Ho Chi Minh City University of Technology

Faculty of Computer Science and Engineering
Course: Computer Architecture Lab (CO2008)

Assignment

BATTLESHIP

Ho Chi Minh City, October 23^{rd} 2023

Contents

1 Outcomes

After finishing this assignment, students can proficiently use:

- MARS MIPS simulator.
- Arithmetic & data transfer instructions.
- Conditional branch and unconditional jump instructions.
- Procedures.

2 Introduction

Battleship is a strategic board game between 2 players. It focuses on the process of planning deducting skill of both players. The game rules can be found in [1], in summary:

1. In a classic game, each player sets up a fleet of battleships on their map (a 10x10 grid). A fleet must contain a predefined set of battleships with different sizes. For example, a fleet of ships can consist of 5 2x1 ships, 3 3x1 ships, 1 5x1 ship.

Figure 1: Example of the battleship game in real life

2. 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.

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

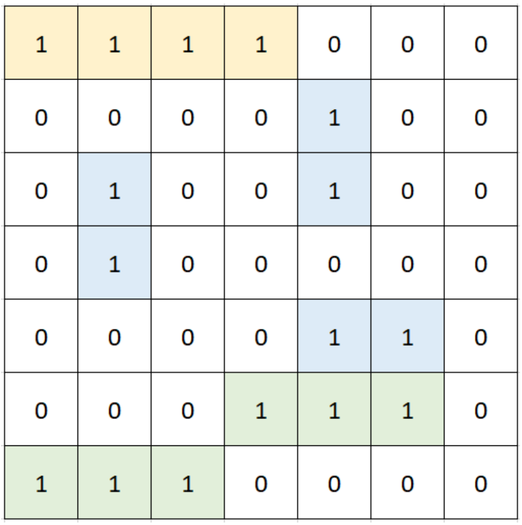
For more details or inquiries related to the assignments, please contact your lab's instructor and Mr. Nguyen Thien An (ngthienan.cse@hcmut.edu.vn) by emails.

3 Requirements

In this project, we will replicate the battleship game in MIPS assembly language. Since the resource is limited and to keep things simple, we will work with a smaller grid size which is **7x7**. Please create a program that does the following:

- A note before entering the program: A ship location is indicated by the coordinates of the bow and the stern of the ship (row_{bow}, $column_{bow}$, row_{stern}, $column_{stern}$). In Figure 2, the coordinates of the yellow 4x1 ship is "0 0 0 3"; the blue 2x1 ships are "1 4 2 4", "2 1 3 1", and "4 4 4 5"; the green 3x1 ships are "5 3 5 5" and "6 0 6 2".

- We begin with the preparation phase where the players set up their ship locations. The program should prompt to ask the players to insert the location of the ships. The values of the cells of the map will be either **1** or **0**. 1 means that the box is **occupied by a ship**, otherwise it's 0. The players will need to setup the exact amount of ships with the same size in order to complete the setup phase. In detail, each player will have 3 **2x1** ships, 2 **3x1** ships, and 1 **4x1** ship. Note that the ships can not overlap with each other. An example of an acceptable arrangement is shown in Figure 2.



1	1	1	1	0	0	0
0	0	0	0	1	0	0
0	1	0	0	1	0	0
0	1	0	0	0	0	0
0	0	0	0	1	1	0
0	0	0	1	1	1	0
1	1	1	0	0	0	0

Figure 2: Example of the the ships that a player can place in this project

- After the setup phase, both players will **take turn** targeting a box on the other player's map blindly. If the attack hits a cell that is occupied by a ship, an announcement should pop up in the terminal and says "HIT!". This will allow the attacking player to know if they hit an opponent's ship or not. The players will need to remember the cells that have been hit themselves. If a cell got hit, its value should change into **0**, as there it is no longer occupied by a ship. A ship is completely destroyed if all of the cells it's occupying are targeted. A player will lose if all of their ships got destroyed first. In other words, the game will end when a player's board contain **only 0's**.
- Note: Bonus points are provided if you can write out the moves of each player into a separate text file for ease of review.

4 Submission

Students are requested to submit the MIPS program(s)/source code (.asm files) and the Assignment report to BK E-learning system (BKEL) no later than the last lab session of your group. Assignment must be done individually. Students have to demonstrate program(s) on MARS MIPS during the last lab session. Students who do not show up during the demonstration time will get 0 for the assignment. The report should not contain code. Instead, students should present the algorithms as well as the idea in your implementation.

5 Plagiarism

Similarity less than 30% in MIPS code is allowed. In other words, you will get 0 for assignment if your answers are similar to another student's more than 30%. Note that, we will use the MOSS tool developed by Stanford for checking similarity (https://theory.stanford.edu/~aiken/moss/).

6 Rubric for evaluation

6.1 Friendly interface - 2 points

Students can design and implement an amicable user interface so that players can play easily without any confusion (2 points).

Students can design and implement a friendly user interface; however, players face some difficulties when playing the game (1.5 points).

Students can design and implement a user interface, but it is not friendly, or players need to do several steps for one move (1 point).

Student can design and implement a user interface, but it fails to allow playing (0.5 points).

6.2 Application implementation - 6 points

Students can implement an excellent application without any errors found (5.0 - 6 points).

Students can implement a good application with some minor errors, but players do not need to restart the application to continue (4.0 - 5.0 points).

Students can implement the application with some errors that prevent players from playing the game (2.0 - 4.0 point).

Students cannot implement the application so that players can play/run (0 - 2.0 points).

6.3 Report - 2 points

Students write such an excellent report that others can understand without any difficulty (2 points).

Students write a good report but it's quite simple or lack of some information to clarify the implementation (1.0 - 1.5 points).

Students write a report with a lot of code embedded without any explanation (0.5 - 1.0 points).

Students with no report submitted (0 point).

References

[1] Wikipedia, “Battleship (game).” [Online]. Available: https://en.wikipedia.org/wiki/Battleship_(game)