

**UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA**  
**FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS**  
**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



**ASIGNATURA: FUNDAMENTOS DE PROGRAMACIÓN 2**

**DOCENTE: REVILLA ARROYO CHRISTIAN ALAIN**

***“Laboratorio 08: Clase Soldado ”***

**Estudiante:**

**Torres Chávez Jhair Alejandro**

**Arequipa – Perú**

**2023**

## **Laboratorio 90;**

### **Enunciado**

Tendrá 2 Ejércitos. Usar la estructura de datos que considere más adecuada. Inicializar el tablero con n soldados aleatorios entre 1 y 10 para cada Ejército. Cada soldado tendrá un nombre autogenerado: Soldado0X1, Soldado1X1, etc., un valor de puntos de vida autogenerado aleatoriamente [1..5], la fila y columna también autogenerados aleatoriamente (no puede haber 2 soldados en el mismo cuadrado). Nivel de ataque y de defensa son aleatorios [1..5]. Se debe mostrar el tablero con todos los soldados creados (usar caracteres como | \_ y otros) y distinguir los de un ejército de los del otro ejército. Además de los datos del Soldado con mayor vida de cada ejército, el promedio de puntos de vida de todos los soldados creados por ejército, los datos de todos los soldados por ejército en el orden que fueron creados y un ranking de poder de todos los soldados creados por ejército (del que tiene más nivel de vida al que tiene menos) usando 2 diferentes algoritmos de ordenamiento. Finalmente, que muestre qué ejército ganará la batalla (indicar la métrica usada para decidir al ganador de la batalla). Hacerlo un programa iterativo.

- Crear el diagrama de clases UML completo

**Link GitHub (CÓDIGO):**

## **Método de resolución**

El laboratorio se plantó desde una perspectiva más dinámica por lo que se crearon 4 clases: Soldado, Ejercito, Tablero y Videojuego. Incluso puede usarse un tablero de NxN con X ejércitos a la vez, considerando el número de espacios en el tablero.

## **Clases**

### **Soldado:**

Esta clase contiene todo lo necesario para cada unidad dentro del videojuego, sus atributos como la vida inicial y actual, nivel de ataque, nivel de defensa, velocidad, etc. Algunos de estos atributos son necesarios acceder mediante métodos que se encuentran en la sección de getters y setters. Se crearon 3 constructores sobrecargados, que dependerá del uso que se le dé dentro del videojuego. Finalmente se encuentran los métodos pasivos y activos las acciones que podrá realizar dentro del videojuego.

### **Ejercito:**

Esta clase es una de las que se usa dentro del main, ya que en sí es un ArrayList para almacenar los diferentes objetos de clase Soldado. Además que en tiempo de ejecución se comunicará con la clase Tablero (en el constructor) para posicionar cada objeto dentro de la matriz bidimensional (tablero).

La clase cuenta con todos los métodos estadísticos de ordenamiento y selección para poder rankear, promediar y organizar los objetos Soldado.

**Tablero:**

Esta clase se basa en el array bidimensional que contiene a objetos de la clase Ejercito, que a su vez contienen a los objetos clase Soldado. Esta clase contiene los métodos de validación para posicionar los objetos generados aleatoriamente dentro de el objeto Ejercito y así poder verificar que es un espacio vacío y poder asignarle sus estadísticas libremente. Además contiene los métodos de graficación de acuerdo al equipo que pertenezcan, cada equipo tiene su propio identificador dentro del gráfico.

**Videojuego:**

Esta clase actúa como la principal donde todos las demás interactúan como un todo, simplemente hace uso de los métodos de las demás para poder ejecutar el videojuego, sin complicarnos en cómo fue realizado o su forma de ejecución u otros métodos.

