Lab 13

Goals:

- Gain experience with Swing.

Using the Konqueror browser, go to the course web-page: http://www.cs.uni.edu/~fienup/cs062s06 and click on the Lab13.zip link and "Save to disk" in the cs2 subdirectory created in lab 1. In a shell window, change to the cs2 subdirectory. Use the "ls" command to list the files in the directory and look for the file "Lab13.zip". Decompress the "Lab13.zip" using the command "unzip Lab13.zip", then change to the Lab13 subdirectory.

Compile and run `PinBallGameDriver.java`. Your task in lab today is to convert the pinball game to run using the Swing GUI component instead of the AWT components. Recall from yesterday's lecture that the general steps for this conversion are:

- Add import statements for javax.swing.* and javax.swing.event.*
- Corresponding Swing components are the same name as AWT, except:
  - 'J' prepended to the name: JButton, JPane, etc.
  - Scroll**b**ar in AWT became JScroll**B**ar in Swing
- Swing requires you to invoke the getContentPane() method before adding the component using the add() method. For example,

```
public class CannonWorld extends JFrame {
  …
  public CannonWorld () {
        …
        getContentPane().add("East", scrollbar);
    }
  …
}
```

- Swing requires you to invoke the parent class `paint` method from inside an overridden `paint` method.

```
public class BallWorld extends JFrame {
  …
  public void paint (Graphics g) {
        super.paint(g); // first call parent, then do
        aBall.paint(g); // class-specific painting
    }
  …
}
```

- Swing JFrame's should not be directly drawn on. Instead, draw the graphics in a JPanel and embed it in the JFrame.

Near the end of the lab, raise your hand and demo your best attempt to complete Lab 13 to the TA (Patrick) or instructor.