

CS 115 - Introduction to Programming in Python

Lab 03

Lab Objectives: Functions

Instructions: For this assignment, you can use your favorite IDE (Spyder or Jupyter recommended). Upload your solutions as a single .zip file to the Lab01 assignment on Moodle before the end of your lab session. Use the following naming convention: **SS_Lab01_Surname_FirstName.zip** where SS is the section number 01, 02, 03, ..., & and Surname is your family name, & FirstName is first name. You must show and explain your solutions to your TA during your lab session to receive a grade.

Note: Sample runs show the user input in red.

1. In this question, write a docstring for your function.
 - a. Write a function, randomOdd, that accepts two parameters, the lower and upper bounds of a range, generates and returns a random odd integer (not divisible by 2) in that range inclusive. Use the function random.randint(a, b) to generate an integer in range [a, b] inclusive **and** math operators to write this function.
 - b. Using your function from part a), write a program that inputs three integers from the user: the first two integers are the lower and upper bounds of a range and the third integer is how many odd numbers should be generated in that range.

Assume that all input values are ints.

Sample Run 1:

Enter the lower bound of the range: **10**

Enter the upper bound of the range: **40**

How many odd integers do you want in this range: **5**

Here are 5 random odd integers in range[10, 40]: 15 35 11 21 31

Sample Run 2:

Enter the lower bound of the range: **9**

Enter the upper bound of the range: **39**

How many odd integers do you want in this range: **8**

Here are 8 random odd integers in range[9, 39]: 25 13 21 23 39 23 13 37

2. In this question, write a docstring for your function.

Write a function, `persistence`, that accepts a positive parameter `num` and returns its multiplicative persistence, which is the number of times you must multiply the digits in `num` until you reach a single digit. Here are some examples:

- `persistence(93) ==> 3` because $9 \cdot 3 = 27$, $2 \cdot 7 = 14$, $1 \cdot 4 = 4$ and 4 has only one digit.
- `persistence(999) ==> 4` because $9 \cdot 9 \cdot 9 = 729$, $7 \cdot 2 \cdot 9 = 126$, $1 \cdot 2 \cdot 6 = 12$, and finally $1 \cdot 2 = 2$.
- `persistence(4) ==> 0` because 4 is already a one-digit number.
-

Don't use strings to solve this question. Use math operators.

See sample runs below.

Sample Run 1: Enter an int: 539 multiplicative persistence of 539 is 3	Sample Run 2: Enter an int: 78999 multiplicative persistence of 78999 is 2
Sample Run 3: Enter an int: 68889 multiplicative persistence of 68889 is 7	Sample Run 4: Enter an int: 2677889 multiplicative persistence of 2677889 is 8

3. In this question, write a docstring for your function.

- a) Write a function named `flipCoinUntil_X_Heads(x)` that takes integer `x` (`x > 0`) as parameter and repeatedly flips a coin until `x` heads in a row are seen. You should use the `random.randint()` function to give an equal chance to a head or a tail. Each time the coin is flipped, what is seen is displayed (H for heads, T for tails). When `x` heads in a row are flipped, an appropriate message is displayed.
- b) Using your function from part a), write a program that inputs a positive integer from the user, and displays the coin flips until the specified number of heads in a row are seen. It should stop inputting an integer from the user when a non-positive integer is specified by the user.

Sample run:

```
Enter how many heads you want in a row: 1
T H
1 heads in a row!
2 coin flips in total

Enter how many heads you want in a row: 1
H
1 heads in a row!
1 coin flips in total

Enter how many heads you want in a row: 1
T T H
1 heads in a row!
3 coin flips in total

Enter how many heads you want in a row: 2
T T H H
2 heads in a row!
4 coin flips in total
```

Enter how many heads you want in a row: 3
H T H T T H H T T T T T T T T H T H H T H H H
3 heads in a row!
24 coin flips in total

Enter how many heads you want in a row: 2
H T T T H H
2 heads in a row!
6 coin flips in total

Enter how many heads you want in a row: 3
T H H T T H H H
3 heads in a row!
8 coin flips in total

Enter how many heads you want in a row: 5
T H H H H H
5 heads in a row!
6 coin flips in total

Enter how many heads you want in a row: 5
T T H H H H T T T H T T T H H H H T T H T H T T H T H H T T H T T H
T H T T H H T H T T H T T H T T H H H H T H T H T T H H H T T H H T T T
T T T T H H H T H T T H T T H H H H T T T T T H T H H H T H T T T T T T
H H H H H
5 heads in a row!
114 coin flips in total

Enter how many heads you want in a row: 5
T H H T H T H T T T H T T H T T T T H T T H H T T T H T H T T H T T H H
H T H T T T T T T H H T T H H T H H T H T T T T H H H T H T T T T T H
T H T T H T H H T T H T T H H H H T H T H T H T H T T H T T H T H T T
H H T H T T H T H H H H H
5 heads in a row!
122 coin flips in total

Enter how many heads you want in a row: 0
bye