Battleship

The Navy needs your help! They are in desperate need for a suitable simulator that can be used to train new sailors. You and your team have been tasked with developing a new strategy based game that the sailors can use to hone their skills. You recall that you once played a game called Battleship as a kid.

For those that had the misfortune of not playing the game here is a synopsis:

1. This is typically a two player game each person has a grid that is similar to the images below where the columns are denoted by a letter and the rows a number

<div style="display:flex;">

Player 1 Board

	A	B	C	D	E	F
1						
2						
3						
4						
5						
6						

Player 2 Board

	A	B	C	D	E	F
1						
2						
3						
4						
5						
6						

</div>

2. Each person has a fleet of three ships (Battleship, Submarine, and Destroyer). Each of the ships are assigned a number of holes. The image below illustrates the breakdown of the number of holes for each type of ship.

BATTLESHIP—4 HOLES

SUBMARINE—3 HOLES

DESTROYER—2 HOLES

3. Each player will randomly distribute the three ships. The player is allowed the place ships horizontally or vertically but not diagonally. An example placement of both player's ships is shown below

Player 1 Board

	A	B	C	D	E	F
1	0	1	1	1	0	1
2	0	0	0	0	0	1
3	0	0	0	0	0	0
4	1	1	1	1	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0

Player 2 Board

	A	B	C	D	E	F
1	1	0	0	0	0	0
2	1	0	1	1	0	0
3	1	0	0	0	0	1
4	1	0	0	0	0	1
5	0	0	0	0	0	1
6	0	0	0	0	0	0

4. Each player takes a turn by giving a row and column combination to try to sink the opposing persons ship(s). After each player call the opposing player indicates a correctly guessed row and column (R & C) combination that contains part(s) of a ship by saying the word "Hit" (User gets unlimited turns so long as they continue to guess correct R & C combination that result in a "Hit") or "Miss" if the player's R & C combination did not contain any part of a ship(s). An example of a "Miss" would be if player one gives the R & C combination of "B1", this would result in player 2 saying "Miss" since that combination did not impact any of their fleet. Likewise, a "Hit" example would be if the same player would have given the R & C combination of "A1". Player 2 would have said "Hit" and player 1 would have been given another turn to try to give a R & C combination that would result in another "Hit".

5. The game continuous until one player has sunk all the other players' fleet (three ships). At the conclusion of the game the looser concedes defeat and the two players shake hands.

Program Requirements

Your team's task is to replicate this game given the following requirements:

1. Must use at least 2 functions

2. Must use two 2D arrays to act as each player's board.

3. The user must be able to choose their difficulty level (similar to minesweeper):

"e" for easy mode   (5 x 5 program board size)

"m" for medium mode (6 x 6 program board size) *as shown*

"h" for hard mode   (7 x 7 program board size)

4. Must use at least 3 pre-populated player boards per level (aka there should be 3 player 1 and player 2 board combinations that you manually develop)

5. Must use random number generator (to randomly choose populated game board for each player)

6. Must refresh terminal screen after each player's move

7. Must display "*" on every location on both player's board at the start of each game.

8. Must display a symbol (i.e. "X") on every "Hit" made by the player on each player's board.

9. Must accept user input for R & C combination(s)

10.   Must comment code and compile program in Linux

Additional information (aka Hints):

- Use a function to populate each board levels choice

- Store each player's board as 1s and 0s. For the example player boards shown above they would translate to:

Player 1 Board

	A	B	C	D	E	F
1		■	■	■		■
2						■
3						
4	■	■	■	■		
5						
6						

Player 2 Board

	A	B	C	D	E	F
1	■					
2	■		■	■		
3	■					■
4	■					■
5						■
6						

- Take the user input (B3) as an example and translate that into an index for the 2D arrays (i.e. B3 translates to (2,1) in the 2D array.

- "and" the users entered location with your 1s and 0s 2D array at the same location (The result will be a 1 if it is a hit or 0 if it was a miss.

