COMPUTER SCIENCE 10A (FALL TERM, 2021)
INTRODUCTION TO PROBLEM SOLVING IN PYTHON
PROGRAMMING ASSIGNMENT 6
DUE: MONDAY, NOV 8TH, 11.59PM

Program Description:

This assignment will test your understanding of the use of **string indexing, for loops, while loops, ASCII values .** Write six scripts to solve the following problems and name each script Problem1.py, Problem2.py, etc.

Problem 1:

You must use while loops. Write a program that prompts the user to enter a positive integer value, and compute the following sequence:

- If the value is even, halve it.
- If it's odd, multiply by 3 and add 1.
- Repeat this process until the value is 1, printing out each value.
- Finally print out how many of these operations you performed.

Note: If the input value is less than 1, print a message containing the word Error and exit the program. You can assume that the input will have smaller than 200 operations. This is how the output should look like:

```
Initial value is: 9
Next value is: 28
Next value is: 14
Next value is: 7
Next value is: 22
Next value is: 11
Next value is: 34
Next value is: 17
Next value is: 52
Next value is: 26
Next value is: 13
Next value is: 40
Next value is: 20
Next value is: 10
Next value is: 5
Next value is: 16
Next value is: 8
Next value is: 4
Next value is: 2
Final value 1, number of operations performed 19
```

Problem 2:

Write a program that prompts the user for a string and an integer and prints out the string with each character repeated the same number of times as the integer entered by the user. Your possible interaction with the user should be like:

```
Please enter a String: red
Please enter a multiplier for each character to repeat: 4
I got: rrrreeeedddd
```

Problem 3:

Write a program that determines if a string is palindrome. A palindrome string is a string that reads the same backward as forward. For example: “ABA”, “madam”, “abba”. Prompt the user for a string and validate if the string is a palindrome. Your program should return True if the string is palindrome, False otherwise. Assume the string the user entered has no punctuations and spaces.

Problem 4:

Write a program that accepts two string variables, **first** and **last**, which the user should populate with his or her name. First, convert both strings to all lowercase. Your program should then create a new string that contains the full name in pig latin with the first letter capitalized for the first and last name. Use only the pig latin rule of moving the first letter to the end of the word and adding “ay”. Output the pig latin name to the screen.

For example, if the user inputs “Iraklis” for the first name and “Tsekourakis” for the last name, then the program should create a new string with the text “Raklisiay Sekourakistay” and print it.

Problem 5:

Write a program that produces a Caesar cipher of a given message string. A Caesar cipher is formed by rotating each letter of a message by a given amount. For example, if your rotate by 3, every A becomes D; every B becomes E; and so on. Toward the end of the alphabet, you wrap around: X becomes A; Y becomes B; and Z becomes C. Your program should prompt for a message and an amount by which to rotate each letter and should output the encoded message.

Problem 6:

Write a program that allows the user to play many rounds of the Rock Paper Scissor game (the user will decide when to end the game). The user and computer will each choose between three items: rock (defeats scissor, but loses to paper), paper (defeats rock, but loses to scissors), and scissors (defeats paper, but loses to rock). If the player and computer choose the same item, the game is a tie. Extend this program to include different algorithmic strategies for choosing the best item. Should the computer always pick a particular item or a repeating pattern of items? Should it count the number of times the opponent chooses various items and base its strategy on this history? Be creative here. Each strategy should be in its own method, and at the start of a round of games the user should be able to choose which strategy to play against. Include at least two strategies.

Guidelines:

You should not have any code, except a call to your `main` function, outside of a function. You should use the `main` function to call other functions that implement the solution.

Include a header comment at the beginning of your program with some basic information and a description of the program in your own words.

```
# Name Student
# COSI 10a, Fall 2021
# Programming Assignment #6
#
# Description: ...
```

You also need to include comments in your code.

For this assignment, you should limit yourself to the Python features covered up to lecture 16. Though we will cover more material while you work on this assignment, please do not use it on this assignment.

Submission and Grading:

All your python scripts should be inside a folder named `yourfirstname_yourlastnamePA6`, then zip the folder into a zip file for submission. The zip file should have the following name: `yourfirstname_yourlastnamePA6.zip` (Please make sure to use exactly this file name, including identical capitalization).

Your program should be submitted via Latte before it is due (for late policy check the syllabus).

You will be graded on:

- **External Correctness:** The output of your program should match exactly what is in the question description. Programs that do not compile will not receive points for external correctness.
- **Internal Correctness**: Your source code should follow the stylistic guidelines shown in class. Remember to include the comment header at the beginning of your program and comment your code.