Arizona State University
Lecturer Acuña

Multiple Classes
Revised 10/08/2018

File Submission Standards

When preparing your files for final submission, please follow these rules:

1. Do not submit your project files. Submissions should only contain source files (e.g., .java, .c, etc.)
2. The descriptive part of programs must be written in proper English. Do not use other languages for naming identifiers, writing comments, etc. Please try to use correct grammar. Correct capitalization is optional. Given that this is not a writing class, we do not expect perfect English but do need to see you making an effort to be professional.
3. Add the following comment header to the beginning of each file. The header includes a basic description of the program, how long it took to write, author names, and a version (typically 1.0). Don't put in the parentheses shown below, they are only meant to show where something should go.

```
/**
 * (basic description of the program or class)
 *
 * Completion time: (estimation of hours spent on this program)
 *
 * @author (your name), (anyone else, e.g., Acuna, whose code you used)
 * @version (a version number or a date)
 */
```

   You should estimate the completion time of your program as the sum of your time spent understanding the assignment, programming the answer, and any review the textbook/slides that was needed. (Time spent on short answer questions should not be included in this total.) Adding the header lets us (and you) more easily identify the contents of the file. Remember that by the end of the semester, you will have hundreds of files! (and we will have tens of thousands...)

4. Try to keep the number of characters on a line limited to 80. In the following example, the line consisting only of numbers shows how "wide" 80 characters will be. The next two lines are shorter than 80 characters are fine. The third is too long, around 85 characters. In some circumstances, you may have lines that close to 80 characters (less than 90) - those are fine too. Just make a effort to ensure lines are reasonably short.

```
/**
The next line is exactly 80 characters:
12345678901234567890123456789012345678901234567890123456789012345678901234567890

This line is fine.
This line is a little longer but still fine.
This line is very very very very very very long and is now longer than 80 characters.
*/
```

   The reason we want to do this is so the code will be easier to read. Notice how annoying seeing the text run into the margin is here! Long lines means we will have to scroll not just up and down, but also side to side. On large monitors it is also useful because the files will be small enough that we can view multiple ones side by side. **The line width requirement is optional.**