CMPS4143: Contemporary Programming Languages
(Major) Programming Assignment 4 (Work with a Partner)
Due: Tuesday, Oct. 4, 2016 75 points

PURPOSE: To develop a windows application that implements an interactive game; to use a variety of GUI components; to use a message box; to write a program using more than one class; to use arrays and array methods; to use the Random class; to use a do-while loop; and to use the ? operator. *Remember you will be graded on applying the concepts learned in class.*

PROBLEM: "**Press Your Luck" Game**

The Game: This game allows three contestants to collect spins by answering trivia questions and then uses the spins on a game board to win cash and prizes. (The spaces on game board randomly jump around until the contestant hits a button.) The person who amasses the most in cash and prizes at the end of the game wins (prizes are associated with dollar values). The show was known for the "Whammy", a red cartoon creature with a high-pitched voice. Landing on any of the Whammy's spaces on the game board took away the contestant's money, accompanied by an animation that would show the Whammy taking the loot, but frequently being chased away, blown up, or otherwise humiliated in the process. To understand the game better, go to https://en.wikipedia.org/wiki/Press_Your_Luck and look under Format.

Here is an example of the scoreboard in the TV game. Note there are 18 spaces, which at the beginning of the spin are given random values. (See Board values on the wiki page.) During the spin, the values move around, until the contestant hits a button. At any time, the user could pass their spins to another contestant, rather than pressing their luck in getting a Whammy. That person had to use up any spins they were passed.

Implement as much of this game as you can. Of the 75 points, 25 will be for how much of the game you can implement – you will be guaranteed 15 points if at a minimum, you allow two contestants to complete one round (the Big Board round) to win the game (without passing spins). You get more points if you have 3 contestants, contestants can

play two rounds, spins can be passed and mandatorily used or you implement special squares (+5 points for each additional feature). DO GET THE MINIMUM REQUIREMENTS WORKING FIRST!

DESIGN HINTS: Utilize at least two classes – one containing the data structure(s) (could hold *values* of the spaces and trivia questions and answers), and the PressYourLuckGame Form (which controls the main flow of the game and contains the GUI components).

Data structure class(es): These need to hold the questions and answers read from the input file, keep track of the scores, etc. Methods include:

- A constructor to perform the following tasks:
    - Initialize the data structure(s)
    - Set counters/accumulators to zero.
- EvaluateAnswer – evaluates answer and provides the feedback reflecting whether answer is correct or not.

PressYourLuckGameForm - The form with the GUI – it controls the overflow of the game. It starts a new game, lets the contestants enter and submit answers, and will repeat until a contestant wins or quits the program. It also permits the user to stop a spin freezing the board. The form should present the answer if the contestant is wrong or indicate the player is correct. The form will need to communicate with the *DataStructure class* for evaluation and feedback of answers, so declare and instantiate the DataStructure class in *this* class.

INPUT: User should be able to

- Enter name of input file containing questions and answers. (I will supply you with file processing code – you'll have to store in array. A file called luckfile.txt is stored in the Y:/ folder on the computers in the lab. Everyone should add 5-10 questions to the file.)
- The contestant answering the question (1, 2 (or 3)).
- Enter an answer.
- Indicate whether they want to spin again or not
- Indicate whether they want to quit or to play another game (may need to be enable or disabled at appropriate times).

OUTPUT: Output is all done on a form. The form should display

- Instructions to play the game
- The current question
- Guessed and correct answers to the questions
- Current scores
- Button(s) to play a new game, quit, submit an answer, spin or pass. I'll give you some leeway, you can use and reuse one button or display both buttons.

TURN IN all materials in a 9x12 envelope:

1) Print out of documented Source code (class.cs file and the PressYourLuckForm.cs file)
2) Screen dumps of images when running
3) Project folder or application (.exe file) on some storage media. The application is in a debug folder.

EXTRA CREDIT OPTIONS:

- Incorporate animations for the Whammys (3 points)
- Incorporate sound effects (3 points)
- Implement the end game (where the winning contestant goes for the big prize). You can either make original components not visible and make new ones visible or use a new form – this involves using another class. (7 points)