PROJECT PROPOSAL

23K-0820 Usman Ahmed

23K-0650 Muhammad Harmain

Introduction:

The game of Battleship, is a classic strategy-based board game. It is a two-player game where each player attempts to sink the ships. Battleship offers a perfect blend of strategy, deduction, and excitement, making it an excellent choice for a programming project.

In our proposal, we aim to design and develop a Battleship game using the C programming language. This project will provide an opportunity to apply fundamental programming concepts and data structures while creating an engaging and interactive gaming experience.

Tools and Technologies:

1: The C language is used to design the project.

2: The text editor and compiler is Dev-C++.

3: The operating system is windows.

Salient Features:

1: Grid-Based Game Board: The game board is displayed as a grid where players will place their ships and track their attacks. A typical grid size is 10x10.

2: Ship Placement: Players can place their ships on the game board. (Will be decided while keeping in view the fact that we have limited time)

3: User Input: Implement a user-friendly interface that allows players to input their attack coordinates using a simple format (e.g., A1, B5).

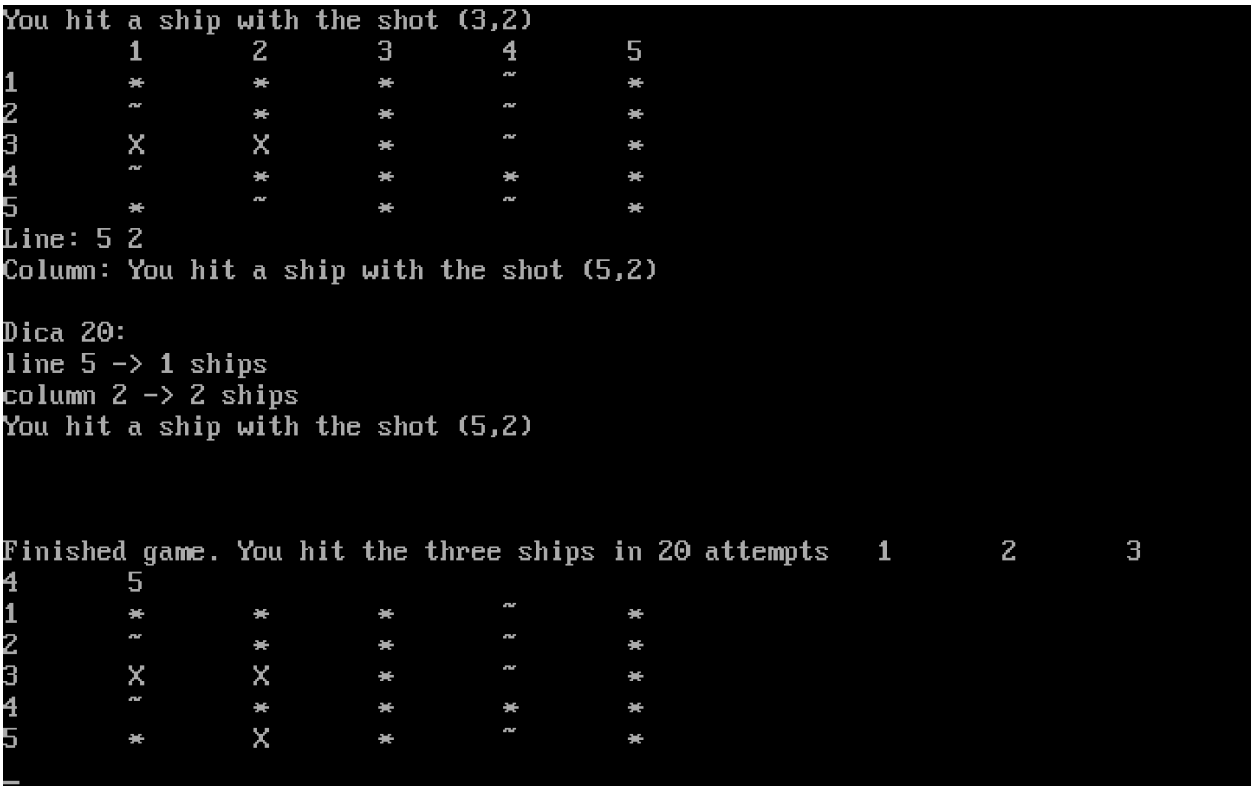
4: Hit and Miss Feedback: After each attack, provide clear visual feedback to indicate whether it was a hit or a miss.

