

CSE 11  
Fall 2013

**START EARLY!**

**Program/Homework Assignment #1 (50 points total. This is a “pint-sized” assignment)**

**Due: 04 October 2013 at 11pm (2300)**

Covers Chapters: 1-2

This is a combination of written responses and a small program

Note that in all commands, the “\$” indicates the shell prompt. You do not type in the \$, only what comes afterwards. You hit “enter” or “return” to execute a command in the shell

Please note that: JAEA = “Java, An Eventful Approach”. A problem as JAEA: Exercise 1.5.2 means do exercise 1.5.2 in your textbook.

Exercises are optional, not graded.

**Exercise #1:** Make certain you can log into the campus lab computers in CSE building B230. All of your programming should be done in Linux. ALL PROGRAMS MUST BE TURNED IN FROM LAB MACHINES.

**Exercise #2:** If you have a Windows or Mac laptop and would like to have Linux environment for development, please go to the class website and look at the page called “Run a Development VM on your Laptop”

**Exercise #3:** If you do not know either of the emacs or vim text editors, then run through the tutorial called “vimtutor” in the lab

### **Homework Problems:**

The goal of this assignment is to get you comfortable in the Lab environment and do some very straightforward programs

Answers for Problems 1 – 5, should be placed in a file call `PR1.txt`. The top of file you have your student ID number, Name, Login similar to the following (replaces with Id, Name, and Login) . Make sure that ID, NAME, LOGIN appear on separate lines.

```
ID: A1111111
NAME: Philip Papadopoulos
LOGIN: cs11e
```

Make sure that answers to each problem are clearly labeled in your `PR1.txt` file. Something like

Problem 1:

<answers to problem 1>

Problem 2:

<answers to problem 2>

would be sufficient. Make it is easy for the Reader to see your answers.

### Problem #1: (10 points)

Create a subdirectory in your home area called HW1

```
$ cd $HOME
$ mkdir HW1
$ cd $HOME
```

Download HelloWorld.java, HelloWorldGraphics.java, Hello2.java and myapplet.html from the class web site (Lecture 2), and place them into your HW1 subdirectory.

Download objectdraw.jar and place it into your home directory. Make a copy of objectdraw.jar and place it in your HW1 subdirectory

Open each .java file and follow the instructions of how to compile and run these applications/applets.

After you have compiled and run each of the programs, Include the output of the following command as part of your homework.

```
$ ls -l $HOME/HW1
```

(Hint: `ls -l $HOME/HW1 >> $HOME/HW1/PR1.txt` will *append* the output of the command to the file PR1.txt, Note the “>>” and be careful not to use just “>”. The second one *overwrites* the files, which means you will lose your previous work.

Briefly, describe the difference between a java application and a java applet.

(Aside: Unix/Linux allows you define *environment* variables for configuration. The java CLASSPATH variable is a useful variable. If you edit \$HOME/.bashrc and place the following definition

```
export CLASSPATH=.:$HOME/objectdraw.jar
```

then any java program you compile/run that uses the objectdraw library will not require and explicit classpath definition when running java or javac. If you have downloaded the Virtual Machine (VM), you can see an example of this in the `~cs11xyz/.bashrc` file. The definition only becomes active when you login or create a new terminal.)

### Problem #2 (10 points)

In HelloWorld.java, introduce four different errors that will cause either the program to not compile or not run properly. For each error, show the line of code with the error, followed by the correct line of code. Below these, include the actual error generated by the compiler (javac) or runtime (java). Finally, for each error describe the error in plain english.

### Problem #3 (5 points)

JAEA: Exercise 1.9.1

Here is a sample class header:

```
public class Hangman extends WindowController
```

explain what is meant by the following words from the above line

- a. class
- b. Hangman
- c. extends
- d. WindowController

#### Problem #4 (10 Points)

Tell whether or not each of the following is a valid Java identifier. If it is not a valid identifier, describe why.

- a) aaaa
- b) 3rdAndVine
- c) Arnold\_is\_the\_terminator
- d) \$there
- e) oneANDdone
- f) whyIsn'tItStillSummer
- g) OLIPHANT
- h) x
- i) Elmer-Fudd
- j) i\_like\_"quotes"

#### Problem #5 (5 points)

define the following colors in terms of RGB, e.g. Blue = Color(0,0,255)

- a. black
- b. white
- c. purple
- d. yellow
- e. navy blue

Suggestion: use the program you will write in #6, and use the setColor() method with each of the above colors to verify your answers

#### problem #6 (10 points)

Write a java program (not an applet) called Fred.java. Fred should do the following

1. create a 400x400 applet window (see examples from class)
2. When you *press* the mouse, the following two objects are drawn
  1. a 30x30 filled oval of color blue is created where the mouse is clicked
  2. A line 45 pixels in length extends along the x-axis where the mouse was clicked
  3. A line 45 pixels in length extends downwards (positive) along the y-axis where the mouse was clicked
3. When you *release* the mouse, the canvas should clear

Please put in comments at the top of your program, the same, ID, NAME, LOGIN that is in your PR1.txt file. This will make grading MUCH easier. You will lose points if you do not do this.

#### Turning in your Program

**YOU MUST BE ON THE LAB MACHINES FOR THIS TO WORK. PLEASE VERIFY WELL BEFORE THE DEADLINE THAT YOU CAN TURNIN FILES**

You will be using the "bundleP1" program that will turn in the following two files

**PR1.txt**  
**Fred.java**

No other files will be turned in and then **must be named exactly as above**. BundleP1 uses the departments standard turnin program underneath.

To turn-in your program, you must be in the directory that has your source code and then you execute the following

```
$ /home/linux/ieng6/cs11e/public/bin/bundleP1
```

Sample output of running this program is as follows

```
[cs11e@ieng6-201]:HW1:30$  
/home/linux/ieng6/cs11e/public/bin/bundleP1  
Good; all required files are present:
```

```
Fred.java  
PR1.txt
```

```
Do you want to go ahead and turnin these files? [y/n]  
OK. Proceeding.
```

```
Performing turnin of approx. 3827 bytes (+/- 10%)  
Copying to /home/linux/ieng6/cs11e/turnin.dest/cs11e.P1  
..  
Done.  
Total bytes written: 3584  
Please check to be sure that's reasonable.  
Turnin successful.
```

You can turn in your program multiple times. The turnin program will ask you if you want to overwrite a previously-turned in project. **ONLY THE LAST TURNIN IS USED!**

**Suggestion:** if you have classes that compile and do some or most of what is specified, turn them in early. When you complete all the other aspects of the assignment, you can turn in newer (better) versions.

Don't forget to turn in your best version of the assignment.

### Frequently asked questions

**Can I do this as an applet?** No. Fred should be a java program.

**Do I really have to use the command line? I used Dr. Java in my high school class.** Yes! Although we will have graphics output on many programs, we are not using any particular development environment.

**START EARLY! ASK QUESTIONS!**