

# C-codePlay

Thapathali Campus

Institute of Engineering

Ajay Kumar Pasi[THA081BEI003]

Devendra Yadav[THA081BEIXXX]

Bishal Upadhayay[THA081BEI010]

# Introduction

- This project is a **collection of 5 mini-games** developed in **C programming language**.
- The games are designed to be **fun, interactive, and educational**, showcasing the team's programming and problem-solving skills.
- The project includes:
  - **Number Guessing Game**
  - **Rock-Paper-Scissors Game**
  - **Tic-Tac-Toe Game**
  - **Hangman Game**
  - **Room Escape Game** (Riddle-based)
- The project is **menu-driven**, allowing users to either **choose a game** or play a **randomly selected game**.

# Objectives

- To create a **multi-game program** that provides entertainment and challenges for users.
- To demonstrate **proficiency in C programming** and problem-solving.
- To explore **game development concepts** such as randomization, user input handling, and game logic.
- To incorporate **educational elements** (e.g., riddles in the Room Escape Game).
- To showcase **teamwork and creativity** in designing and implementing the games.

# Motivation Behind the Project

- **Passion for gaming:** The team wanted to create something fun and engaging.
- **Learning opportunity:** The project provided a hands-on way to learn and apply programming concepts.
- **Real-world application:** Games are a great way to practice logic, algorithms, and user interaction.
- **Innovation:** The project combines multiple games into one program, offering variety and replayability.
- **Team collaboration:** Working together to design, develop, and debug the games was a rewarding experience.

# Overview of the Games

- **1. Number Guessing Game:** A simple game where the player guesses a randomly generated number.
- **2. Rock-Paper-Scissors:** A classic two-player game against the computer.
- **3. Tic-Tac-Toe:** A two-player game where players take turns marking a 3x3 grid.
- **4. Hangman:** A word-guessing game where the player has limited attempts to guess the word.
- **5. Room Escape Game:** A riddle-based game where the player must solve riddles to "escape" the room.

# Number Guessing Game

- **Objective:** The player must guess a randomly generated number between 1 and 100.
- **Gameplay:**
  - The program generates a random number using the **rand()** function
  - The player inputs guesses, and the program provides feedback (too high, too low, or correct).
  - The game ends when the player guesses the correct number.
- **Technical Details:**
  - uses `srand(time(0))` to seed the random number generator.
  - Loops until the correct guess is made, counting the number of attempts.
- **Challenges:**
  - Ensuring the random number is within the correct range.
  - Handling invalid input (e.g., non-numeric input).

# Rock-Paper-Scissors Game

- **Objective:** The player competes against the computer in a game of Rock-Paper-Scissors.
- **Gameplay:**
  - The player chooses Rock, Paper, or Scissors by entering a number (1, 2, or 3).
  - The computer randomly selects one of the three options.
  - The program compares the choices and declares the winner or a draw.
- **Technical Details:**
  - Uses rand() to generate computer's choice.
  - Implements game logic using conditional statements.
- **Challenges:**
  - Ensuring the computer's choice is random and unbiased.
  - Handling invalid input from the player.

# Tic-Tac-Toe Game

- **Objective:** Two players take turns marking a 3x3 grid, aiming to get three of their marks in a row.
- **Gameplay:**
  - The grid is displayed with numbers 1-9 representing positions.
  - Players input their chosen position to place their mark (X or O).
  - The game checks for a winner after each move.
- **Technical Details:**
  - Uses a 2D array to represent the game board.
  - Implements win-checking logic for rows, columns, and diagonals.
- **Challenges:**
  - Ensuring the game handles invalid moves (e.g., overwriting an existing mark).
  - Implementing the win-checking logic efficiently.



# Hangman Game

- **Objective:** The player must guess a hidden word by suggesting letters within a limited number of attempts.
- **Gameplay:**
  - A random word is selected from a predefined list.
  - The player guesses letters one at a time.
  - Correct guesses reveal the letter in the word, while incorrect guesses reduce the number of remaining attempts.
- **Technical Details:**
  - Uses an array of strings to store the word list.
  - Tracks guessed letters and updates the displayed word accordingly.
- **Challenges:**
  - Handling repeated guesses and invalid input.
  - Ensuring the game ends correctly when the word is guessed or attempts run out.

# Room Escape Game

- **Objective:** The player must solve a series of riddles to "escape" the room.
- **Gameplay:**
  - Riddles are loaded from a text file **riddle.txt**
  - The player must answer a set number of riddles correctly to win.
  - The game provides feedback on each answer and tracks the number of correct answers.
- **Technical Details:**
  - Uses file I/O to read riddles and answers from a text file.
  - Implements case-insensitive comparison for answers.
- **Challenges:**
  - Ensuring the file is correctly formatted and accessible.
  - Handling user input and providing meaningful feedback.

# Technologies and Tools

- **Programming Language: C**
- **Libraries/Functions Used:**
  - Stdio.h: For I/O functions.
  - Stdlib.h: For random number generation.
  - String.h: For string manipulation.
  - Time.h: for seeding random number generator

# Challenges and Solutions

- **Challenge 1:** Handling invalid user input.
  - **Solution:** Implement input validation and provide clear error messages.
- **Challenge 2:** Ensuring randomness in games like Rock-Paper-Scissors and Hangman.
  - Use `srand(time(0))` to seed random number generator.

**Challenge 3:** Managing game state and logic in Tic-Tac-Toe.

- **Solution:** Use a 2D array to represent the board and implement win-checking logic

# Conclusion

- The Multi-Game Project is a **fun and educational program** that showcases the team's programming skills.
- It includes **5 unique games**, each with its own challenges and logic.
- The project demonstrates **teamwork, creativity, and technical proficiency**.
- Thank you for your attention! We welcome any questions or feedback.

# Q&A

- A simple slide to invite questions from the audience