

Proyecto battleship

Profesor: Antonio Saad

Integrantes: Diego Valdez #25266699

Ernesto Vera #21248879

Miguel Obando#24964795

Funciones

creador(matriz[10][10]): El creador gestión la creación de los tableros, para esto tiene la ayuda de las funciones:

- **comprobador(int fila, int columna, int sentido, int fragata, char matriz [10][10]):** Comprueba que no estas saliendo de los límites establecidos y que no se está sobreponiendo un barco sobre otro. Retorna "0" si está bien, "-1" si un barco sobrepone a otro y "-2" si te sales de la estructura.
- **colocar (int fila, int columna, int sentido, int fragata, char matriz [10][10]):** Realiza la misma comprobación que ("comprobador") pero ya conociendo el resultado y pasa a colocar el barco.

enviarJugada(acertados, jugada): Muestra la cantidad de tiros acercados que has hecho simplemente imprimiendo la matriz de caracteres y almacena en una matriz tu jugada, luego de esto la función player1 o player2 se encargan de ponerla en la estructura, poner la letra correspondiente a Golpe ("msg=G") y retorna la estructura.

mensaje recibirJugada(mensaje estructura, char tablero[10][10]): Saca la fila y la columna de la estructura y comprueba con el tablero donde están ubicados tus barcos si le diste a uno, le agrega a la estructura una 'O' de que acertaste más el símbolo del barco al que acertaste sino le agrega un 'X' de que fallaste el objetivo y retorna la estructura.

int verificarGanador(char jugadas[10][10]): Cuenta la cantidad de caracteres que tienes distintos de la matriz acertados y si llega a 17 te notifica que ganaste. Retorna -1.

serializar(mensaje estructura , char serializado[4]): Inserta la estructura en un buffer para que sea apta para pasar por el socket. Retorna una matriz de caracteres.

deserializar(mensaje estructura, char serializado[4]): realiza el proceso inverso de serializar y convierte los caracteres a enteros. Retorna una estructura.

escribirServidor(char serializado[4]): Manda un mensaje al servidor.

leerServidor(char serializado[4]): Recibe una mensaje del servidor.

cerrarCliente(): Cierra la conexión del socket.

Estructura básica:

```
struct Mensaje {  
char msg;  
int fila;  
int columna;  
char símbolo;  
};
```

La estructura se compone:

Msg: es el mensaje/resultado o acción de cada uno de los jugadores

Los mensajes pueden ser:

- W: Ganaste
- I: Inicia tu tablero
- L: Listo
- G: Golpe
- O: Acertaste
- F: Fallaste
- T: Tu turno

Fila: es la fila donde se va a disparar

Columna: es la columna donde se va a disparar

Símbolo: En caso de que hayas acertado a algo el tomara el símbolo de ese barco.

Como trabaja el programa:

1. Yo inicio la estructura en {'I',0,0,'I'};
2. Se paso esa estructura a player1, al ser 'I' de inicia tu tablero, llamo a creador, una vez allí retorno la estructura con 'L' de anunciando que su tablero está listo y lo envía por el socket una vez serializado.
3. El player2 la recibe y comprueba 'L' entonces como él está listo, el player2 crea su tablero (invocando la función "creador") y luego de eso a la estructura le pone el mensaje de 'L' y retorna la estructura por el socket una vez serializado.

4. Player1 recibe la estructura y cambia en la estructura el "msg" a una 'T' indicando que es el turno del player2
5. Player2 al recibir la 'T' entra en la función 'EnviarJugada', le modifica en la estructura, fila, columna y el mensaje (esta vez a 'G' indicando que estas mandando un golpe) y retorna la estructura por el socket una vez serializado.
6. Player1 la recibe y comprueba en su matriz "tablero" mediante la función "RecibirJugada" si la fila y columna recibida coinciden con algún barco, de ser así, agrega un 'O' si acertó y un 'X' si no acertó. Retorna la estructura por el socket una vez serializado.
7. Player2 recibe el mensaje verificando la estructura, si es un 'O' agrega el símbolo en la estructura a su matriz 'acertados' y si no agrega una 'X' (ambos con la posición de la fila y columna que están en la estructura). Si es un 'O' verifica si ganó contando la cantidad de golpes que ha acertado mediante la función 'verificar ganador'. Una vez esto la función cambia "msg" a 'T' y se repite el proceso del 5 hasta el 7 pero con player1.

Para mayor referencia del código implementado en este proyecto dirigirse:

[Github.com/diegovaldesjr/Battleship-por-Sockets](https://github.com/diegovaldesjr/Battleship-por-Sockets)