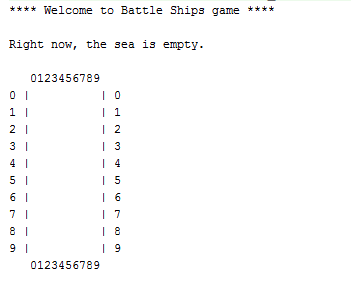
**Battle Ships Warmup**

I hope this will help you get more familiar with game concept – from developer side of view. You will be recreating the game of battleships, a player will place 5 of their ships on a 10 by 10 grid. The computer player will deploy five ships on the same grid. Once the game starts the player and computer take turns, trying to sink each other’s ships by guessing the coordinates. The game ends when either the player or computer has no ships left.

**Task 1 – Create the grid**

The ocean map is a 10 by 10 grid. The grid is represented by two-dimensional array. You will use this 2D array to save where the user and computer decide to place their ships, as well as when someone tries to attack a location and misses. At the start of the game the array will be empty.

Proposed look:



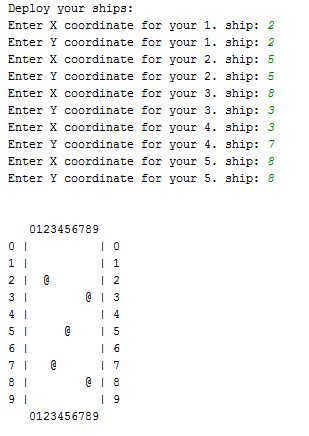
**Task 2 – Deploy player’s ships**

Once you have your ocean map, you'll need to ask the user where they would like to place their ships. The player should deploy 5 ships. They will tell you coordinates and you need to remember where the user wants to place those ships.

As the user is telling you where to place their ships you need to check if that is an appropriate location:

* you can NOT place two or more ships on the same location
* you can’t place ships outside the 10 by 10 grid If the player is trying to put the ship somewhere it can't be, reprompt them until they place the ship appropriately.

You should store the player's ships within the OceanMap as '1' but they should show up in the output as a '@' or something else of your choosing.



**Task 3 – Deploy computer’s ships**

The computer will deploy 5 ships by randomly picking X and Y coordinates.

* you cannot place the ship on a location that is already taken by another ship (player’s or computer’s)
* you can’t place ships outside the 10 by 10 grid If the computer tries to place the ship somewhere it can't be, regenerate random coordinates until all ships are placed appropriately.

You should store the computer's ships within the OceanMap as '2' and they should be invisible on the ocean map.

**Task 4 – Battle**

Once the player and computer have placed their ships it's time to start the battle. During the battle, the player and computer will take turns guessing X and Y coordinates of the opponent’s ships. Every coordinate guessed should be marked so they players know not to guess there again.

When the player enters X and Y coordinates you should check if those coodinates are valid within the Ocean Map and haven't been guessed by the user yet, keep reprompting until the user enters a valid guess. Once the guess is valid your program needs to evaluate the result of the move. There are three possible options:

* Player correctly guessed coordinates of computer’s ship (computer loses ship)
* Player entered coordinates of his/her own ship (player loses ship)
* Player missed. No ship on the entered coordinates In all of these cases you should mark the coordinates on the ocean map, so the player knows how to guess better next time.

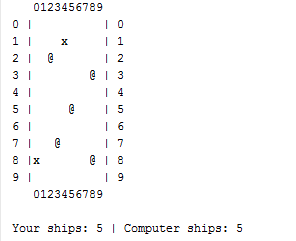
After the player guesses a coordinate it's the computer's turn to guess. This should be two randomly generated coordinates, you need to keep generating random numbers until you get a valid guess. When the computer produces a valid guess there are three possible outcomes:

* Computer guessed coordinates of the player’s ship (player loses ship)
* Computer guessed coordinates of its own – do not allow that, generate another coordinates until valid.
* Computer missed. No ship on guessed coordinates In all of these cases you should mark the coordinates on the ocean map. Make sure you mark the misses so the computer doesn't guess the same locations twice.

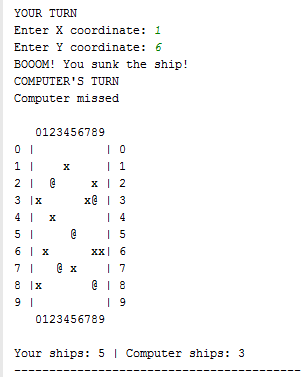
The battle will continue to run until one of the players is out of ships.

**Task 5 - Game Over**

When the user and computer are done guessing, display the current state of the ocean map and score.



Notice that the coordinates of player’s and computer’s shots are marked with “x” on the screen. You can mark known coordinates with number 3 on your ocean map. This is how the screen will look after couple turns.



The game is over when one player or computer has no ship left.

\*\*\*I will test your solution and I will do it thoroughly 😊

(It should be error prone and predict all stupid behavior that user can do)