INSY 5336
Python Programming
Spring 2023
Homework 1 (100 points)

The following guidelines should be followed and will be used to grade your homework:

- All code to be implemented and submitted as a jupyter notebook (.ipynb) file.
- This is an individual homework assignment; no group submissions will be accepted. If you discuss in groups, please write your code individually and submit.
- Sample runs shown in the question should be used as a guide for implementation. However extensive testing needs to be done on your code to deal with all test cases that might possibly be executed.
- The high-level algorithm of how you are solving the problem should be documented in the cell preceding the code in markdown language.
- The instructions for running of each cell and the expected results should be documented in the cell preceding the code using markdown language.
- Every code segment in the jupyter notebook cells should be well documented with comments. Use # in the code to provide comments and they should explain the algorithm step and what the code segment is doing.
- Error checking in your code is very important and differentiates a high-quality programmer from a low quality one. Hence you should account for invalid user inputs, infinite loops, out of range results, etc. and resolve them by appropriate error messages. **The homework will be graded for robustness of your code.**

1. (10 points) Implement a program that requests three **strings** from the user. Your program should concatenate the first two strings ***in the reverse order*** and compare the concatenated string with the third string. If they are equal, your program should print "They are equal", otherwise, the program prints "They are not equal". A sample run is shown below:

   Enter first string: you!
   Enter second string: Thank
   Enter third string: Thankyou!
   They are equal

   Enter string 1: you2!
   Enter second string: Thank
   Enter third string: Thankyou!
   They are not equal

2. (10 points) Write a Python program that reads in 3 integers (user first determines 3) and displays the following:

a. the average of the numbers (two decimal places suggested)
b. the maximum of the numbers

Do not use Python's internal functions for average and maximum.

A sample run is shown below:

Enter set of 3 integers: 3
Enter first number: 7
Enter second number: 10
Enter third number: 1
The average of 7, 10 and 1 is: 6.0
The maximum of the three numbers is: 10

Enter set of 3 or 4 integers: 4
Enter first number: 7
Enter second number: 10
Enter third number: 1
Enter fourth number: 6
The average of 7, 10 and 1 is: 6.0
The maximum of the three numbers is: 10

four

3. (10 points) An integer, greater than 1, that is only divisible by 1 and itself is called a prime number. All other numbers greater than 1 are called composite numbers. The integers 0 and 1 are neither prime nor composite. Write a python program that requests a positive integer from the user, determines if it is a prime, composite or neither prime or composite and prints the message.
Sample runs are shown below:

what about 0?

Enter positive integer: 3
It is a prime
Enter positive integer: 4
It is a composite
Enter positive integer: 1
It is neither prime nor composite
Enter positive integer: -5
Error: You did not enter a positive integer
Enter positive integer: -2.5
Error: You did not enter a positive integer
Enter positive integer: Hi
Error: You did not enter a positive integer

4. (15 points) Implement a Python function with 3 arguments called SimpleInterest(Principle, Interest, Years). The arguments are:
   a. A principle amount (float)
   b. Interest Rate (0 to 100% as a float)

c. Years (integer)

Your function should return the simple interest amount.
The formula for a simple interest is (Principle*InterestRate*Years)/100
Write your function definition in one cell and the function calling code below it or in a different cell in Jupyter Notebook.
Your code should ensure that the user input is in the correct type and format

5. (15 points) Write a python program for a shopping cart. The program should allow shopper to enter the product name and price. Use loop so that shopper can enter as many inputs as necessary and validate the inputs as product name should be string and price should be more than $0. At the end, the output should
   - display the total the shopper needs to pay. Use f-string to format the total value for two decimal points and comma.
   - print the name and price for all the entries with appropriate headings.

6. (20 points) "Rock-paper-scissors is a hand game that is played by two people. The players count to three in unison and simultaneously "throw" one of three hand signals that correspond to rock, paper or scissors. The winner is determined by the rules:
   - Rock smashes scissors
   - Scissors cuts paper
   - Paper covers rock

   Rock-paper-scissors is a surprisingly popular game that many people play seriously (see the Wikipedia article for details).
   Write a Python program to ask the user's choice of Rock paper scissors. The program then, randomly, chooses a choice for itself (the computer) and then compares it with the user's choice. The output should show, the user's choice, the program's choice and the winner.
   Sample runs are shown below:

   Enter your choice: rock
   Computer chooses scissors
   You, the user, win!

   Enter your choice: paper
   Computer chooses rock
   I, the computer, win!

   Enter your choice: scissors
   Computer chooses scissors
   User and computer tie!

   Finally, you want to play at least 10 times. Store and display the computer choices and user inputs in a pandas dataframe with headings.

What is the logic to guess? Random or binary search?

7. (20 points) Write a program where the user and the program/computer play a number guessing game. The program should prompt the user for a number (between 1 and 100, inclusive) then the program/computer has to guess what the user entered. Keep track of the number of iterations it takes for the computer to guess the number. Sample runs are shown below:

   Enter number to be guessed: 88
   You entered 88, and it took the program 3 iterations to guess

   Enter number to be guessed: 55
   You entered 55, and it took the program 19 iterations to guess