

Eclipse & Silver Dollar Game

Wednesday, January 30, 2013

Lab 1
CSC 062: Spring, 2013

Eclipse

This laboratory is intended to get you started with the programs and systems in the lab, and to give you an opportunity to complete your first substantial program for the class. Be sure to read the description of the Silver Dollar Game on the next page before coming to class, and think about how you would solve it. The material on Setting up Eclipse is best read while sitting in lab.

Even though many of you have used Eclipse before, please follow the introductory steps below carefully. Correcting configuration errors is not impossible, but it is time-consuming and distracting. Before leaving the lab session, please practice turning in your code so that you will know how to do it with the regular assignment.

- 1. Setting things up:** You should all have your accounts and passwords. Log into one of the computers in the classroom. Among the icons in the dock at the bottom of the screen are four that will be important for the class.



Terminal



Eclipse



Safari



Emacs

You will use Terminal and Eclipse today. Safari is a web browser (try it out!), through which you can read the course pages and documentation. Emacs is an editor that we will use later in the course.

For now, open Safari and navigate to <http://www.cs.pomona.edu/classes/cs062/csc062.html>. I suggest that you add it to the “Bookmarks bar”, if it isn’t already, so that it will be easily accessible to you.

Open a terminal window and type the following commands. They will create a (private) folder for this class and a workspace within that folder.

```
mkdir mycs062
chmod 700 mycs062
cd mycs062
mkdir workspace
```

Explanation: The command `mkdir` makes a directory, and `cd` changes to that directory. The `chmod` command makes the directory private, so that only you can see its contents. A few other commands for your information: `ls` lists the contents of the current directory, `cd ..` changes the directory to one directory higher, `rm` removes a specified file and you can type `man <command>` for any command and it will give you more information (short for “manual”).

Note that much of this could have been accomplished using the mouse and interacting with menus. However, one goal for today is to get you used to using the terminal to accomplish tasks. All of the commands you are typing into the terminal window are UNIX commands, as the Macintosh operating system is a variant of UNIX (BSD UNIX, to be exact). These same commands will generally work with other flavors of UNIX, like LINUX. However, Windows computers use a different operating system, with slightly different commands to accomplish the same tasks.

Now start Eclipse. When it asks you for a workspace, select Browse, find the `mycs062/workspace` folder you just created, and click OK. With Eclipse still as the frontmost application, go to the menu at the top titled Eclipse and select Preferences. Then in the Preferences window, expand Java and then Build Path, and then select Classpath Variables. Click on New and enter the name BAILEY and the path

```
/common/cs/cs062/bailey.jar
```

(or navigate to that file using your mouse, and select it.) These steps need only be done once.

Next, make sure that Eclipse is setup to Java 7, which we will require for several of our projects. With the Eclipse preferences screen still open, click on the **Compiler** subheading under the **Java** option. Make sure the "Compiler compliance level" is set to "1.7" and there is a check mark next to "Use default compliance settings." Finally, click on "Installed JREs" and make sure either "Java 7" or "Java 1.7.*" is selected (where the * may be any subversion number). Click "OK" to exit the Preferences screen.

2. **Adding libraries:** Eclipse works in units called "projects." There will be one for each laboratory and each assignment in the class. To start a project, select **File/New/Java Project**. Give your project a name, like **Lab1**, but *do not click Finish yet!* Click **Next**, then **Libraries**, and then **Add Variable**. Add the variable **BAILEY**, and then click **OK** and **Finish**.

This step adds the bailey library (which we just setup) to our project so that our project knows that code is available. This allows us to use the `import` command in our code to bring outside resources (i.e. resources that are not included with java). For many of the labs, you will need to add the **BAILEY** variable to your project.

Silver Dollar Game

1. Your program for lab today is to write a text-based version of the Silver Dollar Game. See Section 3.10 of *Java Structures* for a description of the game. Make sure you understand how the game is played before coding it up! Think about the methods described there and choose an appropriate data structure to model the game. Do not hesitate to ask questions or share your thoughts.

