

Project Title		Rock-Paper-Scissors Game Against the Computer.	
Course Title		CPEG 1611	
S#	Student Name	Registration Number	
1	Muhammad Shahmir Abdullah	BCPE-243030	
2	Zunaira Chaudhry	BCPE-243025	
3	Hania Shakeel	BCPE-243015	



Capital University of Science & Technology

Department of Electrical and Computer Engineering

Project Report

Project Title: Rock-Paper-Scissors Game Against the Computer

Idea:

The “Rock, Paper, Scissors” game project is a simple yet engaging program where a user competes against the computer in the classic hand game. The player selects one of the three options—rock, paper, or scissors—while the computer makes a random choice. The game then determines the winner based on the rules: rock beats scissors, scissors beats paper, and paper beats rock. The project can include features like a scoring system to track multiple rounds, error handling for invalid inputs, and a user-friendly interface. For an advanced version, the game can be expanded with a graphical interface, animations, or smarter AI-driven computer decisions. This project is ideal for beginners to practice coding fundamentals like conditionals, loops, and randomization.

Objectives:

1. The project aims to create a Rock, Paper, Scissors game where the user competes against the computer.
2. It teaches basic programming concepts like conditionals, loops, and functions.
3. The computer’s choice is generated randomly to add unpredictability.
4. The game determines the winner based on standard rules and displays the result.
5. Players can extend the project with features like scoring systems or graphical interfaces for added fun.

Applications:

1. Helps beginners learn basic programming skills.
2. Provides a simple and fun game for entertainment.
3. Acts as a foundation for building advanced games or features.

Block Diagram:

5. **Basic Algorithm Design:** The project follows a simple algorithm that checks inputs, compares them, and displays the results, making use of a clear, sequential flow of operations.

Code:

```
#include <iostream>
#include <cstdlib>
#include <ctime>
Using namespace std;

// Function to get the computer's choice
String get_computer_choice() {
    Int random_choice = rand() % 3; // Generates a number between 0 and
2
    If (random_choice == 0) {
        Return "rock";
    } else if (random_choice == 1) {
        Return "paper";
    } else {
        Return "scissors";
    }
}

// Function to determine the winner
String determine_winner(string player, string computer) {
    If (player == computer) {
        Return "It's a tie!";
    } else if ((player == "rock" && computer == "scissors") ||
        (player == "scissors" && computer == "paper")) ||
```

```
        (player == "paper" && computer == "rock")) {  
    Return "You win!";  
} else {  
    Return "Computer wins!";  
}  
}
```

```
Int main() {  
    Srand(time(0)); // Initialize random seed for random number generation  
  
    Cout << "Welcome to Rock, Paper, Scissors!" << endl;  
  
    While (true) {  
        String player_choice;  
        Cout << "Enter rock, paper, or scissors (or 'quit' to stop): ";  
        Cin >> player_choice;  
  
        // Exit the game if the player types 'quit'  
        If (player_choice == "quit") {  
            Cout << "Thanks for playing!" << endl;  
            Break;  
        }  
  
        // Validate input  
        If (player_choice != "rock" && player_choice != "paper" &&  
player_choice != "scissors") {  
            Cout << "Invalid input, please try again." << endl;  
            Continue;  
        }  
    }  
}
```

```
// Get computer's choice
String computer_choice = get_computer_choice();
Cout << "Computer chose: " << computer_choice << endl;

// Determine and display the winner
String result = determine_winner(player_choice, computer_choice);
Cout << result << endl;
}

Return 0;
}
```

Result:

Compile Result

```
Welcome to Rock, Paper, Scissors!
Enter rock, paper, or scissors (or 'quit'
to stop): rock
Computer chose: scissors
You win!
Enter rock, paper, or scissors (or 'quit'
to stop):
```

References:

1. **C++ Basics:** Learn about conditionals, loops, and I/O from W3Schools C++.

2. **Random Numbers:** Use <cstdlib> and <ctime> for randomness (GeeksforGeeks Guide).
3. **Sample Code:** Explore examples on GitHub.

1	Project Title			
2	Lab	CYG1611- Applications of Information and Communication Technologies Lab	Semester	Fall 2024
3	Student Name & Registration No.			
4	Instructor Name & Signature	Mr. SM Waqas Ayub Shah		

Project Demonstration

Assessment Criteria	Very Poor 0-1	Poor 2-3	Satisfactory 4-5	Good 6-8	Excellent 9-10	Score Student 1
Design Evaluation and Testing	No or very poor design prototype and demonstration.	Design prototype is not working, and no testing of design has been done	The design prototype is partially functional and little testing of design has been done.	Design prototype is functional, and some testing of design has been done.	The design prototype is fully functional, and the design has been exhaustively tested.	
Usage of software tools (Visual Studio, MS Office Applications) in design and evaluation	No or very poor software tool (Visual Studio, MS Office Applications) usage in project design and results evaluation	Insignificant evidence of software tool (Visual Studio, MS Office Applications) usage in project design and results evaluation	Little evidence of ability to select appropriate software tools (Visual Studio, MS Office Applications), in project design and results evaluation	Some evidence of skills to use software tools (Visual Studio, MS Office Applications) in project design and results evaluation	Clear evidence of skills to use software tools (Visual Studio, MS Office Applications) in project design and results evaluation	

Project Report

Assessment Criteria	Very Poor 0-1	Poor 2	Satisfactory 3	Good 4	Excellent 5	Score Student 1
Literature Survey, Problem Analysis and Design Procedure	No or very poor literature survey done. No problem analysis performed. Nonworthwhile design procedure exists.	Insufficient literature survey. Problem analysis part is skipped or does not contribute to creating an effective design. Does not follow any design procedure.	Partial literature survey. Problem Analyses performed is haphazard and design parameter selection is spontaneous. Little use of design procedure.	Adequate literature survey. Problem analysis performed correctly. Project demonstrates some use of design process.	Clear and complete literature survey, effective problem analyses is performed to choose design parameters. Project demonstrates effective use of design process.	
Language, Grammar and References	A lot of spelling and grammatical mistakes with poor English. The list of references are clearly inadequate. Table of content missing.	Frequent spelling and grammatical errors. The list of references should be expanded.	Occasional spellings and grammatical errors. The list of references appears reasonable, but citation does not follow the standard format.	Very few spelling and grammatical errors. Organization is good. The list of references appears reasonable and citation follows standard format.	Almost no spelling or grammatical mistakes. Excellent organization. A comprehensive list of references are cited using the standard format.	

Viva Voce

Assessment Criteria	Very Poor 0-1	Poor 0-2	Satisfactory 0-3	Good 0-4	Excellent 0-5	Score Student 1
Knowledge of Project Implementation details (Q/A)	No or very poor knowledge of implementation and design process.	Poor knowledge of implementation and design with wrong/no answers	Satisfactory knowledge of implementation, vague answers	Adequate knowledge of project implementation with majority of correct answers	Exceptional knowledge of implementation and overall design with clear and spontaneous answers.	

