

## CSC171 — Homework 9

### Graphical User Interfaces (GUIs)

The goal of this assignment is to give you experience with graphical user interfaces and the Swing GUI toolkit. For these questions, you should create a graphical application by defining a class that extends `javax.swing.JFrame`. This class should initialize the user interface components in its constructor:

1. Set the layout for the frame's content, for example:

```
setLayout(new FlowLayout());
```

If you don't do this, your controls will probably position themselves strangely.

2. Create your controls, configure them, setup any event listeners, and add them to the frame.
3. Finally, set the frame's size and default window closing behavior. You would usually not make the frame visible in the constructor, but rather leave that to the code that created the frame.

You can use a `main` method in the class itself to create an instance and display it (make it visible), or have a separate “test” class to do that. By the end of these questions, you should have one application with a label and two buttons. The label keeps count of total button clicks, and the buttons can be independently turned “On” or “Off”. You should only need a single `.java` file.

A useful reference for GUIs (in addition to the textbook) is the [Java Tutorial Trail: Creating a GUI with JFC/Swing](#).

### Questions

1. Create an application with a GUI containing a `JButton` which you save in an instance variable. Make your application class implement the `java.awt.event.ActionListener` interface and register itself as an `ActionListener` for the button. Your `actionPerformed` handler can simply print the event for now.
2. Add a `JLabel` to your GUI. Keep track of the number of times the button has been pressed in your application class and update the label to show the count on each button press. Hint: Check out the `Integer.toString` and/or `String.format` methods.

3. Add another `JButton` with text label “Off” to your GUI and register the application class as an `ActionListener`. Modify the code for the event handler so that when the second button is clicked it changes the text of the button itself “Off” to “On” or vice-versa. Hint: Use `EventObject.getSource()` to find out which object generated the event, and compare it to your instance variables to figure out what to do.

## Grading Scheme

Equal weight for each part.

Doesn't compile or is trivial	< 50%
Compiles and is non-trivial	≥ 50%
Complete and correct with good style and comments	100%
Incomplete, incorrect, bad style, no comments	< 100%

## Submission Requirements

Your submission **MUST** include a file named “`README.txt`” with your name, your NetID, the assignment number, and your lab section. This file should explain anything we need to know about how to build and run your project. In particular, be sure to explain how to run what parts of your submission for each question in the assignment.

Submit your solution as a single ZIP archive to BlackBoard before the deadline.

Mildly late homework will be accepted with a penalty. Significantly late work will not be accepted.

All assignments and activities associated with this course must be performed in accordance with the University of Rochester's Academic Honesty Policy.