

## COMP1006/1406 – Summer 2016

You need to complete and pass this assignment before you are allowed to submit any of the remaining assignments in this class.  
This assignment has 0 marks.

### 0: Academic Regulations [0 marks]

Read the Carleton University Academic Integrity Policy.

(<http://carleton.ca/senate/wp-content/uploads/Academic-Integrity-Policy1.pdf>)

In particular, be sure to read section VI (Academic Integrity Standards). After you read this, complete the Assignment 0 quiz on cuLearn.

### 1: Style [0 marks]

It is essential that your programs be human-readable. The Java compiler does not care how your code looks or if you have comments explaining tricky or unusual blocks of code. At some time, however, a human will be looking at your code. That person might be the TA marking your code, fellow workers that need to use or maintain your code, or even yourself when you look back at the code you have written. As such, your code must be easy to read.

Read the Programming Style Guide note that is posted on the course webpage. In Assignment 1, you will receive Style marks for your submitted code for following this guide. In Assignment 2 and onwards, you will be penalized marks for not following this guide.

### 2: Assignment Grading [0 marks]

A common mistake is to submit a `.rar` file instead of a `.zip` file. Another common mistake is to submit an assignment with a different name than specified in the assignment. It is crucial that you follow the specifications as outlined in the assignment specifications. There will be severe penalties for violating these specifications.

For example, submitting a `.rar` file will result in an automatic zero grade for the assignment. Submitting a single zip file other than `assignmentXX.zip`, where XX is the assignment number, will result in a 25% deduction for the assignment.

In order to receive any correctness marks for a given problem, your `.java` file(s) must compile without any errors using Java 8. Correctness marks for each problem are based solely on testing your code with different test cases and comparing the output (or behaviour) with the expected output (or behaviour). Any Style or Design marks will be given regardless of correctness (provided they correspond to an actual solution).