CS 340 Spring 2022
Programming Project 1
Due 11:59 PM Friday September 16
This project must be submitted on time.
Programming Project 1

• Implement the SortedStringList class shown on the following slides. See the discussion in class for more details about the structure of the list.

• The class implements a list of strings that are maintained in both ascending lexical order and descending length order. The list does not contain duplicate strings. The list could contain multiple strings of the same length. Strings with the same length must be ordered in descending lexical order. See the example on the next slide. For each string in the list there must only be one node.

• For this assignment two strings match if the contents of the strings match

• You can add private methods and instance variables as needed.

• Include your name in a comment at the top of the file
Programming Project 1 (Insert Richard)

head[0] -> Baker -> Davis -> Faut -> Hickson -> Kamenshek -> Kurys -> Lessing

head[1] -> Wisniewski -> Sams -> Perlick -> Nesbitt

next[0] references shown in black
next[1] references shown in red
import java.io.*;
import java.util.*;

public class SortedStringList {
    //Implements a list of strings
    //the strings can be accessed in ascending lexical order
    //or descending length order
    //there is only 1 node for each string

    private class Node {
        private String data;
        private Node next[]; //next[0] is the next reference for ascending lexical order
        //next[1] is the next reference for descending length order
        
        private Node(String s, Node aN, Node dN) {
            data = s;
            next = new Node[2];
            next[0] = aN;
            next[0] = dN;
        }
    }
Programming Project 1

private Node heads[]; //heads[0] is the head of the ascending lexical order list
  //heads[1] is the head of the descending length order list

public SortedIntList() {
    heads = new Node[2]; //no sentinel nodes
}
public void insert(String s) {
    //if a string matching s is not in the list, insert s into the list maintaining the
    //ascending lexical and descending length orders
    //otherwise no changes to the list are made
}

public void remove(String s) {
    //if a string matching s is in the list, remove s from the list maintaining the
    //ascending lexical and descending length orders
    //otherwise no changes to the list are made
}
public void ascPrint() {
    // print a comma delimited list of the strings in ascending lexical order
}

public void descPrint() {
    // print a comma delimited list of the strings in descending length order
}
Programming Project 1 Submission

• Upload one zip file called p1.zip to Canvas. The zip file must contain **only one file called SortedStringList.java**. Do not upload your whole Eclipse project!

• You have to write code to test your program but I will write my own test driver. Do not upload your test driver code.