

## CPSC 233 – Team Assignment 1 – Requirements

Update the GuessingGame class from your individual assignment 1\_3 such that:

1. The user is reprompted in getAndCheckGuess if the user enters an invalid guess. The user should first see the same error message and then is reprompted. Use a while loop to accomplish this.
2. Update the main method to call getAndCheckGuess five times. In other words, the user gets five tries to guess the number. After the five tries or after the user guesses the number (whichever comes first) the program should end with one of the following two messages: 'Well done!' or 'The number to guess was <numToGuess>'. Use a for loop to implement this.

Go to the D2L page for this assignment to get an updated unit test. Note that the getAndCheckGuess tests are updated but there are no tests for the main method. This will be tested manually.

### Rubric

	2 marks	1 mark	0 marks
Functionality	All tests pass. When running the programming, user can enter up to five guesses to guess the number. All messages displayed are correct and as required.	All tests pass except possibly the error checking messages or the user can't guess multiple times.	Program does not compile or does not have minimal functionality.
Design	Good use of if statements, while loops and for loops. A while loop was used for re-prompting and a for loop was used to allow multiple guesses.	Good use of conditionals and loops but loops were not as required by assignment.	Design of code was overly complex.
Legibility	Good use of white space, self-documenting variables names. Statements and code overall is easy to read.	Some improvement possible but overall code is legible.	Code is difficult to read.
In-line documentation	Blocks of code are documented to explain what the code is intended to accomplish.	Each line of code is documented (possibly because it is difficult to read) or some blocks of code that could have used documentation don't have any.	No in-line documentation.