The behavior, in the console, will look like this:

```
_o___oo_o_ Next move? 6 4
 oo___o_o_ Next move? 2 2
Illegal move!
 oo___o_o_ Next move? 1 1
o_o___o_o_ Next move?
...
ooooo_____ You win!!
```

The pair of numbers after "Next move?" signifies the location of a coin and the number of squares that the coin is to move to the left. Remember that in languages like Java/C/C++ we start counting at 0 not 1, so the leftmost square is location 0, not 1.

For this first exercise, we have given you a start on the code for the class `TextCoinStrip`. It can be found at `/common/cs/cs062/labs/lab1/`. To get it in to your directory, copy the file `TextCoinStrip.java` into your `src` directory. To do this, type:

```
cp /common/cs/cs062/labs/lab1/TextCoinStrip.java ~/mycs062/workspace/Lab1/src/
(wherelab1" in the destination of the copy should be replaced by the name of your project for this lab). Then, go in to Eclipse and "ctrl+click" on the project and select "Refresh". You should now see the starting code. (By the way, the ~ is an abbreviation for your home directory, and will be used a lot.)
```

Take some time to understand the code (I promise you'll be much better off spending 5-10 minutes looking at the code before doing any coding yourself), then fill in the four missing methods.

- `toString` which creates the string representation of the strip,
- `isLegalMove` which determines if a move is legal,
- `makeMove` which makes a (legal) move, and
- `gameIsOver` which determines if the game is completed.

Submitting Your Work

For practice, we would like you to submit your project before you leave the lab today. You will receive full credit if you successfully turn in a file, so it doesn't matter if it has errors. The main thing here is for you to learn how to turn in programs. You can turn them in using UNIX commands, but it will be just as easy for us to use Eclipse to export your project and then drag it into the dropoff folder.

To set up your desktop to turn in your homework, we are going to set up a "soft" link to the dropoff folder on the file server.

Open up a terminal window. Inside the window type:

```
ln -s /common/cs/cs062 ~/Desktop/
```

This sets up a "soft link" between the cs062 folder on the server and a new folder with the same name on your desktop. By double clicking on the folder, you will get direct access to all of the files and folders in the corresponding folder on the server.

This need only be done once. As long as you don't accidentally drag it somewhere else, it will be there every time you log in.

The next part will have to be repeated every time you want to turn in a program.

- From within Eclipse, select "Export" from the "File" menu.
- Click on the triangle next to "General" in the dialog box. Select "File system" and click next.
- Make sure all the files in the dialog on the right hand side are checked, and then click the "Browse" button next to the "To directory:" entry
- Select "Desktop" on the left of the window and click on "Open". Then click "Finish."
- This should create a new folder on the desktop. You should now rename that folder by clicking on it and pressing return, and renaming it to "Lab1-lastname" where *lastname* is your last name. Note that dashes in names are OK, but don't include spaces.
- Quit Eclipse.
- Now open the "cs062" folder icon on the desktop by double-clicking on it. Within the "cs062" folder you should see a "dropbox" folder.
- Drag the folder you just created into the dropbox folder. When you do this, the computer may warn you that you will not be able to look at this folder. That is fine. Just click "OK".
- You can now drag the folder on your desktop into the trash. Do save the original folder in your workspace in case you need to access your program again. (So only the folder you "exported" should be thrown away.)

2. Extension: This is a suggestion for those who want to go beyond today's exercise; it is not a course requirement. Let the computer assume the role of one player and experiment with strategies for the Silver Dollar Game. You may conduct your experiments in either the text-based or the graphical world. Here are some simple strategies:

- Move the leftmost possible coin as far as possible.
- Move the rightmost possible coin one square.
- Find a coin capable of moving the farthest and move it the full distance.
- A "gap" is a sequence of unoccupied squares between two coins. If possible, make a move to increase the number of gaps.