<u>Module 3 – Assignment – GuessTheNumber</u>

In ICS3U, you created a GuessingGame using code that was mostly provided for you. Now you will start from the beginning and create a superior program from scratch, implementing more advanced concepts.

The basic GuessTheNumber game is a number guessing game played between the computer and one player. The algorithm follows:

1. Determine a secret number between 1 and 20
2. Prompt the player for a number between 1 and 20
3. Compare the player's number to the secret number.
4. If the player's number matches the secret number, then display a "You won!" message. Otherwise display a "Try Again." message.

a) Write the pseudocode for the game (use the algorithm as a guide, recall that pseudocode is made up of English-like instructions. Refer to the case study "WordGuess" for an example of pseudocode.

b) Create the GuessTheNumber application based on your pseudocode. The application output should look similar to:

```
Enter a number between 1 and 20: 14
Try again.
Enter a number between 1 and 20: 1
Try again.
Enter a number between 1 and 20: 10
You won!
```

c) Modify the algorithm to include a call to a method named giveHint(), which **returns** a String that is the hint. Display the hint from your main() method. The new output should look similar to:

```
Enter a number between 1 and 20: 14
Hint: try a lower number
Enter a number between 1 and 20: 1
Hint: try a higher number
Enter a number between 1 and 20: 10
You won!
```

d) Now modify your code.

e) Write down how the application was tested.

f) BONUS: In order to achieve a mark of level 4 on this project, you must <u>improve upon the basics</u> of this program in a significant way by correctly implementing many different concepts.

Reminders:

- A mark of level 4 will only be achieved if you use <u>many</u> of the programming concepts from this module and improve upon a basic implementation
- Project must be submitted to the dropbox, and your project folder name must be "Last Name, First Name – Mod 3 Asn – GuessTheNumber". Add your pseudocode and how you tested your project to the document "ICS4U – Activity Submission"

<u>Module 3 – Assignment – GuessTheNumber Rubric</u>

Categories	Level 1 (50 − 59%)	Level 2 (60 − 69%)	Level 3 (70 − 79%)	Level 4 (80 − 100%)
Pseudocode (weight: 1)	Pseudocode does not describe an accurate solution to the problem.	Pseudocode has serious errors or was missing major elements	Pseudocode had some errors or was missing elements/ had to many elements	Pseudocode was included and was also correct.
Program Testing (weight: 1)	Program testing was missing crucial elements.	Program testing was insufficient to conclude that the program runs properly	Program testing missed covered a considerable number of possible cases and was summarized in a logical way	Program testing was thorough, and summarized in a logical and succinct way
Program Submission (weight: 1)	Program was not submitted properly	Serious or multiple omissions with assignment submission (such as incorrectly placed in dropbox, incorrect naming convention, late)	Small omission with assignment submission (such as incorrectly placed in dropbox, incorrect naming convention, late)	Program was submitted to the dropbox correctly, using proper naming conventions, on time
Program Execution (weight: 2)	Program doesn't run properly, or the output has serious errors.	Program runs properly. Output has errors.	Program runs properly. Output is correct with only minor errors.	Program runs properly as is. The output is identical or superior to the sample provided.
Source code (weight: 4)	Coding conventions are missing or incomplete.	Coding conventions are followed with some attention to detail	Coding conventions are followed with considerable attention to detail	Coding conventions are followed with thorough attention to detail
Programming Concepts (weight: 5) Focus: methods	Few of the programming concepts from the unit are used properly	Some of the programming concepts from the unit are used properly	Most of the programming concepts from the unit are used properly	Many or all of the programming concepts from the unit are used properly to maximize the efficiency of the code