

## Lab 0: Day of the Week

Due 11:59 pm Mon, Feb 11, 2008

### 1 Assignment

Complete the “The Day of the Week Calculator” from *Java Structures*, Chapter 1.11 pages 29—31. Come to your lab session already having read the assignment and prepared to work on it. We will go through the first steps of using the Mac OS/Unix environment together in lab and then you will begin the assignment. You will probably need additional time outside of the scheduled lab to complete the assignment.

As for every assignment in CS136: Use Javadoc comments for all classes and methods. Use `structure5.Assert` pre-, post-, and general assertions wherever merited to detect corrupt state or illegal arguments. Make methods and fields private/protected and/or static wherever merited.

When complete, clean up your directory (see the `clean136` script) and then submit your solution using the command:

```
turnin -c 136 -d coin
```

which you must execute from the parent directory of “coin” (*caveat: there may be updated submission instructions in lab*). For this lab, your solution should be a single file named “Date.java”. Do not submit files ending in “.class” or “~”.

Do not submit written answers to the thought questions (do think about them, though...this is fair game for a future quiz or exam!)

### 2 Evaluation

The grading guidelines for this assignment are:

<b>Correctness</b>	<b>10</b>
<b>Readability</b>	<b>15</b>
Comments	2
Whitespace	2
Good variable names	3
Helper methods	5
Other style (e.g., private)	3
<b>Efficiency</b>	<b>5</b>
<b>Total</b>	<b>30</b>

You will often encounter situations where there are multiple ways of implementing a feature. Maybe recursion is more readable for a particular case, but less efficient than iteration. Perhaps using a long variable name eliminates the need for a comment on that variable—or perhaps a short variable name is in order because it will be used many times in a complex equation.

These are interesting decisions with many tradeoffs. As a programmer, you’re the one who will make the decision about what the best way is in each case. Programming is not mechanical. Writing good code requires judgment, knowledge, and creativity. In CS136 you have a lot more freedom (and responsibility) with design than in previous science courses you may have taken. Take a moment to enjoy that freedom—that’s what makes programming and computer science fun!

### 3 Suggestions

If your solution is not between 100 and 200 lines of code that might be a sign that something is wrong with it. My solution is about 160 lines of code and comments. I stored several tables of data in String and int arrays to cut down on the code size and make it more efficient. My longest method (except for `main()`) was about ten lines and contained many blank lines.