Introduction

This week's lab is another example of a split lab to give you more experience with planning a project before you sit down to write code. We will be combining several concepts this week.

Lab Objectives

By successfully completing today's lab, you will be able to:

- Conduct a brainstorming session with one or two individuals in the class.
- Create a planning document outlining your solution to the code.
- Implement a programming project that uses random numbers and loops.
- Get more experience working and exchanging ideas with people in your 1011 lab. (An important skill!)

Prior to Lab

- You should be familiar with input and output, conditionals such as if, if/else, and switch statements, as well as for and while loops.
- This lab corresponds with zyBooks Chapters 4.1 – 4.4.
- Random numbers are covered in zyBooks Chapter 2.19.

Deadline and Collaboration Policy

- Part 2 of this assignment is due by 11:59 PM on Tuesday (9/24/2019) via Canvas.
- Part 3 of this assignment is due by 11:59 PM on Friday (9/27/2019) via Canvas.
    - More instructions on what and how to submit are included at the end of this document.
- You should write your solutions to Part 2 and Part 3 of this lab by yourself. In this lab, you will talk at a high level with others in your lab about a solution. However, you will create all code by yourself and **you should not talk about specific code to anyone but a course instructor or lab teaching assistant.**
- Your zyBooks chapters and lecture slides are available resources you can use to assist you with this lab.

Lab Instructions

The Problem:

This week's lab involves creating an electronic version of a timeless game: Rock, Paper, Scissors! In this game, two players choose either to be a "rock", "paper", or a pair of "scissors" and shows their choice to the other player at the same time. The winner of each match is chosen by the following rubric:

- Rock wins against scissors
- Scissors wins against paper
- Paper wins against rock
- If both players select the same entity, the result is a tie.

Again, we structured this week's lab to have substantial planning on the first day, and you coding your solution on the second day:

Lab 05 Structure	
Tuesday	Thursday
30 minutes: work with 1-2 other individuals on Part 1. 20 minutes: work individually on Part 2. Submit results of Part 2 to Canvas by 11:59 PM	50 minutes: work individually on Part 3. Submit to Canvas by 11:59 PM on Friday.

Part 1: Planning with a Group (First 30 minutes)
***** Do not use a computer or calculator in this part *****

Working with the people in your row, brainstorm ways that you can implement a game of rock, paper, scissors between a user and the computer. Take a look at the specifications for how your program needs to operate:

- The user must be prompted for the number of games the user wants to play.
- The program should keep track of the score of both the computer and the human.
- The program should keep running until the number of games specified are completed.
- During each turn, the computer randomly selects its choice.
- After each turn, the program should display who won the previous round and the score after that turn.
- After all matches are played, a summary of the final score is displayed.

Specific items that you should consider with your group during your planning session:

- What are the best ways to keep track of scores?
- What are all the tasks that need to be accomplished and where do they occur
  - (e.g., inside a loop? At the beginning of the program? At the end?)
- Is a loop needed? If so, which one best accomplishes the goal? What would the conditional be?
- How do we check to see if input is correct?
- What is the best way for the computer to make its choice?

Sample Output

```
./a.out

Starting the CPSC 1011 Rock, Paper, Scissors Game!

Enter the number of matches to play: 3

    Match 1: Enter R for rock, P for paper, or S for scissors: R
    The computer chose scissors. You win!
    Scores:  You-1

    Match 2: Enter R for rock, P for paper, or S for scissors: R
    The computer chose paper. You lose.
    Scores:  You-1  Computer-1

    Match 3: Enter R for rock, P for paper, or S for scissors: S
    The computer chose scissors. You tied.
    Scores:  You-1  Computer-1  Ties-1

The game of 3 matches is complete. The final scores are:
You:        1
Computer:   1
Ties:       1
```

Part 2: Planning on Your Own (20 minutes of Tuesday Lab)
***** It's okay to use a computer and/or calculator for this portion *****

Taking the information that was created during your brainstorm, write your own solution *in psuedocode*. How you choose to structure the planning document is of your choice, but you need to do the following:

- Review the explanation of *pseudocode* in the Safari playlist for this lab (https://learning.oreilly.com/playlists/1480ac0b-84d1-4ef6-97f4-f6936af15daf) or using the pseudocode.pdf document in Canvas.
- *Pseudocode* is NOT code. It is structured English that where you convey the key parts of your algorithm. For example, consider this *pseudocode* to calculate the average grade given 10 grades:

```
Set total to zero
Set grade counter to one

While grade counter is less than or equal to ten
    Input the next grade
    Add the grade into the total
    Add one to the grade counter
End While

Set the class average to the total divided by ten
Print the class average
```

- Thoroughly consider all the issues you need to fully create a program as specified above (such as how to make the formatting look correct, how many variables do you need, what type, etc.).
- Include the following mandatory header:
  - Your First (or preferred name) and Last Name
  - Your Lab and Lecture Section
  - Lab 05 – Part 2
  - Today's Date
  - Collaboration Statement: In this statement you indicate who you worked with, and which lecture sections they are enrolled in.

