Assignment 1: Rock-Paper-Scissors

Summary

In this assignment, students will develop their own version of a game of *Rock Papers Scissors*. This game will be developed using Python, leveraging what has been learnt in class.

Program Description

Create a game of *Rock Papers Scissors*. The user will play against the computer until one of the players wins a specified number of rounds – winning the "match". For each round, both players reveal their choice of rock, paper, or scissors. Scissors beats paper, paper beats rock, rock beats scissors. At the end of the match, the program will ask if the user would like to play again.

Submission Checklist

Submit the following in Slate:

1. Visual Studio Code solution containing all the source files and git repository, compressed in a ZIP file.

Requirements

Project Setup (10%)

- Create a Python program with source files in one directory/folder.
- Be sure to use version controlling with a **git repository**. **Commit changes throughout the development of the program. Be sure to start committing changes from the start!**
- The program should be developed, and able to run, using Visual Studio Code.

Program Structure (45%)

It is important that your code is clear, concise, and adheres to standards and naming conventions discussed during class.

Your code must meet all the following conditions:

- Must have no syntax errors in your submission.
- Can only import the `random` and `math` modules (in addition to any module you create).
- **Use only techniques and concepts demonstrated in the course (up to, and including, week 4).**
- Must **not** make use of custom types/classes or collections (such as arrays).
- Should have meaningful names for variables and functions as well as follow Python naming conventions.
- Uses a file named `program.py` to manage the program's flow and **all** user interactions (input and output).
- Uses best practices shared in class, including writing modular code (e.g. defining small functions that do different things).

In your code, provide meaningful and relevant comments on design decisions made throughout the program. This includes comments for:

- Branching and looping.
- Functions (placed at definition).
- Header information for each file, describing the purpose of the file/module.
- Header information of the title, course, and author in `program.py`
- Where needed to explain complex code or specific decisions.

Hint: investigate using `range()` with the `random` package's `choice()` function.

User Interaction (45%)

Implement basic user interaction that allows the user to interact with the program according to the specifications below:

1. On first launching the program, the user should see a welcome message and provide any relevant instructions on how to play. This message and instructions only appear on first run of the program.
2. There are 2 players in the game: the user and the computer.
3. For each match:
    a. The user will specify the number of rounds either user must win to win the match. Assume the user will always enter an integer value.
    b. The user must specify a number-of-rounds that is greater than or equal to 1, requesting user input until a valid number is entered.

    c. The user will keep playing rounds with the computer until either user has won the specified number of rounds.
    d. If one of the players wins the specified number of rounds, the player wins the match. The user is notified of this result.
4. For each round:
    a. Prompt the user to choose either Rock (R), Paper (P), or Scissors (S).
    b. The user can input an option in either upper or lower case. The user can enter either the entire word or the first letter.
    c. If the user enters `quit`, the match should end immediately.
    d. If the user provides any other input, the program should indicate it does not understand and try to get input again.
    e. The program outputs a random choice of either Rock (R), Paper (P), or Scissors (S). This is displayed immediately following the user's input.
    f. The players' choices are compared to determine who wins the round. The user is notified of who wins the round. It is possible to have a tie for a round.
    g. An updated score is also displayed as user to computer (e.g. `1:0`).
5. At the end of a match, or on a request to quit, the use is asked if he/she wants to play another match. If no, exit the program.

Note: You can adjust the wording and style of the user interaction to make playing your submission unique and fun.

For a sample of user interaction, see Appendix 1.

Notes

1. The assignment shall be submitted by the specified due date. Late submissions will be penalized with 10% per day for up to 3 calendar days after which the assignment cannot be submitted anymore. Assignments are not accepted after the final deadline.
2. This assignment shall be completed individually.
3. Advanced AI tools are not permitted in any aspect of the assignment (e.g. artificial intelligence or machine learning tools such as ChatGPT). The assignment is to be completed without substantive assistance from others, including automated tools. Remember that completing the assignment by yourself will ensure your success on the midterm and final exam. See the Academic Honesty guidelines at Sheridan.
4. **Reminder: if you want to include code found elsewhere, or not sure about how you can use a resource or technique, please consult with your professor first.**

5. Submitting the assignment is done using the SLATE Assignment folder. DO NOT email your submission.
6. Try to have fun when working on this assignment!

Appendix 1: Sample User Interaction

```
================== ROCK PAPER SCISSORS ==================

Welcome to Rock Paper Scissors!

Rules:
Scissors beats paper, paper beats rock, rock beats scissors.

How many rounds must a player win to win the match? 2

Sounds great! Let's play...

Choose Rock (R), Paper (P), or Scissors (S): S
Computer: Rock
You win! Score now: 1:0

Choose Rock (R), Paper (P), or Scissors (S): Paper
Computer: Paper
Tie! Try again...

Choose Rock (R), Paper (P), or Scissors (S): 1
Not sure what you mean. Try again...

Choose Rock (R), Paper (P), or Scissors (S): r
Computer: Paper
I win! Score: 1:1

Choose Rock (R), Paper (P), or Scissors (S): R
Computer: Scissors
You win! Score: 2:1

You win the match! Do you want to play another match? N

Thanks for playing! Bye for now!
```