

Each square in the 10x10 board is represented by its row (A-J) and its column (0-9). Each line gives the two ends of the ship. It starts with the starting square of the ship, followed by one space, then the ending square of the ship. The starting square will be the left end of the ship if it is placed horizontally or the top end of the ship if it is placed vertically. The five lines give the location of the 5 ships in the order of **a**, **b**, **c**, **s**, and **p**. The ships will not overlap, that is, any square can be occupied by at most one ship.

When the game starts, each player will take the shot by specifying the square to shot from the keyboard. A valid shot is a square on the board, a letter between A and J followed by a single digit between 0 and 9. For example, on A0, the starting end of the aircraft carrier will be hit; and shot I2 will miss.

Execution of the program and the Output

This project requires you write your output both to an output file and to the monitor.

1. print out the board with player 1's ship deployment on the monitor (See the master program for exact output format.)
2. prompt player 1 for confirmation by printing out on the monitor:
Player 1: confirm the ship deployment (Y/N):
3. on input Y, clear the monitor and repeat steps 1 and 2 for player 2.
4. terminate the program if any player enters N.
5. print out both players' shoot boards (not showing the locations of the ships): x for a hit, o for a miss, and - (minus sign) for squares have not been shot yet.
6. ask player 1 to shoot.
7. terminate the program if player 1 enters Q (for quit) as the row to shoot.
8. if player 1 enters an invalid square, ask the player to shoot again.
9. on a valid shoot selection, clear the monitor, tell player 1 whether the shot is a hit or miss
10. repeat steps 5-9 for player 2
11. repeat steps 5-10 for the next round of shoot until one player wins the game
12. clear the monitor and print out both players' shoot boards on the monitor and to an output file: x for hit, o for miss, - for square not being shot, and ship name (a,b,c,s,p) for square that the ship locates and not being shot.

Project Requirements:

1. You must program using C under GLUE UNIX system and name your program **p2.c**.
2. Submit your program **p2.c** electronically before the due time.
3. **IMPORTANT:** Your program's output, both to the output file and to the computer monitor, should be exactly the same as that produced by the master program.

Grading Criteria:

Correctness:	80%
Good coding style:	10%
Proper documentation:	10%
Late submission penalty:	-40% for the first 24 hours
	No submission will be accepted after the first 24 hours.