CS424/524 Programming Assignment #1

DUE: April 6, 2017

Very short description:

Write a Python program that allows the user to play the two dice games Fifty and Pig.

More detailed description:

Your computer program should allow 2 or more players to play the dice games Fifty and Pig. The rules for each game are below. The program should continue executing until the user chooses to quit playing. After each game, the user should be allowed to choose which game to play next (or choose to quit).

You must implement 2 versions of the program.

- One version should use imperative constructs only, and must take advantage of procedural abstraction. In other words, your program must be broken down into an appropriate number of functions, instead of just being in one or two large functions.
- The second version should use object-oriented constructs. Therefore, your program must be organized using classes. You need to demonstrate inheritance with your program, as well as other object-oriented techniques.

The rules for Fifty are as follows:

Goal: The goal of Fifty is to be the first player to reach 50 points. You get points by rolling doubles.

Play: A turn consists of a player rolling a pair of dice (with the goal of rolling doubles), and scoring the roll as described below. Play continues with each player taking one roll per turn. The first player to score 50 or more points is declared the winner.

Scoring: All doubles except 3s and 6s score 5 points. Double 6s are worth 25 points. Double 3s wipe out the player's entire score, and the player must start again at 0. Non-double rolls are 0 points.

Note: As you are developing and testing your program, you might want to make the winning score something smaller than 50, such as 10. It can take a while to reach 50 points. Just be sure to change it back to 50 before turning in the program. Alternatively, you could allow the user to set the winning score.

The rules for Pig are as follows:

Goal: The goal of Pig is to be the first player to reach 100 points. You get points by rolling a single die multiple times and adding the value on each roll of the die to your current score.

Play: The first player rolls the die as many times as they want. The value of each throw is added

onto the score until the player decides to end his turn and passes the die to the next player. Play continues until one player reaches 100.

Scoring: The value of each throw is added to the current player's score. If the player rolls a 1, the player's score goes back to 0, and their turn ends.

At one extreme, any player who gets a 1 on the first roll is immediately out. At the other extreme, the first player could theoretically reach the winning score on the first turn, as long as they don't roll a 1. If the player succeeds, the game ends there.

Note: As you are developing and testing your program, you might want to make the winning score something smaller than 100. Just be sure to change it back to 100 before turning in the program. Alternatively, you may allow the user to set the winning score.

Program Requirements:

Your program must do the following

- Print an introductory message at the beginning of the program to tell the user what the program does.
- Allow the user to play games of "Fifty" with another player.
- Allow the user to play games of "Pig" with another player.
- When a player wins, the program should print a congratulatory message that includes which player won and the final score.
- Be sure any user interface elements are clear for the user.
- Your output should be clearly labeled and neatly formatted. Be sure to print helpful information, such as allowing the user to view the game rules before a game, and printing updated scores after each roll.
- Use good variable names and appropriate comments throughout your program. Make use of whitespace (blank lines, indenting, etc.) in order to make your program readable.
- You must have a comment block at the beginning of your program that lists your name, the date, the class number and section and a description of what your program does.

You must submit a printout of your program in addition to an electronic copy.

- The printout must be turned in during class.
- The electronic copy should be submitted via Canvas.
- Please review the rules regarding late programs as outlined in the syllabus
- Only turn in your .py files. Your files should be named **dicegames.py** (the imperative version) **and dicegames_oo.py** (the object-oriented version).