VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

COMPUTER ARCHITECTURE (CO2007)

Assignment (Semester: 231)

BATTLESHIP

Advisor: Băng Ngọc Bảo Tâm, CSE-HCMUT
Student: Trần Đình Đăng Khoa - 2211649.

HO CHI MINH CITY, NOVEMBER 2023

Contents

1 Introduction

Battleship (also known as Battleships or Sea Battle) is a strategy type guessing game for two players. It is played on ruled grids (paper or board) on which each player's fleet of warships are marked. The locations of the fleets are concealed from the other player. Players alternate turns calling "shots" at the other player's ships, and the objective of the game is to destroy the opposing player's fleet.

Figure 1: Example of the classic battleship game

1.1 Description

The game is played on four grids, two for each player. The grids are typically square - usually 10×10 - and the individual squares in the grid are identified by letter and number. On one grid the player arranges ships and records the shots by the opponent. On the other grid, the player records their own shots.

Before play begins, each player secretly arranges their ships on their primary grid. Each ship occupies a number of consecutive squares on the grid, arranged either horizontally or vertically. The number of squares for each ship is determined by the type of ship. The ships cannot overlap (i.e., only one ship can occupy any given square in the grid). The types and numbers of ships

allowed are the same for each player. These may vary depending on the rules. The ships should be hidden from players sight and it's not allowed to see each other's pieces. The game is a discovery game which players need to discover their opponents ship positions.

After the ships have been positioned, the game proceeds in a series of rounds. In each round, each player takes a turn to announce a target square in the opponent's grid which is to be shot at. The opponent announces whether or not the square is occupied by a ship. If it is a "hit", the player who is hit marks this on their own or "ocean" grid (with a red peg in the pegboard version), and announces what ship was hit. The attacking player marks the hit or miss on their own "tracking" or "target" grid with a pencil marking in the paper version of the game, or the appropriate color peg in the pegboard version (red for "hit", white for "miss"), in order to build up a picture of the opponent's fleet.

If all of a player's ships have been sunk, the game is over and their opponent wins.

2 MIPS Assembly Implementation

2.1 Introduction

In this project, we aim to emulate the Battleship game using the *MIPS* assembly language. Due to resource constraints and to keep things simple, we will work with a smaller grid size which is **7**×**7**

A ship location is indicated by the coordinates of the bow and the stern of the ship (row_{bow}, $\text{column}_{\text{bow}}$, $\text{row}_{\text{stern}}$, $\text{column}_{\text{stern}}$).

2.2 Overall Structure

Describe the overall structure of the code.

Highlight the main sections, such as data section (.data), text section (.text), and various segments like the game menu, rules screen, ship placement, and shooting.

2.3 Constants and Definitions

Explain the purpose of constants defined using .eqv (e.g., system calls, characters, and game-related constants).

2.4 Data Section

Describe the data section (.data) and the defined constants.

Discuss the purpose of player maps, grid size, ship and shot coordinates, and the title/rules strings.

2.5 User Interface

Explain how the game menu is displayed to the user.

Discuss how the rules screen is presented with ASCII art.

2.6 Game Logic

Analyze the game flow from the main function.

Discuss how players take turns placing ships and shooting, including input validation.

2.7 Helper Functions

Describe the purpose of each function $e.g., game_menu, rules_screen, read_ship, read_shot.$

Highlight the functionality of key functions and their interaction.

2.8 Input Validation - Error Handling

Discuss how the code validates user input for ship and shot coordinates.

Analyze the error messages and how they guide the user.

2.9 Gameplay Logic

Explain the sequence of events during ship placement and shooting.

Discuss how ships are drawn on the maps and the conditions for hitting or missing.

3 Conclusion

References

[1] Wikipedia. (n.d.). *Battleship (game)*. https://en.wikipedia.org/wiki/Battleship (game) Accessed 2023.