Project 9: Lottery Tickets

COSC 1423 | Fall 2023

Professor: Megan Avery

Topics: lists

Turn In Instructions: A zip file containing the following files:

Planning:

- ☐ A PDF named project09_planning_firstname_lastname.pdf that contains your planning, test cases, and followup questions

Implementation:

- ☐ A Python file named project09_firstname_lastname.py. Your name will be at the top of the file in a comment. Your followup questions will be answered in a multiline comment at the end of the file.

Planning (10 pts): Figure out how you are going to approach this problem. Take notes along the way.

Style (10 pts): See Style Guide

Implementation (60 pts):

Write a program to simulate the user playing the lottery. Each lottery ticket contains 7 numbers, each between 1 and 100 inclusive, and must match the numbers in the winning ticket to be considered a winner. The numbers don't have to be in the same order in both tickets.

- ☐ **Objective 1:** Create constants for the number of numbers in a ticket and the max/min allowed numbers on the ticket.

- ☐ **Objective 2:** In the main write a for loop that asks the user for their 7 lottery numbers, storing each number into a list along the way.

- ☐ **Objective 3:** Write an in_range(number) function that returns True if the number is a valid lottery number and false otherwise. Use this function in an input validation loop for the numbers entered by the user in Objective 3.

- ☐ **Objective 4:** Write a generate_winning_ticket() function that generates and returns a list of 7 random numbers between 1 and 100. This will simulate a winning ticket.

- ☐ **Objective 5:** Write a won_the_lottery(player_ticket) that calls generate_winning_ticket(), saves this winning ticket into a variable, and then compares the winning ticket to the provided player_ticket. The order of the numbers in each ticket doesn't matter, just that they contain the same numbers. If they guessed correctly, return True, else return False.
    - *Hint: there is a list function that can help make the list of numbers easier to compare*
    - *Hint: While testing you might want to temporarily hard code the winning ticket.*

- ☐ **Objective 6:** Call won_the_lottery after the player_ticket list has been created in the main. Print out a message indicating if the user won or not.

Test Cases (10 pts): See Project FAQ

Followup Questions (10 pts):

1. How long did you spend on the project?
2. Did you complete the extension? If no, why not?
3. How did you like getting objectives for a project? Explain in detail.
4. Was there any part of the assignment that you got stuck on? Explain.

Extension (+5 pts):
Update the code so that duplicate numbers are not allowed in either the player's ticket or the winning ticket.

Sample Run 1:

```
Let's play the lottery!

Number #1: 89
Number #2: 45
Number #3: 32
Number #4: 14
Number #5: 67
Number #6: 67
Number #7: 23
```

Sample Run 2:

```
Let's play the lottery!

Number #1: -90
Not in range, try again: -24
Not in range, try again: 167
Not in range, try again: 67
Number #2: 67
Number #3: 45
Number #4: 32
Number #5: 12
Number #6: 78
Number #7: 98

You won!!!
```