

The battleship strategy.

## Introduction:

Battleship is a strategy guessing game, which player guess the position of an opponent's ship. The two players will take turns, with each player guessing the position of their opponent. After one placed a guess, the result would be known, either a hit or a miss. If the result is a hit, the player would get another chance to hit.

The game of battleship take place on a 10\*10 grid, at the start of the game, each player have a chance to place their ships. Each player should have:

1. Carrier(5 space)
2. Battleship(4 space)
3. Cruiser(3 space)
4. Submarine(3 space)
5. Destroyer(2 space)

You can only place the ship horizontally or vertically. The game ends when any player's all 5 ship sinks.

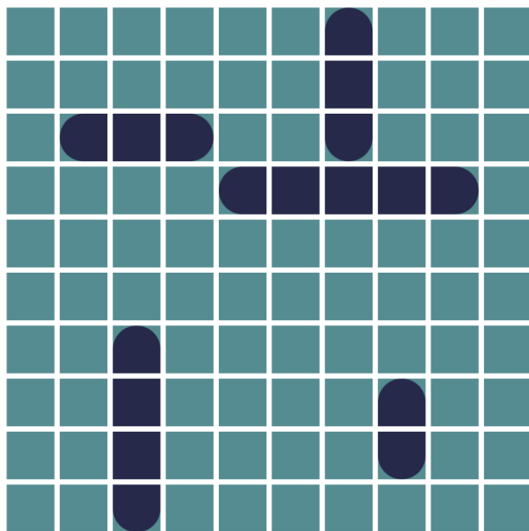


Figure 1: An example of arrangement of ships

Although seems random, the game of battleship has strategy and patterns. I collected 1000 rounds of battleship games and concluded that on average, there are 49.4 moves per game, with the most move at 97 and the least at 17 moves..

I concluded the least frequent hit location in the first 39 action, shown in this picture:

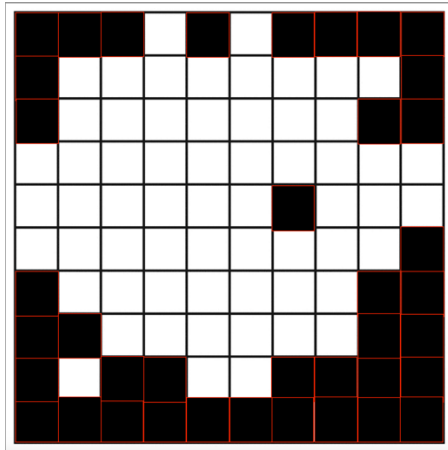


Figure 2: Least hit position

Theoretically, with ships placed at these location, the probability of getting hit would be the lowest.

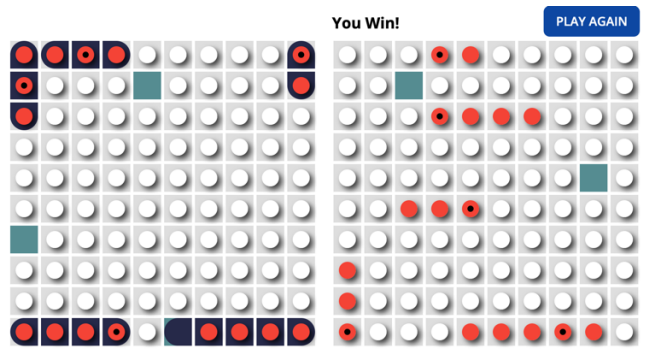


Figure 3: Example with winning with defensive strategy

To test the strategy, I played 52 games and win 33 games. The p-value as calculated:

$$\sum_{n=33}^{52} \binom{52}{n} \cdot (0.5)^{52} = 0.0351971105$$

Is lower than 0.05.

## Attacking

For the attacking strategy, I coded a code that calculates the probability of a targeted ship length, it sums up all the possible location a ship can be at and returns the location with highest probability. Sometimes, there will be multiple locations with the same probability, in this case, the algorithm will return the first position found.

```
[2, 3, 4, 4, 6, 6, 5, 4, 3, 2]
[3, 4, 5, 4, 7, 7, 6, 5, 4, 3]
[4, 5, 6, 4, 8, 8, 7, 6, 5, 4]
[4, 4, 4, 0, 5, 6, 6, 6, 6, 5]
[6, 7, 8, 5, 10, 10, 9, 8, 7, 6]
[6, 7, 8, 6, 10, 10, 9, 8, 7, 6]
[5, 6, 7, 6, 9, 9, 8, 7, 6, 5]
[4, 5, 6, 6, 8, 8, 7, 6, 5, 4]
[3, 4, 5, 6, 7, 7, 6, 5, 4, 3]
[2, 3, 4, 5, 6, 6, 5, 4, 3, 2]
x:5
y:5
```

Figure 4: 4 positions with the same probability, the algorithm returns (5,5), which is the first found

The Algorithm is fairly simple, it calculates all possible arrangement of the opponent's ship, and find the location that contains the most "ships".

## Getting help:

For the completion of this strategy, I asked a computer engineer on the process of building a website, since my code was originally written in python, I've encountered some problems putting it on websites.

Battleship assistance: <https://battleship.onrender.com/>