Below is an example input and output for this project (based on the example boards shows above). →

Full program example output (and input)

>> Welcome to the Battleship Game Simulator! Please choose your level of difficuly (e = easy, m = medium, h = hard)

>> m

>> You have chosen a medium level game

>> Please enter in Player # 1's Name:

>> John

>> Please enter in Player # 2's Name:

>> Fred

>> Here is the current Gameboard:

```
            John's Board                      Fred's Board

         A   B   C   D   E   F            A   B   C   D   E   F

1    *   *   *   *   *   *        1    *   *   *   *   *   *

2    *   *   *   *   *   *        2    *   *   *   *   *   *

3    *   *   *   *   *   *        3    *   *   *   *   *   *

4    *   *   *   *   *   *        4    *   *   *   *   *   *

5    *   *   *   *   *   *        5    *   *   *   *   *   *

6    *   *   *   *   *   *        6    *   *   *   *   *   *
```

>> John's Turn, please enter a Location:

>> B2

>> I am sorry John that was a MISS!

>> Here is the current Gameboard:

```
            John's Board              Fred's Board

        A   B   C   D   E   F      A   B   C   D   E   F

1       *   *   *   *   *   *    1  *   *   *   *   *   *

2       *   *   *   *   *   *    2  *   *   *   *   *   *

3       *   *   *   *   *   *    3  *   *   *   *   *   *

4       *   *   *   *   *   *    4  *   *   *   *   *   *

5       *   *   *   *   *   *    5  *   *   *   *   *   *

6       *   *   *   *   *   *    6  *   *   *   *   *   *
```

>> Fred's Turn, please enter a Location:

>> F1

>> Congratulations Fred that was a HIT!

>> Here is the current Gameboard:

```
            John's Board                  Fred's Board

        A   B   C   D   E   F          A   B   C   D   E   F

1       *   *   *   *   *   X        1  *   *   *   *   *   *

2       *   *   *   *   *   *        2  *   *   *   *   *   *

3       *   *   *   *   *   *        3  *   *   *   *   *   *

4       *   *   *   *   *   *        4  *   *   *   *   *   *

5       *   *   *   *   *   *        5  *   *   *   *   *   *

6       *   *   *   *   *   *        6  *   *   *   *   *   *
```

>> Fred's Turn, please enter a Location:

>> F2

>> Congratulations Fred that was a HIT!

>> Fred sank John's Destroyer!

>> Here is the current Gameboard:

```
            John's Board                      Fred's Board

        A   B   C   D   E   F            A   B   C   D   E   F

1       *   *   *   *   *   X      1     *   *   *   *   *   *

2       *   *   *   *   *   X      2     *   *   *   *   *   *

3       *   *   *   *   *   *      3     *   *   *   *   *   *

4       *   *   *   *   *   *      4     *   *   *   *   *   *

5       *   *   *   *   *   *      5     *   *   *   *   *   *

6       *   *   *   *   *   *      6     *   *   *   *   *   *
```

>> Fred's Turn, please enter a Location:

>> F3

>> I am sorry Fred that was a MISS!

>> Here is the current Gameboard:

```
            John's Board                      Fred's Board

        A   B   C   D   E   F            A   B   C   D   E   F

1       *   *   *   *   *   X      1     *   *   *   *   *   *

2       *   *   *   *   *   X      2     *   *   *   *   *   *

3       *   *   *   *   *   *      3     *   *   *   *   *   *

4       *   *   *   *   *   *      4     *   *   *   *   *   *

5       *   *   *   *   *   *      5     *   *   *   *   *   *

6       *   *   *   *   *   *      6     *   *   *   *   *   *
```

>> John's Turn, please enter a Location:

Etc... until the game is finished...

References

Picture:

[1] https://cf.geekdo-images.com/images/pic288374.jpg

[2] http://petevsdanbattleship.com/game-rules/