

CARLOS III UNIVERSITY OF MADRID



Battleship

Computer Programming

BSc in Statistics and Business

Table of Contents

Problem Specification 3

Game rules 5

Option 1: File input..... 5

Option 2: Keyboard input..... 5

Option 3: Statistics 6

Option 4: Quit the game 6

Program Code..... 6

Documentation 6

Submission rules..... 7

Problem Specification

Battleship is a game developed on a 10 x 10 square matrix. The fleet consists of:

- 1 boat occupying 5 squares
- 2 boats occupying 4 squares
- 2 boats occupying 3 squares
- 3 boats occupying 2 squares
- 4 boats occupying 1 square.

Boats can only be placed vertically or horizontally (not diagonally). The boats should not come into contact with each other (either horizontally, vertically or diagonally), but they may touch the edges of the board. Build a program for two users to play Battleship.

The program must:

1.- Display a main menu that starts the game and includes the following options from which the user can choose:

A) Option 1: One-Player Game, reading the initial position of the fleet from an external file in the format specified below.

B) Option 2: Two-Player Game, where each user inputs the positions of their fleets by keyboard. This will be the fleet against which the other user will play.

C) Option 3: Statistics

D) Option 4: Quit the Game.

2.- The game begins after the selection of either Option 1 or Option 2. The player alternately take shots until the opponent's entire fleet is sunk or else surrenders. In the event of surrender, the program shows the position of the fleet and returns to the main menu.

3.- In each round of Options 1 and 2, the program must:

- Display a message indicating whether the shot is a: "MISS" if the square is not occupied by any boat; "HIT" if the square is occupied by a ship; "SUNK" if all the squares occupied by the boat have been hit.
- Display the current state of the board:
 - "b": if there is no boat occupying the position
 - "x": if there is a boat occupying the position
 - "*": asterisks in the positions occupied by a sunken ship
 - "H": if the boat has been "HIT" but not sunk.
- Example of how the board should be displayed to the user:

b	b	x	b	b	b	x	x	b	b
x	b	b	b	x	b	b	b	x	b
b	b	b	b	b	b	b	b	b	b
b	b	b	b	b	x	x	x	x	b
b	b	b	x	b	b	b	b	b	b
x	b	b	x	b	b	x	x	x	x
x	b	b	b	b	b	b	b	b	b
x	b	b	b	x	x	x	b	b	x
x	b	b	b	b	b	b	b	b	x
x	b	b	x	b	b	x	x	b	x

- When all ships have been sunk:
 - Display the message: "SUNKEN FLEET"
 - Display the user's game board (s).
 - Display the statistics of the current game:
 - Total number of shots
 - Number of hits
 - Number of misses
 - Percentage of hits
 - Percentage of misses.
- Record the statistics to provide a game log.

- Record the final board in a file called "Output.xls".

In two-player games, a user will continue to play until the targeted position is a "Miss", and the turn will pass to the other player.

Game rules

1. Users should specify the row and column to announce their shots .The first indicator is a digit between 1 and 10 specifying the row and the second is a letter between A and J specifying the column.
2. When a user surrenders, the input should be a string with the single character "R".
3. If the input does not meet these conditions, it is considered invalid, and the data prompt will be repeated without this affecting the program state.
4. The game ends when the user has sunk the entire fleet or has surrendered.
5. Repeated shots targeting the same position do not need to be validated.

Option 1: File input

For the file input case (Option 1), the input file is called flota.xls. Suppose that the file exists and is not empty.

Option 2: Keyboard input

The data on each of the boats shall be input as follows:

- The position of each boat is made up of four characters with the following constraints:
 - The first is a digit between 1 and 10, specifying the row.
 - The second is a capital letter between A and J, specifying the column.
 - The third is a capital letter, H or V, specifying whether the boat is positioned horizontally (H) or vertically (V).
 - The fourth is a digit between 1 and 5, specifying how many squares the boat occupies.
- Check that the boats that are described fit on the board and do not come into contact with each other.

For example:

- 1BH3, this boat would be in positions: 1B -1C - 1D
- 2DV2, this boat would be in positions: 2D-3D

Option 3: Statistics

A statistics log will be displayed:

- Number of turns
- One-player games
- Two-player games
- For each player
 - Shots
 - Hits
 - Misses
 - Percentages

Option 4: Quit the game

Game is over.

Program Code

The program code must be fully commented. It should be saved to a file called "GameGroupxx.txt". Only one student from each group should upload the file.

Documentation

This is a Microsoft Word document, called "Memoria_GrupoXX.docx", including, at least, the following information:

- Cover sheet: Title of the practical assignment, student ID numbers, group number, student names and surnames
- Table of contents of the practical assignment report
- Description of the data structures used

- List of the functions included, briefly explaining: what each function does, the function input parameters, variables and content.
- Example of program operation: function calls; inputs and outputs.
- Program code must be commented.

Submission rules

The practical assignment is to be completed in groups of two students.

To be eligible for assessment, the following must be submitted:

- In electronic format:
 - The documentation of the practical assignment as specified in the Documentation section above
 - The program code.
- On paper:
 - The documentation as specified above.
- Practical assignment exam. Each group member has to sit an exam that will be graded individually.