

Report on Server-Client Code

Work Distribution:

- **Faizal:** Implemented Maze Escape Game, merged each function to one server client & Made the README file.
- **Yash:** Implemented Snake Game and Polynomial Root Solver, converted codes for server-client implementation.
- **Patil:** Implemented Differential Equation Solver and System of Linear Equation Solver.
- **Sunny:** Implemented Tic Tac Toe and contributed to README and beautification of server-client coding.

Introduction:

The provided code implements a server-client architecture for various games and mathematical problem-solving scenarios. It allows clients to connect to the server and choose from a menu of different games and mathematical tasks to interact with. The server responds to client requests, facilitates gameplay, and performs computations as needed.

Implemented Features:

Tic Tac Toe (Game 1):

- Players can play Tic Tac Toe against each other.
- The server manages the game state, validates moves, and determines the winner or a draw.
- Communication between server and client involves sending game state and receiving player moves.

Snake and Ladder (Game 2):

- Players can play the Snake and Ladder game against each other.
- The server simulates the game, handles player turns, and updates the board.
- Communication involves sending board state and receiving dice rolls from clients.

System of Linear Equations Solver (Game 3):

- Clients can solve a system of linear equations.
- The server performs Gaussian elimination to solve the system and sends the solution back to the client.
- Communication includes sending coefficients and constants of equations and receiving the solution.

Polynomial Equation Solver (Game 4):

- Clients can solve polynomial equations using various methods like bisection, Newton's method, and secant method.
- The server computes roots using the chosen method and sends them back to the client.
- Communication involves sending equation details and receiving roots.

Maze Escape Game (Game 5):

- Game 5 is implemented but not fully defined in the provided code. It only receives an integer from the client.
- Additional functionality needs to be implemented for this game.

Differential Equation Solver (Game 6):

- Clients can solve ordinary differential equations using the Runge-Kutta method.
- The server calculates the solution and sends it back to the client.
- Communication involves sending equation details and receiving the solution.

Functionality and Communication:

- The server waits for client connections and communicates using TCP/IP sockets.

- It listens for client requests, receives inputs, performs computations, and sends appropriate responses.
- Error handling is implemented for socket creation, binding, listening, and client acceptance.

Conclusion:

The provided server-client code demonstrates the implementation of a versatile system capable of hosting various games and mathematical problem-solving tasks. It allows clients to interact with the server, engage in gameplay, and receive solutions to mathematical problems. However, some games like the Maze Escape Game require further implementation to be fully functional. Overall, the code provides a foundation for building a robust server-client system for gaming and mathematical applications.