** with the argument and inserts the node into its proper place within the list to **maintain a sorted list in ascending order**.
   f. A method ***printList*** that traverse through the list and prints data values of all the nodes.
   g. A method ***merge*** that can merge a ***SortedList*** it receives as an argument with the ***SortedList*** that calls the method.

3. Write an application that
   a. Creates two ***SortedList*** objects.
   b. Create and inserts 10 node objects into each of the lists, each node being created with a random integer between 0 and 100.
   c. Print all the numbers from the individual lists.
   d. Merge both the lists, and print all the numbers from the merged list.

Sample output:

```
List1: 0 9 15 35 47 63 66 73 82 93
List2: 1 5 10 13 43 56 58 61 83 97
After merging List2 with List1
List1: 0 1 5 9 10 13 15 35 43 47 56 58 61 63 66 73 82 83 93 97
```