Submit this part of the assignment to Canvas by 11:59 PM (as specified in the submission Guidelines)

Part 3: Implementation (50 minutes on Thursday).

Lab Exercise

Using your planning document from Part 2, you will create a C program called rockpaperscissors.c that creates the game. When playing, the game should match the example given above, visually and in functionality.

Make sure you include the standard lab comment block introduced in Lab 3.

Make sure the computer selects its choice randomly and doesn't factor in the input from the user in making its decision. While technically this could be done, cheating isn't allowed in Rock Paper Scissors!

Bonus Exercise

Create a copy of your rockpaperscissors.c file called rockpaperscissors-enhanced.c that extends your program by the following:

- At the start of the game, the user is prompted to decide between two different versions of the games as shown in the output below.

```
Starting the CPSC 1011 Rock, Paper, Scissors Game!

Would you like to play until:
(Option S) a set number of matches are played?
(Option M) either player wins a certain number of matches?

Enter S or M:  M
Enter the number of matches a player must win to end the game: 3

        Match 1: Enter R for rock, P for paper, or S for scissors: P
        The computer chose rock. You win!
        Scores:   You-1  Computer-0

        Match 2: Enter R for rock, P for paper, or S for scissors: R
        The computer chose scissors. You win!
        Scores:   You-2  Computer-0

        Match 3: Enter R for rock, P for paper, or S for scissors: R
        The computer chose paper. You lose.
        Scores:  You-2  Computer-1

        Match 4: Enter R for rock, P for paper, or S for scissors: P
        The computer chose scissors. You lose.
        Scores:  You-2  Computer-2

        Match 5: Enter R for rock, P for paper, or S for scissors: S
        The computer chose rock. You lose.
        Scores:  You-2  Computer-3

The game is complete, because a player reached 3 wins. Final scores:
You:            2
Computer:       3
```

Submission Guidelines

- Part 1: No submission, but don't skip it. Thinking aloud and collaborating at a high level is incredibly important in computer science. It'll help you ask better questions at office hours, write cleaner code, do great group work, and even do better in interviews.
- Part 2: Submit your writeup to the Canvas assignment associated for this lab by 11:59 PM on Tuesday (9/24/2019). You should check within Canvas to make sure your submission was uploaded correctly and in its entirety.
- Part 3: Submit your rockpaperscissors.c program to Canvas assignment associated for this lab by 11:59 PM on Friday (9/27/2019). You should verify within Canvas to make sure your submission was uploaded correctly and in its entirety.
    - If you opt to complete rockpaperscissors-enhanced.c, submit the program to the associated Canvas assignment by 11:59 PM on Friday (9/27/2019). Please submit both rockpaperscissors.c and rockpaperscissors-enhanced.c to the assignment if you choose to complete the optional bonus.

Grading Rubric

- If you are not present and attending lab during Part 1 and Part 2 of this lab on Tuesday, you will not receive credit for the planning portion of this lab.
- If your document for Part 2 is not submitted successfully to Canvas by the due date, you will not receive credit for the planning portion of this lab.
- If you do not check in your code with a TA for Part 3 of this lab on Thursday, you will not receive credit for the coding portion of this lab!
- If your code for Part 3 is not submitted successfully to Canvas by the due date, you will not receive credit for the coding portion of this lab.
- Your assignment will be graded out of 100 points. The approximate grading distribution is:
    - Brainstorming document completeness/accurate
        - Logic covers all requirements specified — 10 points
        - Document is simple and easy to follow — 10 points
    - Program rockpaperscissors.c
        - Program operates correctly — 20 points
        - Output is formatted correctly — 20 points
    - Program compiles without errors or warnings — 10 points
    - Proper code formatting and commenting (See code style section on Canvas) — 30 points
    - Optional bonus exercise rockpaperscissors-enhanced.c — up to + 10 bonus points
        - Requires rockpaperscissors.c is submitted

Additional Resources

- If you need some help with random numbers, they are covered in zyBooks Chapter 2.19.