Complex Computing Problem (CCP) Proposal

Title: ***Simplified Battleship Game***
Team Name: ***Army of Coders***
Group Members:

- **Hans Raj (CT-187)**
- **Ayesha Aqil (CT-175)**

Problem Statement

Traditional games like Battleship help improve strategic thinking, but playing them physically requires a board, markers, and two players in the same place. This can be difficult and time-consuming. A computerized version of the game can provide the same fun and challenge in a simpler, faster, and more interactive way.
The problem is to create a **simplified Battleship game using C language** that allows users to place ships on a grid and then guess positions to hit or miss ships until the game is over.

Objectives

- To design a 2D grid (array) to represent the game board.
- To allow placement of ships on the board using arrays.
- To take user input for selecting coordinates (row, column).
- To detect hits and misses using conditional logic.
- To keep track of the number of attempts until all ships are sunk.
- To provide an interactive game experience through C programming basics.

Scope

- The project will use a **2D array** for the board representation.
- A limited number of ships (e.g., 3 ships) will be placed randomly or manually.
- The user will guess positions one by one until all ships are destroyed.
- The system will show results of each attempt (hit or miss).

- Only one player will play against the system in this simplified version.
- The project will demonstrate the use of arrays, loops, conditions, and functions.

Proposed Solution

- Use a **2D array** (for example, 5x5 or 10x10 grid) to represent the game board.
- Assign values for ships (e.g., 1 for ship, 0 for empty cell).
- Take user input for guessing positions (row and column).
- Use **loops and conditions** to check whether the guess is a hit or a miss.
- Update the board after each attempt and display the result.
- Continue the game until all ships are sunk.
- Optionally, allow replaying the game without restarting the program.

Complexity Justification

This is a **Complex Computing Problem** because it involves:

- Managing a **2D array** for storing and updating board data.
- Implementing **logic for placement, searching, and hit detection**.
- Using **loops** for repeated guesses and gameplay flow.
- Applying **conditional statements** for decision-making (hit or miss).
- Designing a **reusable and extensible program** (can add larger boards, more ships, multiplayer in future).

Expected Outcomes

- A working **simplified Battleship game** in C language.
- A practical example of applying arrays, loops, conditions, and structured logic.
- Improved problem-solving and programming skills through game development.

- A fun and interactive project that demonstrates **core programming concepts** in action.
