CSC108H lab – week 11

This document contains the instructions for the week 11 CSC108H lab. To earn your lab mark, you must actively participate in the lab. *You don't need to finish in the time alloted, you just need to try hard.*

In this lab, you will verify your A4 partner's information, and work with `Vectors` of `JButtons`. Remember to typecast as you get the buttons out of the `Vector`.

1 Arrays of `JButtons`

Back in lab week 7 you wrote code to change the text of a `JButton` when it was clicked. Here is the solution to that lab:

`http://www.cs.toronto.edu/~pgries/108/02f/labs/lab-week7/ButtonTAJFrame.java`

Modify the class so that instead of a single `JButton` instance variable it has a `Vector` of `JButtons`. Leave the text area and click count as they are.

The constructor should take an `int` as a parameter and fill the `Vector` with that number of `JButtons`. Give each button the name `"Button i"`, where `i` is unique for each button. Also, tell each button that `this` is an action listener. You will, of course, need a loop to do all that processing.

Last, the constructor should add the first `JButton` in the north, and the text area in the center.

Modify `actionPerformed` so that it does not change the text of the button that was clicked. It should still modify the click count message in the text area.

Compile it and test it from the Interactions pane. You won't be able to test whether you've done the `Vector` stuff right yet, but you will be able to soon.

2 actionPerformed

Switch roles: the navigator becomes the driver and the driver becomes the navigator.

In method `actionPerformed`, you can test to see which button was clicked. It looks like this:

```
JButton b= (JButton) e.getSource();
if (b == v.get(0)) {
  // The first button was clicked.
}
```

You can, of course, have `else ifs` to tell whether other buttons were clicked. Notice that `==` is used, rather than `.equals`. Why is this okay?

Add code to `actionPerformed` to add `"First"` or `"Last"` at the end of the text area when the first or last buttons in the `Vector` or clicked.

Compile it and test it from the Interactions pane. (You can't test whether the last button feature works yet, because you can't yet see it. See the next section.)

3 JPanels

Switch roles: the navigator becomes the driver and the driver becomes the navigator.

Each of the five areas of the content pane can only hold one component. But a `JPanel` is a component that can hold many components –if you put a bunch of buttons into a `JPanel`, and add the `JPanel` to the content pane, you'll see all the buttons.

Modify the constructor as follows: declare and instantiate a `JPanel` local variable, and modify the loop so that each button gets `added` to the `JPanel`. For `JPanels`, `add` takes only the component being added, not the location.

Add the `JPanel` in the north instead of adding the first button.

Compile it and test it from the Interactions pane. Make sure that all the parts of this lab are working properly.