CS 340 Homework 1
Due Monday September 16
Homework 1

- Implement the data structure shown on the following slides. The data structure is a double linked list of T1 values and T2 values. A single node holds a T1 value and a T2 value. T1 values and T2 values are both instances of classes that implement (or inherit from a class that implements) Comparable. Each node exists on two lists. One list is sorted in ascending order by T1 value and the other list is sorted in descending order by T2 value. You must write a test driver to test your implementation. Your implementation of the list must not depend on your test driver because I will use my own test driver to test your program.
import java.io.*;
import java.util.*;

public class SortedLists<T1 extends Comparable<? super T1>, T2 extends Comparable<? super T2>> {
    //Implements a double linked lists of T1 and T2 values
    //The T1 values are sorted in ascending order (referred to below as the listT1)
    //The T2 values are sorted in descending order (referred to below as the listT2)
    //There is only one node for each data item (one T1 value and one T2 value)
    //A node contains one T1 value and one T2 value
    //The node is part of two lists.

    private class Node {
        private T1 data1;
        private T2 data2;
        private Node nextT1;
        private Node nextT2;
        private Node prevT1;
        private Node prevT2;
        private Node(T1 d1, T2 d2, Node n1, Node n2, Node p1, Node p2) {
            data1 = d1;
            data2 = d2;
            nextT1 = n1;
            nextT2 = n2;
            prevT1 = p1;
            prevT2 = p2;
        }
    }
}
private Node head1;
private Node head2;

public SortedLists() {
    // Sentinel nodes are not used
    head1 = null;
    head2 = null;
}

public void insert(T1 d1, T2 d2) {
    // A new element (d1,d2) is inserted so both listT1 and listT2 remain sorted
    // Only one new Node is created by this function
    // Duplicates are allowed
}
public void remove(T1 d1, T2 d2) {
    //remove all occurrences of elements that match (d1, d2)
}

public LinkedList<T2> findT2s(T1 d1) {
    //return a LinkedList of T2 values for each T1 value that matches d1
}

public T1[] findT1s(T2 d2) {
    //return an array of T1 values for each T2 value that matches d2
    //return null if no matches are found
}
public class T1Iterator implements Iterator<String> {
    //An iterator that returns data in the list in ascending T1 order
    //The format of the string returned by each call to next is (T1 value, T2 value)

    public T1Iterator() {
    }

    public boolean hasNext() {
    }

    public String next() {
    }

    public void remove() {  
        //optional method not implemented
    }

    public Iterator<String> T1_Order() {
        //return a new iterator T1Iterator object
    }
}
public class T2Iterator implements Iterator<String> {
    //An iterator that returns data in list in descending T2 order
    //The format of the string returned by each call to next is (T2 value, T1 value)

    public T2Iterator() {
    }

    public boolean hasNext() {
    }

    public String next() {
    }

    public void remove() {
        //optional method not implemented
    }

    public Iterator<String> T2_Order() {
        //return a new iterator T12Iterator object
    }
}
Homework 1 Submission

- At the top of the file include a comment that lists your name

- Add a comment for each private method or private instance variable you add

- Email me (tgendreau@uwlax.edu) only one file called SortedLists.java. Do not email me your test driver. The file must contain your implementation of SortedLists. If you use System.out.println to debug your program those lines should be put in a comment (or removed). Your debug output should not print when I test your program.