CS5001: Lab 8. Due on Friday, Nov-3-2023.

Name(s): Xujia Qin

Email(s): qin.xuj@northeastern.edu

You can work on this lab either individually or in small group of two or three students. If working in a group, include names of all the students in the submission PDF.

Getting credit for this lab. This lab handout has several empty boxes that prompt you to answer a question. As part of the lab, you are to write the answers to these questions inside the boxes/blanks. When you are finished, you should create a PDF and upload it on Canvas. If you don't finish, you have until 11:59 PM on Friday, Nov-3-2023 to submit.

What computer to use? If your primary computer is a laptop, bring it to the lab to work on, as lab is an excellent opportunity to get started with Python on your machine. You should follow the instructions on the course website. Ask a TA for help if you have problems with your installation. If you prefer, you could also use one of the machines in the lab room to work on this lab assignment.

Lab Materials. Lab materials can always be found on Canvas under the appropriate lab posting.

For today's lab, you need: this handout, DiceRolls.py, ShowAdd1.py, ShowAdd2.py. All the files are posted on Canvas with the lab handout.

1 Dice Rolls

The given module DiceRolls.py contains the function randiList(). The same module has an application script with parts that you have to complete. Fill in code below the comments indicating "TODO:" and check your answers by running DiceRolls.py. For this question, you are not allowed to use count or sum functions/methods.

Specifically, you need to write code for the following three features in the provided DiceRolls.py file:

1. Compute the average value in list D and assign it to the variable “ave0”. **(20 points)**
2. Compute an estimate of the probability that the sum of the three die values is 7 and assign it to a variable “Prob1”. **(20 points)**
3. Compute an estimate of the probability that one die value is at least as big as the sum of the other two die values and assign to a variable “Prob2”. **(20 points)**

Notes:

- Please do not change/remove/modify any of the print statements already included in the file DiceRolls.py.
- Do not change any of the other function, program logic implemented in DiceRolls.py.
- Don’t change the value of “N = 100000” in your final submission.
- Implement your solutions to the above functionality and submit the completed version of DiceRolls.py on Canvas along with the writeup for Q2 inside a single zip file.

2 Functions and lists

2.1

A)

```
def Add1(x,y):
    """
    PreC: x and y are lists of numbers with len(x)==len(y)
    """
    z = []
    for k in range(len(x)):
        s = x[k]+y[k]
        z.append(s)
    return z


if __name__ == '__main__':

    # Example 1
    print ('\nExample 1:')
    a = [1,2,3];  b = [10,20,30];  c = Add1(a,b)
    print (a, b, c) #Print-1

    # Example 2
    print ('\nExample 2:')
    a = [1,2,3]; b = [10,20,30]
    b = Add1(a,b)
    print (a, b) #Print-2
```

In the code above, without executing the code, what do you think should be the output of the two print statements marked by “#Print-1”, “#Print-2”? **(10 points)**

```
Example 1:
[1,2,3] [10,20,30] [11,22,33]
Example 2:
[1,2,3] [11,22,33]
```

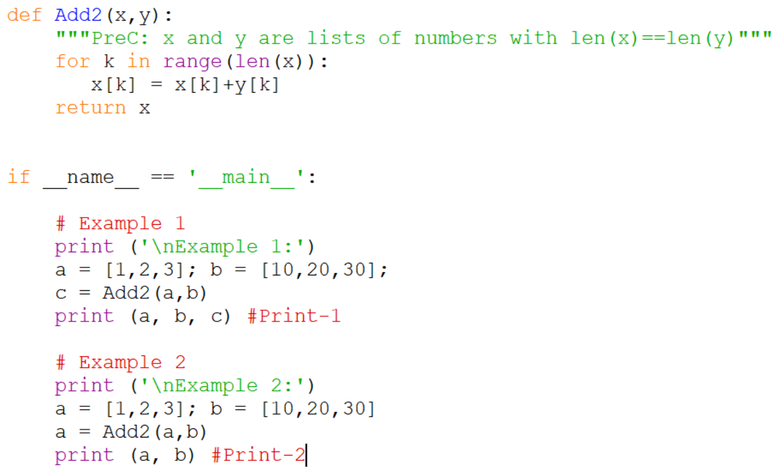
B) Now execute the file ShowAdd1.py. What is the actual output? **(10 points)**

```
Example 1:
[1, 2, 3] [10, 20, 30] [11, 22, 33]

Example 2:
[1, 2, 3] [11, 22, 33]
```

2.2

A)



```
def Add2(x,y):
    """PreC: x and y are lists of numbers with len(x)==len(y)"""
    for k in range(len(x)):
       x[k] = x[k]+y[k]
    return x


if __name__ == '__main__':

    # Example 1
    print ('\nExample 1:')
    a = [1,2,3]; b = [10,20,30];
    c = Add2(a,b)
    print (a, b, c) #Print-1

    # Example 2
    print ('\nExample 2:')
    a = [1,2,3]; b = [10,20,30]
    a = Add2(a,b)
    print (a, b) #Print-2
```

In the code above, without executing the code, what do you think should be the output of the two print statements marked by “#Print-1”, “#Print-2”? **(10 points)**

Example 1:
[11,22,33] [10,20,30] [11,22,33]

Example 2:
[11,22,33] [10,20,30]

B) Now execute the file ShowAdd2.py. What is the actual output? **(10 points)**

```
Example 1:
[11, 22, 33] [10, 20, 30] [11, 22, 33]

Example 2:
[11, 22, 33] [10, 20, 30]
```