

Universidad Rafael Landívar Facultad de Ingeniería.  
Ingeniería Civil  
Laboratorio de programación – Sección:18  
Catedrático: Ing. Hugo Adolfo Tzul Pérez



## **PROYECTO 2 (PYTHON)**

### **“BATALLA NAVAL”**

Stefanny Gisselle Carpio Hidalgo  
Carné: 1361923

Guatemala 31 de octubre de 2023

# CLASES

## CLASES UTILIZADAS

**Barco:** Esta clase representa un barco en el juego y tiene un método para ocupar las casillas del tablero con un barco.

**Jugador:** Esta clase representa a un jugador y contiene métodos para colocar barcos en su tablero, disparar al tablero del oponente y gestionar su nombre.

Estas clases son fundamentales para organizar y ejecutar el juego de Batalla Naval. Cada una de ellas cumple un propósito específico en el juego.

## ATRIBUTOS Y TIPOS DE DATOS

### **Clase Tablero:**

#### **Atributos:**

dimensión (int): Almacena la dimensión del tablero, que en este caso es 10x10 (valor constante).

matriz (lista de listas de str): Representa el tablero de juego, donde cada elemento es una cadena que puede ser "O" (casilla vacía), "B" (parte de un barco) o "X" (disparo repetido o fallo).

**Descripción:** La clase Tablero crea y almacena el tablero del juego, permitiendo su visualización y seguimiento de las casillas ocupadas.

### **Clase Barco:**

**Atributos:** fila (int): Almacena la fila de inicio del barco en el tablero. columna

(int): Almacena la columna de inicio del barco en el tablero. orientación (str):

Almacena la orientación del barco ("horizontal" o "vertical").

tipo (str): Almacena el tipo del barco ("pequeño" o "grande").

**Descripción:** La clase Barco representa un barco en el juego, almacenando su ubicación en el tablero y su tipo.

### **Clase Jugador:**

**Atributos:** Nombre (str): Almacena el nombre del jugador.

tablero (instancia de la clase Tablero): Almacena el tablero del jugador. barcos (lista de instancias de la clase Barco): Almacena los barcos del jugador.

**Descripción:** La clase Jugador representa a un jugador del juego de Batalla Naval. Almacena su nombre, su tablero y los barcos que coloca en el tablero.

Estos son los atributos y sus respectivos tipos de datos utilizados en las clases del juego. Los atributos son esenciales para mantener un seguimiento de la información

### RETORNOS Y ATRIBUTOS

Clase Tablero:

**Método** \_\_init\_\_(self)

**Descripción:** Constructor de la clase Tablero que inicializa el tablero con casillas vacías.

**Retorna:** No retorna ningún valor explícito (porque es un constructor).

**Método** mostrar(self)

**Descripción:** Muestra el tablero en la consola.

**Retorna:** No retorna ningún valor.

### CLASE BARCO:

**Método** \_\_init\_\_(self, fila, columna, orientación, tipo)

**Descripción:** Constructor de la clase Barco que inicializa los atributos del barco.

**Retorna:** No retorna ningún valor explícito (porque es un constructor).

**Método** ocupar\_casillas(self, tablero)

**Descripción:** Ocupa las casillas del tablero con el barco de acuerdo a su posición y orientación.

**Retorna:** No retorna ningún valor.

### Clase Jugador:

**Método** \_\_init\_\_(self, nombre)

**Descripción:** Constructor de la clase Jugador que inicializa los atributos del jugador.

**Retorna:** No retorna ningún valor explícito (porque es un constructor).

**Método** colocar\_barcos(self)

**Descripción:** Permite al jugador colocar sus barcos en el tablero.

**Retorna:** No retorna ningún valor.

**Método** disparar(self, oponente)

**Descripción:** Permite al jugador realizar un disparo en el tablero del oponente.

**Retorna:** Una cadena que indica el resultado del disparo ("Hundido", "Repetido" o "Fallo"). necesaria para jugar y controlar el juego.

### CONDICIONES Y RESTRICCIONES

Para que el programa de Batalla Naval funcione correctamente y sea una experiencia de juego justa, se debe tener en cuenta varias condiciones y restricciones:

Ubicación válida de barcos: Los barcos deben colocarse dentro de los límites del tablero (filas de A a J y columnas de 1 a 10).

Cantidad y tipo de barcos: Debe haber 3 barcos pequeños y 2 barcos grandes para cada jugador.

Alternancia de turnos: Los jugadores deben tomar turnos para disparar

Reglas de disparo: Cada jugador debe disparar dentro de las coordenadas válidas del tablero del oponente.

## ALGORITMO DEL PROGRAMA

Inicialización del juego:

Se crean dos objetos Jugador para representar a los jugadores 1 y 2.

Cada jugador tiene su propio tablero, nombre y lista de barcos.

Colocación de barcos:

Cada jugador, por turno, coloca sus barcos en su propio tablero:

El jugador selecciona una coordenada de inicio y una orientación para cada barco.

Alternancia de turnos: Los jugadores toman turnos para disparar. El jugador 1 comienza el juego.

En cada turno, el jugador actual elige una casilla en el tablero del oponente para disparar.

Disparos y resultados: El programa verifica el resultado del disparo:

Si la casilla contiene un barco, se marca como "Hundido".

Si ya se disparó en esa casilla, se marca como "Repetido".

Si no hay un barco en la casilla, se marca como "Fallo".

Se actualiza el tablero del oponente con el resultado del disparo.

Determinación del ganador: Después de cada disparo, el programa verifica si todos los barcos de un jugador han sido hundidos.

Repetición de turnos

Fin del juego