# Assignment 7 – Functions

## Learning Objective

Define and call functions to simulate a game of ROCK PAPER SCISSORS.

## Assignment Description

Write a program that allows the user to play ROCK PAPER SCISSORS against the computer. Your code will allow the user to enter their hand and then randomly choose an option for the computer.

A winner is selected based on the following rules:

- ROCK smashes SCISSORS (If one player chooses rock and the other chooses scissors, then the player who chooses rock wins).
- SCISSORS cut PAPER (If one player chooses scissors and the other chooses paper, then the player who chooses scissors wins).
- PAPER covers ROCK (If one player chooses paper and the other chooses rock, then the player who chooses paper wins).
- If both players make the same choice, then it is a tie.

The program will allow the user and computer to play games until one of them wins two times.

## Steps

1. In PyCharm (Community Edition), open your existing ITP115 project.
2. Under the Assignments directory, create a new directory called **a7_*last_first*** where *last* is your last/family name and *first* is your preferred first name. Use all lowercase letters.
3. In the directory, create a new Python file called **assignment7.py**.
4. At the top of the file, put comments in the following format and replace the name, email, and section with your actual information:

```
# Name, USC email
# ITP 115, Fall 2022
# Section: number or nickname
```

```
# Assignment 7
# Description:
# Describe what this program does.
```

5. Import the random module.
6. Define the **displayRules()** function.
   - Parameters (input): None
   - Return value (output): None
   - Displays the game rules to the user:

```
Let's play Rock Paper Scissors.
The rules of the game are:
   ROCK smashes SCISSORS
   SCISSORS cut PAPER
   PAPER covers ROCK
   If both the hands are the same, it's a tie
```

   - You will call this function in the main() function.
7. Define the **userPlays(optionsList)** function.
   - Parameter: optionsList is a list with the three choices: ROCK, PAPER, SCISSORS
   - Return value: a string that contains the user's hand
   - Get input from the user with the following prompt:

```
Enter ROCK, PAPER, or SCISSORS:
```

   - Use a loop to make sure that the user enters a valid choice. A valid choice is determined by being in the optionsList parameter. Allow the user to enter their choice using upper and lower case letters. Remove any whitespace before or after the user's input. Here's a sample with the user's input in green.

```
Enter ROCK, PAPER, or SCISSORS: r
Enter ROCK, PAPER, or SCISSORS: 0
Enter ROCK, PAPER, or SCISSORS:   rock
```

   - Display to the user what the user entered. Here's a sample:

```
User plays ROCK
```

   - You will call this function in the main() function.

8. Define the **computerPlays(optionsList)** function.
   - Parameter: optionsList is a list with the three choices: ROCK, PAPER, SCISSORS
   - Return value: a string that contains the computer's hand.
   - Use the random module to get a random choice from the optionsList parameter.
   - Display to the user what was randomly selected for the computer. Here's a sample:

```
Computer plays PAPER
```

   - You will call this function in the main() function.
9. Define the **gameOutcome(computerStr, userStr)** function.
   - Parameter 1: computerStr is a string with the computer's hand
   - Parameter 2: userStr is a string with the user's hand
   - Return value: an integer representing the result of the game: -1 if the computer won, 0 if it was a tie, and 1 if the user won
   - Print a message saying the game was a tie or who won (computer or user). You will print one of the following three messages:

```
You and the computer tied.
```

```
Computer wins.
```

```
You win!
```

   - This function contains the game logic. It simulates the game and determines a winner. Use the logic in the Assignment Description above. Use branching.
   - Return the appropriate integer depending on the result of the game.
   - You will call this function in the main() function.
10. Define and call the **main()** function. This function will not have any parameters nor a return value.
    - Create a list variable that has the three choices: ROCK, PAPER, and SCISSORS. Make sure that the words are in all upper case.
    - Create variables to hold the number of times the computer wins and the user wins.
    - Display the rules of the game by calling the displayRules() function.

- Create a loop to play the game until the computer or user wins twice.
- In the loop, allow the user to play by calling the userPlays() function. Use the list variable you created in the main() function as the argument to match the optionsList parameter. Make sure to capture the function's return value in a variable.
- Still in the loop, allow the computer to play by calling the computerPlays() function. Use the list variable you created in the main() function as the argument to match the optionsList parameter. Make sure to capture the function's return value in a variable.
- Still in the loop, determine the outcome of the game by calling the gameOutcome() function. Use the appropriate variables as arguments when calling the function to match the function's parameters. Remember that the names for the arguments do not need to be the same as the names for the parameters.
- After the loop, print who won the game two times. You will print one of the following messages depending on how won twice:

| |
|---|
| **You won 2 games!** |
| **Computer won 2 games.** |

- After you defined the main() functions, don't forget to call it.

11. Be sure to comment your code. This means that there should be comments throughout your code. Put a comment block before each function stating the parameters, return values, and what that function does. Points will be deducted for not having comments.
12. Follow coding conventions. You should use lowerCamelCase or snake_case for variable names. You are welcome to create any variables that you need.
13. Test the program. Look at the Sample Output below. Assignments that do not run are subject to 20% penalty.
14. Prepare your submission:
    - Find the **a7_*last_first*** folder on your computer and compress it. This cannot be done within PyCharm.
    - On Windows, use **File Explorer** to select the folder. Right click and select the Send to -> Compressed (zipped) folder option. This will create a zip file.

- On Mac OS, use **Finder** to select the folder. Right click and select the Compress "*FolderName*" option. This will create a zip file.

15. Upload the zip file to your Blackboard section:
    - On Blackboard, navigate to the appropriate item.
    - Click on the specific item for this assignment.
    - Click on the **Browse Local Files** button and select the zip file.
    - Click the **Submit** button.

## Grading

- This assignment is worth 35 points.
- Make sure that you the program runs. Points will be taken off if the graders have to edit the source code to test your program.
- Make sure to submit your assignment correctly as described above. Points will be taken off for improper submission.

| Item | Points |
|---|---|
| displayRules() | 4 |
| userPlays() | 8 |
| computerPlays() | 5 |
| gameOutcome() | 8 |
| main() | 10 |
| **Total** | **35** |

## Sample Output

```
Let's play Rock Paper Scissors.
The rules of the game are:
      ROCK smashes SCISSORS
      SCISSORS cut PAPER
      PAPER covers ROCK
      If both the hands are the same, it's a tie

Enter ROCK, PAPER, or SCISSORS: r
Enter ROCK, PAPER, or SCISSORS: 0
Enter ROCK, PAPER, or SCISSORS:   rock
User plays ROCK
Computer plays PAPER
You win!

Enter ROCK, PAPER, or SCISSORS: p
Enter ROCK, PAPER, or SCISSORS: 1
Enter ROCK, PAPER, or SCISSORS:   Paper
User plays PAPER
Computer plays SCISSORS
Computer wins.

Enter ROCK, PAPER, or SCISSORS: s
Enter ROCK, PAPER, or SCISSORS: 2
Enter ROCK, PAPER, or SCISSORS:  scizzors
Enter ROCK, PAPER, or SCISSORS:   scissors
User plays SCISSORS
Computer plays SCISSORS
You and the computer tied.

Enter ROCK, PAPER, or SCISSORS: PAPER
User plays PAPER
Computer plays SCISSORS
Computer wins.

Computer won 2 games.
```