

CS112 - INTRODUCTION TO PROGRAMMING

Programming Project #2 - Due Wednesday, March 6, at the start of class

Purposes:

1. To give you experience writing a Java applet that accepts input from the user.
2. To give you experience with Java arithmetic and the use of the Math class.
3. To give you experience with drawing in Java using the methods of the Graphics class.

Introduction

This project is based on problem 5.16 in Wu, which is, in turn, based on Problem 4.14 . **(Please read both problem over carefully before continuing to read this assignment.)** However, instead of using a drawClock method in the applet as suggested in Wu, you will use the Clock class that you developed for Lab 4. Thus, your program will consist of a minimum of two objects: an applet and a clock. The constructor method for the applet should create the clock object, use add() to add it to itself, and set its bounds appropriately. The actionPerformed method of the applet, which is called when the user presses Enter after entering a time, will call the clock's setTime() method with the new values entered by the user. As was the case in Lab 4, the clock's setTime() method will include a call to repaint() that will cause it to be redrawn with the new setting.

The minimal requirement is that you solve the problem as stated in Wu, with the change in methodology discussed above. For maximal credit, you will be asked to make various improvements, as discussed below. You may enjoy looking at several applets the book author has written, available at: http://www.drcaffeine.com/applets/Clock_applets.html

After the projects are completed, your applet will be posted on the department's web server so that parents, friends, etc. can see what you've created.

Evaluation

Your grade on this project will be based on three criteria:

1. Correct, neat, and visually pleasing operation (maximum 40-60 points, depending on option chosen)
2. Good methodology, including documentation, use of comments, overall structure, choice of names, and good use of white space to aid readability (indentation and blank lines). (maximum 20 points)
3. A project quiz, to be given on the due date (maximum 20 points)

A blank project cover sheet is attached and should be stapled to the front of your project submission.

Requirements

Option 1 (Required - maximum 40 points for correct operation)

Fulfill the requirements as stated above. Your clock need not be fancy - a circle and two hands is enough. The remaining options presuppose that this option is fulfilled correctly.

Option 2 (Maximum of 10 additional points)

Improve the appearance of your clock in one or more of the following ways:

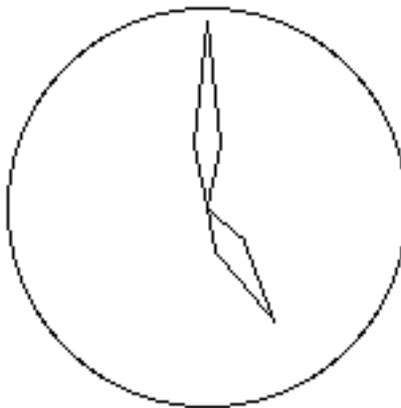
- Add numbers and/or markers around the face of the clock
- Improve the appearance of the clock hands. We'll discuss possibilities for this in class.
- Use color to make the clock more visually appealing.
- Other creative possibilities of your own devising

Credit will be based on the aesthetics of the finished product. It is more beneficial to do what you do well than to do lots of things sloppily. Maximum credit will require demonstrating that you understand the formulas for positioning hands, markers etc. and that you have learned something beyond what we have explicitly covered in class or lab.

Option 3 (Maximum of 10 additional points)

North America encompasses a total of 8 time zones - listed here in order from east to west: Newfoundland, Atlantic, Eastern, Central, Mountain, Pacific, Alaska, and Hawaii. Information on converting times between these can be found at <http://atm.geo.nsf.gov/ieis/time.html> (all but Newfoundland) and http://www.education.gov.nf.ca/liv_rel_time.htm (Newfoundland Time). Modify your applet as follows:

- Instead of displaying one clock, display eight clocks - each showing the time in a different time zone. Each clock should have a label beneath it naming the time zone.
- One clock (Eastern Standard or the time zone of your home) should be displayed noticeably bigger than the other clocks (e.g. perhaps twice as big). The user will input the time in this zone, and the 8 clocks will each be set to the appropriate value. (Be sure your labelling of the input fields makes it clear to the user what zone the input is supposed to be in.). You don't need to worry about daylight savings time - after all, it's still winter!
- In addition to displaying the time by the position of the hands on the clock face, each clock should also display the time digitally. (This can be combined with the label naming the time zone.) The following illustrates what this might look like:



5:00 Hawaii

If you do this option, you'll need to take into account issues like the fact that the time 1 hour before 1 is 12, and the fact that Newfoundland time differs from EST by 1 hours 30 minutes. Also, give some time to making your overall screen layout aesthetically pleasing.

Implementation Notes:

1. Use absolute positioning to lay out your applet - see pp. 211-212 in Wu.
2. Since newly created clocks display 12:00, you should set the initial values in your two input fields to 12 and 00. The constructor for a TextField allows you to specify a String parameter that contains the initial value for the field. If you do the multiple clocks option, each clock should be set to show the correct initial time relative to 12:00 EST by calling its setTime() method with an appropriate value at start up.
3. Regardless of which option you do, your applet class should be called Project2, and you should create an html file called Project2.html to run it. All files should reside in a folder called Project2.
4. A demonstration of the *functionality* you are expected to achieve if you do all 3 options is accessible from the course web page. Please note that this example does not use color or other features to produce good aesthetics - it simply behaves correctly. If you go for maximal credit, your project should be both functional and aesthetic. (But a beautiful program that does not work correctly is not a beautiful thing - functionality comes first, then aesthetics!)

Turn in the following, neatly stapled in the order listed:

1. Project Coversheet (attached)
2. Documentation including Problem Statement, Overall Plan, Design Document, and Object Diagram. (Note: you did similar documents in Lab 4, but the ones you turn in with the project should reflect the requirements of this project, not the lab.)
3. Printout of the java sources for the classes you created, and of your html file. Note that - in contrast with project 1 - you are just to turn in a single program, reflecting the highest option you did.

Leave on server:

Source and compiled versions of the program you turned in, plus html file to run it.

CS112 - INTRODUCTION TO PROGRAMMING - PROJECT TWO

Option(s) - check all that apply: 1 2 3 Author _____

I. Correct Operation / Clear and Neat Output

Points _____

II. Methodology

1.External documentation exhibits clear understanding of the problem and demonstrates good planning for solving it.

Definitely 3 Mostly 2 Partially 1 Not at all 0

2.Prologue comments for files, classes and methods make the purpose of each item clear.

Definitely 3 Mostly 2 Partially 1 Not at all 0

3.Indentifiers (class, method, and variable names) clearly describe the item they name and follow OO naming conventions.

Definitely 3 Mostly 2 Partially 1 Not at all 0

4.Features of the Java language and class library are used appropriately and efficiently.

Definitely 3 Mostly 2 Partially 1 Not at all 0

5.Whitespace (indentation and blank lines) follows a consistent convention and makes the overall structure of the program clear by enhancing its readability.

Definitely 3 Mostly 2 Partially 1 Not at all 0

6.Overall methodology results in a program that is easily understandable and obviously correct

Definitely 5 Mostly 3 Partially or not at all 0

Points _____

III. Quiz

Points _____

OVERALL TOTAL (Max 100)

Points _____