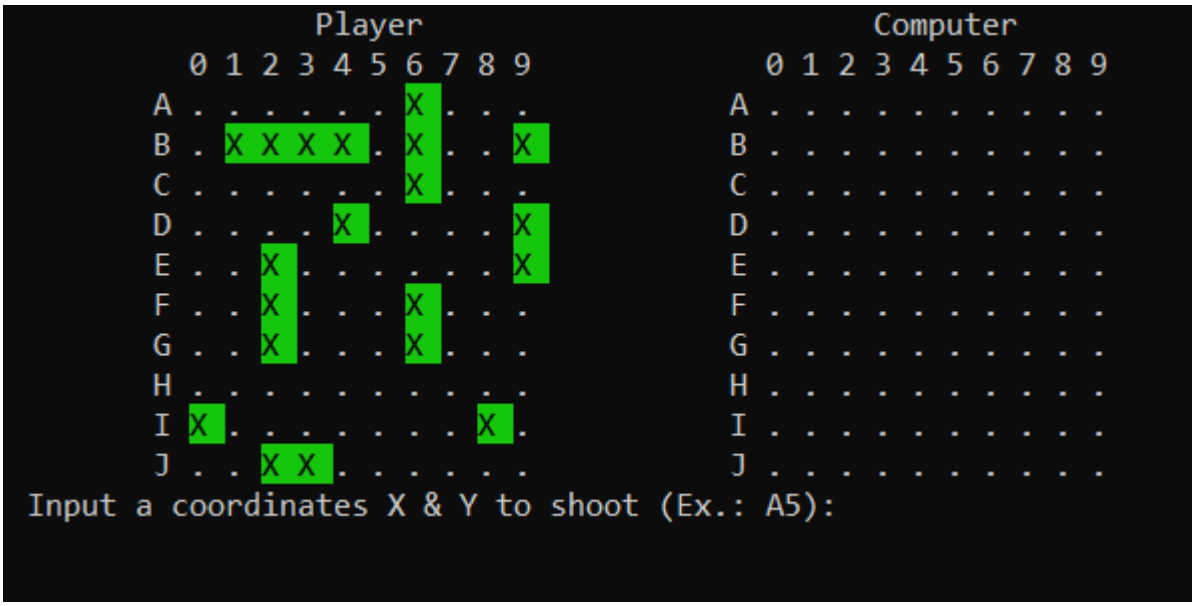
5: Simple AI Opponent (Optional): If you decide to include an AI opponent for single-player mode, keep its behavior straightforward. It should make random moves without employing advanced strategies.

6: Game Ending Screen: When the game ends, display a game-over screen that announces the winner and allows players to start a new game or return to the main menu.

7. Main menu: Displays primarily one choice (That is single player mode with computer) and secondarily two choices where the second choice is two player mode. (second choice is optional) along with game credentials option and exit game option.

Existing Systems:





Problem Statement:

Above are a few programs analogous to this project. It is expected that our program will be a more refined extension of both the programs including some additional features which will enhance the overall UI/UX, thereby making a more engaging program.

Proposed solution:

New features will be incorporated as follows

1) Pre-Menu: We will make use switches in the beginning giving the user the options to either play the game, view credentials or exit. (within the play game option there might be further options for the user as to play the game against computer or player which is subject to time. If we intend to incorporate this feature, then we will make use of nested switches)

2) Grids and mapping of terrain: This will be achieved using 2D arrays

3) Hit, miss and turns status: This will be operated using loops

4) For random: For making the computer generate random values of x row and y column we will make use of a random function.

5) For 2 player game and mapping controls: We will do research

6) Post-Menu: We will make use switches at the end of the gameplay giving the user the options to either replay or exit the game.

7) Input: For taking input for attack from the user we will either take it in the form of program 1 or program 2, whichever seems optimal.

8) For cancelling out boxes that have been hit by the players we will do research so that their status remain inactive after player hits them once.