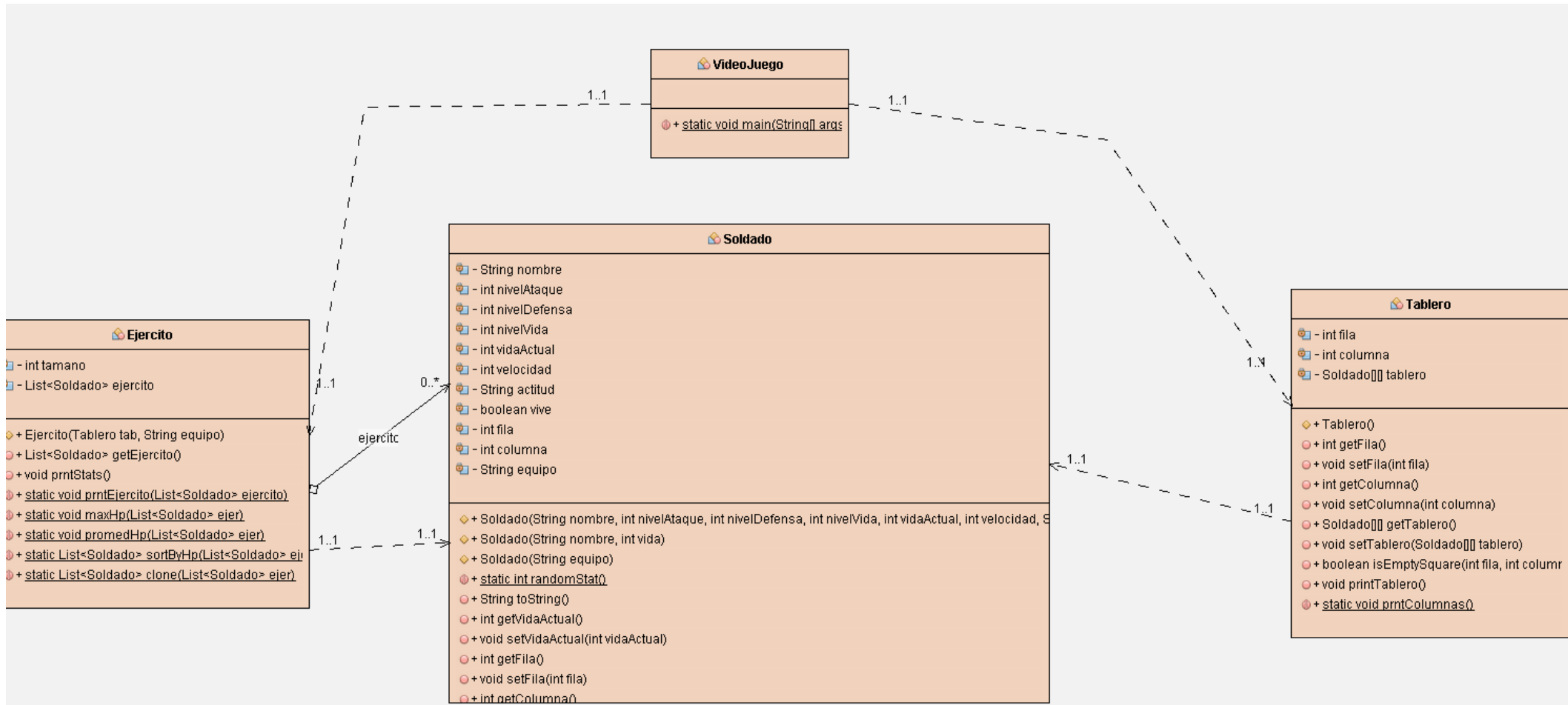
Ejército A:

@@@@@  
@@@@@  
@@@@@

Ejército B:

=====  
=====  
=====

## Diagrama UML



### Resultado final:

**Los métodos gráficos de la clase Tablero se ejecutan mostrando las unidades creadas, y las diferencia mediante símbolos( @ y =)-**

[illegible]

## Estadísticas:

EJERCITO A  
SOLDADO MAX HP  
Nombre: Soldado5x0  
Vida: 3  
Fila : 6 Columa: 1

PROMEDIO DE HP  
1.75

Ejercito original  
Posicion 1  
Nombre: Soldado1x6  
Vida: 1  
Fila : 2 Columa: 7

Posicion 2  
Nombre: Soldado5x1  
Vida: 1  
Fila : 6 Columa: 2

Posicion 3  
Nombre: Soldado5x0  
Vida: 3  
Fila : 6 Columa: 1

Posicion 4  
Nombre: Soldado2x6  
Vida: 2  
Fila : 3 Columa: 7

RANKING PODER  
Posicion 1  
Nombre: Soldado5x0  
Vida: 3  
Fila : 6 Columa: 1

Posicion 2  
Nombre: Soldado2x6  
Vida: 2  
Fila : 3 Columa: 7

Posicion 3  
Nombre: Soldado1x6  
Vida: 1  
Fila : 2 Columa: 7

Posicion 4  
Nombre: Soldado5x1  
Vida: 1  
Fila : 6 Columa: 2

EJERCITO B  
SOLDADO MAX HP  
Nombre: Soldado0x6  
Vida: 5  
Fila : 1 Columa: 7

PROMEDIO DE HP  
3.2

Ejercito original  
Posicion 1  
Nombre: Soldado4x4  
Vida: 3  
Fila : 5 Columa: 5

Posicion 2  
Nombre: Soldado4x0  
Vida: 2  
Fila : 5 Columa: 1

Posicion 3  
Nombre: Soldado0x6  
Vida: 5  
Fila : 1 Columa: 7

Posicion 4  
Nombre: Soldado5x6  
Vida: 1  
Fila : 6 Columa: 7

RANKING PODER  
Posicion 1  
Nombre: Soldado0x6  
Vida: 5  
Fila : 1 Columa: 7

Posicion 2  
Nombre: Soldado5x3  
Vida: 5  
Fila : 6 Columa: 4

Posicion 3  
Nombre: Soldado4x4  
Vida: 3  
Fila : 5 Columa: 5

Posicion 4  
Nombre: Soldado4x0  
Vida: 2  
Fila : 5 Columa: 1

Posicion 5  
Nombre: Soldado5x6  
Vida: 1  
Fila : 6 Columa: 7