CSC 1302: PRINCIPLES OF COMPUTER PROGRAMMING II
Lab 13

How to Submit
Please submit your answers to the lab instructor once you have completed.
Failure to submit will result in a **ZERO FOR THIS LAB. NO EXCEPTIONS**.

1. Execute the following PushCounter class and PushCounterPanel class.

```
import javax.swing.JFrame;
public class PushCounter
{
//-----------------------------------------------------------------
//  Creates and displays the main program frame.
//-----------------------------------------------------------------
      public static void main(String[] args)
      {
            JFrame frame = new JFrame("Push Counter");
            frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

            PushCounterPanel panel = new PushCounterPanel();
            frame.getContentPane().add(panel);

            frame.pack();
            frame.setVisible(true);
      }
} //end of PushCounter.java


import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class PushCounterPanel extends JPanel
{
   private int count;
   private JButton push;
   private JLabel label;

   //-----------------------------------------------------------------
   //  Constructor: Sets up the GUI.
   //-----------------------------------------------------------------
   public PushCounterPanel()
   {
      count = 0;

      push = new JButton("Push Me!");
      label = new JLabel();
      push.addActionListener(new ButtonListener());
      add(push);
      add(label);

      setBackground(Color.cyan);
      setPreferredSize(new Dimension(300, 40));
   }
```

```
   //-----------------------------------------------------------------
   //  Represents a listener for button push (action) events.
   //-----------------------------------------------------------------
      private class ButtonListener implements ActionListener
      {
            public void actionPerformed(ActionEvent event)
            {
                  count++;
                  label.setText("Pushes: " + count);
            }
      }
} //end of PushCounterPanel.java
```

2. Modify the above application such that every time the button is pushed, the label displays a random number between 1 and 100.
3. Write an application that presents two buttons and a label to the user (example: copy the ***PushCounterPanel.java*** code and call it ***'PushCounterPanelMath.java'***) The display text in the buttons should be: **Increment** and **Decrement** respectively. Display a numeric value (initially 50) using the label. Each time the **Increment** button is pushed, increment the value displayed (in the label) by a value of one. Likewise, each time the **Decrement** button is pressed decrement the value displayed by a value of one.
4. Design and implement an application that has 5 text boxes, a button (make its text **SORT**) and a label. (example: copy the ***PushCounterPanel.java*** code and call it ***'PushCounterPanelSorter.java'***) The user can enter 5 different numbers using the 5 text boxes. When the user clicks the **SORT** button, sort the 5 numbers using any of the sorting algorithms we learned in class and display the numbers separated by comma using the label for showing the output: -a sorted list.