

Honors Computer Science C++

Mr. Clausen

Program 14A

PROGRAM 14A Rock Paper Scissors (Lizard Spock) (30 points : 5 Extra Credit points for Lizard Spock)

This program is going to practice functions, if statements, random numbers, loops, and functions (**remember each function only does ONE TASK**). Write a program to play the game Rock, Paper, Scissors (RPS), OR Rock, Paper, Scissors, Lizard, Spock (RPSLS). The user will be playing against the computer. Ask the user for their choice, generate a random choice for the computer, determine who wins or if there is a tie, keep a running total of who wins each round, and display the results each time the user chooses to play the game. This program can be written using if, else if statements to determine the winner, and to convert from the user's choice to a number and to convert the computer's numeric choice to a corresponding string representing the computer's choice.

- 1) Type comments at the beginning of the program to display your name and other information just like those used for program 4A. **Be sure to change the program name, program number, and program description.**

- 2) Include the preprocessor directives:

```
#include <iostream>          //necessary for cout and cin
#include <iomanip>             //necessary for setw
#include <cstdlib>            //necessary for random numbers
#include <ctime>              //necessary for the clock to seed the random numbers
//The string class is built in to standard C++. You don't need to "include" anything.
```

- 3) Type: using namespace std;
- 4) Next declare all of the constants necessary for this program. Declare a constant of type **int** for LOW = 1, and declare a constant of type **int** named HIGH = 5 if you are writing RPSLS. If you are writing RPS, then HIGH = 3.
- 5) Declare the **int** main() function here.
- 6) Declare all of the variables necessary for this program. Make sure that you use descriptive variable names to create readable, self-documenting code.
- 7) Include a 'do...while' loop in the int main() to repeat the function calls until the user enters a 'Q' or 'q' to quit the program. You will also need to add an "if statement" later so when the user enters a 'q' or 'Q' the game doesn't play and we don't see any scores.
- 8) Type the following comment:

```
//-----Display My Information-----
```

Follow this comment with the function Display_My_Info() that uses cout statements to display

your name and period output just like those used for program 4A. **Be sure to change the program name, and program number.**

- 9) You must use functions throughout this program. You must have a main function that declares all the variables (except variables that may be local to another function), and calls all the other functions. You will need one function for the menu (which gets the user's choice), one function to get the computer's choice (generate a random number for this), one function to convert the player's character choice from characters to names ('1' is rock, etc.), one function to convert the computer's choice from integer to names (1 is rock, etc.), one function to determine the outcome (and does nothing else except return a variable with the outcome: player won, computer won, or tie), one or three function(s) to count how many times the player and computer won and number of "ties", and one function to display the results. **Do not use a "Control_Menu_Execution function. All functions are called from inside the do...while loop in the main function.**
- 10) Display the menu (as pictured below – remove two choices if you are only writing RPS) and ask the user to enter their choice. Upper and lower case values for 'q' or 'Q' should work for the menu choice to quit.

```
Rock Paper Scissors Lizard Spock Main Menu
1. Rock
2. Spock
3. Paper
4. Lizard
5. Scissors
Q. Quit the game
```

- 11) After the user has selected their choice, generate a random integer to represent the computer's choice. Don't forget to use: **srand(time(0));** when generating random numbers.
- 12) Determine who won or if there is a tie.
- 13) Keep a running total of how many times the player wins, the computer wins, of if there is a "tie". This can be done with one function, or with 3 separate functions – one for each (player wins, computer wins, tie).
- 14) Display who chose what, who won (or if there was a tie), and how many times the "player" won and how many times the "computer" won, the number of ties every time the player makes a choice other than to quit. The output should look something like the picture below after every choice except to quit the game.

```
1) Rock
2) Spock
3) Paper
4) Lizard
5) Scissors
Q) Quit the game

Please enter your choice: 1

Player choice:   Rock
Computer choice: Spock
Winner:         computer

Player wins:     0
Computer wins:   1
Ties:            0
```

15) Save your program as LastNameFirstNameP14.cpp.

Use blank lines to separate each of the program sections listed in all of the steps above. When you are finished with your program, have tested it thoroughly to make sure that your calculations are correct, and are sure that you don't need to make any changes, then save your program in the "T" network mapping, in the Program 14 folder.