**CPSC1012 Core Portfolio 3 – Loops**

**Weight: 5% of your final mark**

**Core Portfolio #3 - Loops**

**Slot Machine Simulation**

A slot machine is a gambling device that the user inserts money into and then pulls a lever (or presses a button). The slot machine then displays a set of random images. If two or more of the images matches, the user wins an amount of money that the slot machine dispenses back to the user.

Create a new C# Console App project named as **CorePortfolio03-*YourFullName*** (eg: CorePortfolio03-CodeGuru) that simulates a slot machine. When the program runs, it should do the following:

* Ask the user to enter the amount of money to enter into the slot machine.
* Instead of displaying images, the program will random select a word from the following list: ***Cherries, Oranges, Plums, Bells, Melons, Bars***
* To select a word, the program can generate a random number in the range of 0 through 5. If the number is 0, the selected word is ***Cherries***; if the number is 1, the selected word is ***Oranges***; and so forth. The program should randomly select a word from this list three times and display all three of the words.
* If none of the randomly selected words match, the program will inform the user that he or she has won $0. If two of the words match, the program will inform the user the he or she has won two times the amount entered. If three of the words match, the program will inform the user that he or she has won three times the amount entered.
* The program will ask whether the wants to play again. If so, these steps are repeated. If not, the program displays the total amount of money entered into the slot machine, the total amount won, and the net gain total.
* Your program should be fool-proof, guarding against errors caused by the user entering an incorrect choice or non-numeric data.

--------------------------

| Slot Machine Simulator |

--------------------------

This program simulators a slot machine.

Enter the amount to deposit into the slot machine: 0

“0” is not a valid amount. Try again.

Enter the amount to deposit into the slot machine: -100

“-100” is not a valid amount. Try again.

Enter the amount to deposit into the slot machine: 100

Cherries Oranges Melons

No words match. You won $0

Do you want to play again (yes|no)? maybe

“maybe” is not a valid choice. Try again.

Do you want to play again (yes|no)? yes

Enter the amount to deposit into the slot machine: 200

Cherries Melons Melons

Two words match. You won 2x.

Do you want to play again (yes|no)? no

Total amount deposited: $300.00

Total amount won: $400.00

Net gain/loss total: $100.00

**Coding Requirements**

The following coding standards must be followed when developing your program:

* A C# comment block at the beginning of the source file describing the **purpose**, **input**, **process**, **output, author, last modified date** of the program.
* Write only one statement per line.
* Write only one declaration per line.
* Use camelCase for local variable names.
* Use PascalCase for constant variable names.
* If continuation lines are not indented automatically, indent them one tab stop (four spaces).
* Do NOT use the goto statement.
* There can only be one exit point for a loop, do not use the break statement inside a loop

Submission **Requirements**

* Submit a compressed (zip) copy of your Visual Studio 2019 project folder to Moodle on or before the due date.

**Marking Rubric**

|  |  |
| --- | --- |
| **Mark** | **Description** |
| 5 | Excellent – program passes all test cases and coding follows best practices and class standards |
| 4 | Very Good – program passes all test cases, but coding does not follow best practices and class standards |
| 3 | Acceptable – coded all the requirements and program produce the expected results for some of the test cases |
| 2 | Needs Work – coded all the requirements but program fails to produce expected results |
| 1 | Unsatisfactory – coded less than 50% of the requirements |
| 0 | Not done. |