ITI1121: SUMMER 2016
ASSIGNMENT #3

DUE DATE\TIME: JULY 25 @23:00

1) Implement an Interface **List** that contains all the methods mentioned below (1-18). The interface should use a generic type T that extends Comparable.
Note that the java interface Comparable as a method : int compareTo(T o) that compares this object with the specified object for order. Returns a negative integer, zero, or a positive integer as this object is less than, equal to, or greater than the specified object.

Deliverable: List.java (5 marks)

2) Implement a SinglyLinkedList class that implements the interface List to store elements of a generic type: (85 marks)
(i.e. : public class SinglyLinkedList <T extends Comparable<T>> implements List<T>).

Just as in the lectures, the class must have a nested static class called **Node** that stores the generic **value** and a reference variable **next** that points to the next node in the list.

Here are the methods included in the interface:

1) public int size(); returns the size of the list.

2) public T get(int i); returns the element at position i (first position index is 0).

3) public int indexOf(Object item); returns the position of item (note that it is of type objects).

4) public void add(int i, T item); adds item at position i of the list.

5) public T remove(int i); removes item at position i of the list.

6) public T min(); returns the item with the minimum value in the list;

7) public T max(); returns the item with the maximum value in the list;

8) public boolean Empty(): returns true if the list is empty.

9) public void addAthead(T item) : adds an “element at the head of the list

10) public void addAtEnd(T item): adds an element at the end of the list.

11) public void replace(T first, T second): takes two elements as an input, searches for the first and replaces it with the second.

12) public List<T> duplicate(T item): A recursive method to duplicate every element in a linked list, that is equal to a certain value. For example, if the linked list contains 4, 8, 4, 10 the element to duplicate is 4 the new list should contain 4,4,8,4,4, 10.

13) public void reverse(): A recursive method **that prints** out the data elements of a linked list in reverse order.

14) public List <T> countGreaterThan(T threshold): A recursive method that returns a list containing all the elements in the original list that are larger than threshold.

15) public Boolean equals(Object other): A recursive method returns true if other is a list that has the same elements in the list with the same order.

16) public String toString(): returns a string representation of the elements in the list separated by a comm..

17) public List<T> inorder(); returns a new list that has all the elements in the list sorted in an ascending order.

18) public void removeEven(): removes all the elements at even positions from the list (0,2, 4, ..) and keeps only the elements at the odd positions.

Deliverable : **SinglyLinkedList.java**

b) Write a test program that creates two linked lists List1 and List2 that store Integers. Call all the methods you implemented (in the order of your choice) to test your code. After each call to a method print out the list to see its contents. (10 marks).

Deliverable: Mytestclass.java

Note: all methods must take care of all illegal (e.g., removing an element from an empty list, inserting at an out of boundary position, etc) . For each of these cases an exception must be thrown. To learn about Exception take the following tutorial http://www.tutorialspoint.com/java/java_exceptions.htm and consult the provided lecture notes.