

# Informe de Laboratorio 06

## Tema: Laboratorio 06

Nota

Estudiante	Escuela	Asignatura
Victor Mamani Anahua vmamanian@unsa.edu.pe	Escuela Profesional de Ingeniería de Sistemas	Fundamentos de la Programación II Semestre: II Código: 20230489

Laboratorio	Tema	Duración
06	Laboratorio 06	04 horas

Semestre académico	Fecha de inicio	Fecha de entrega
2023 - B	Del 09 Octubre 2023	Al 16 Octubre 2023

### 1. Tarea

- Cree un Proyecto llamado Laboratorio6
- Usted deberá crear las dos clases Soldado.java y VideoJuego3.java. Puede reutilizar lo desarrollado en Laboratorios anteriores.
- Del Soldado nos importa el nombre, puntos de vida, fila y columna (posición en el tablero).
- El juego se desarrollará en el mismo tablero de los laboratorios anteriores. Pero ahora el tablero debe ser un ArrayList bidimensional.
- Tendrá 2 Ejércitos. 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).
- 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).

## 2. Equipos, materiales y temas utilizados

- Sistema Operativo Ubuntu GNU Linux 23 lunar 64 bits Kernell 6.2.v
- Visual Studio Code.
- VIM 9.0.
- OpenJDK 64-Bits 19.0.7.
- Git 2.39.2.
- Cuenta en GitHub con el correo institucional.
- Programación Orientada a Objetos.
- Actividades del Laboratorio 06.

## 3. URL de Repositorio Github

- URL del Repositorio GitHub para clonar o recuperar.
- <https://github.com/VictorMA18/fp2-23b.git>
- URL para el laboratorio 01 en el Repositorio GitHub.
- <https://github.com/VictorMA18/fp2-23b/tree/main/Fase02/Lab06>

## 4. Actividades del Laboratorio 06

### 4.1. Ejercicio Soldado

- En el primer commit bueno reutilizamos el archivo que seria nuestra clase Soldado el cual la utilizaremos para poder avanzar.
- El codigo y el commit seria el siguiente:

Listing 1: Commit

```
$ git commit -m " Anadimos la clase para poder avanzar con el siguiente ejercicio  
reutilizamos el archivo del laboratorio 5 ya que serian los mismos atributos solo  
que esta vez lo usaremos con un ArrayList bidimensional"
```

Listing 2: Las lineas de codigos del metodo creado:

```
public class Soldado { //CREAMOS LA CLASE SOLDODADO PARA PODER USAR UN ARREGLO  
    BIDIMENSIONAL DONDE NECESITAMOS LA VIDA , EL NOMBRE DEL SOLDADO Y TAMBIEN SU  
    POSICION COMO LA FILA Y LA COLUMNA  
  
    private String name;  
    private int heatlh;  
    private int row;  
    private String column;  
  
    //Constructor  
    public Soldado(String name, int health, int row, String column){
```

```
this.name = name;
this.health = health;
this.row = row;
this.column = column;
}

// Metodos mutadores
public void setName(String n){
    name = n;
}
public void setHealth(int p){
    health = p;
}
public void setRow(int b){
    row = b;
}
public void setColumn(String c){
    column = c;
}

// Metodos accesoros
public String getName(){
    return name;
}
public int getHealth(){
    return health;
}

public int getRow(){
    return row;
}
public String getColumn(){
    return column;
}

// Completar con otros metodos necesarios
public String toString(){ //CREAMOS ESTE METODO PARA IMPRIMIR LOS DATOS DEL OBJETO
    String join = "\nNombre: " + getName() + "\nVida: " + getHealth() + "\nFila: " +
        getRow() + "\nColumna: " + getColumn(); //Agregamos un espaciador para poder
        separar
    return join;
}
}
```

## 4.2. Ejercicio Soldado

- En el primer commit bueno reutilizamos el archivo que seria nuestra clase Soldado el cual la utilizaremos para poder avanzar.
- El codigo y el commit seria el siguiente:

Listing 3: Commit

```
$ git commit -m "Anadimos la clase para poder avanzar con el siguiente ejercicio
reutilizamos el archivo del laboratorio 5 ya que serian los mismos atributos solo
que esta vez lo usaremos con un ArrayList bidimensional"
```

Listing 4: Las líneas de códigos del método creado:

```
public class Soldado { //CREAMOS LA CLASE SOLDADO PARA PODER USAR UN ARREGLO
    BIDIMENSIONAL DONDE NECESITAMOS LA VIDA , EL NOMBRE DEL SOLDADO Y TAMBIEN SU
    POSICION COMO LA FILA Y LA COLUMNA

    private String name;
    private int health;
    private int row;
    private String column;

    //Constructor
    public Soldado(String name, int health, int row, String column){
        this.name = name;
        this.health = health;
        this.row = row;
        this.column = column;
    }

    // Metodos mutadores
    public void setName(String n){
        name = n;
    }
    public void setHealth(int p){
        health = p;
    }
    public void setRow(int b){
        row = b;
    }
    public void setColumn(String c){
        column = c;
    }

    // Metodos accesoros
    public String getName(){
        return name;
    }
    public int getHealth(){
        return health;
    }

    public int getRow(){
        return row;
    }
    public String getColumn(){
        return column;
    }

    // Completar con otros metodos necesarios
    public String toString(){ //CREAMOS ESTE METODO PARA IMPRIMIR LOS DATOS DEL OBJETO
        String join = "\nNombre: " + getName() + "\nVida: " + getHealth() + "\nFila: " +
            getRow() + "\nColumna: " + getColumn(); //Agregamos un espaciador para poder
            separar
        return join;
    }
}
```

```
}  
}
```

### 4.3. Estructura de laboratorio 06

- El contenido que se entrega en este laboratorio06 es el siguiente:

/Lab06      "MOSTRAR RAMA"

## 5. Rúbricas

### 5.1. Entregable Informe

Tabla 1: Tipo de Informe

<b>Informe</b>	
<b>Latex</b>	El informe está en formato PDF desde Latex, con un formato limpio (buena presentación) y facil de leer.

## 5.2. Rúbrica para el contenido del Informe y demostración

- El alumno debe marcar o dejar en blanco en celdas de la columna **Checklist** si cumple con el ítem correspondiente.
- Si un alumno supera la fecha de entrega, su calificación será sobre la nota mínima aprobada, siempre y cuando cumpla con todos los ítems.
- El alumno debe autocalificarse en la columna **Estudiante** de acuerdo a la siguiente tabla:

Tabla 2: Niveles de desempeño

	Nivel			
Puntos	Insatisfactorio 25 %	En Proceso 50 %	Satisfactorio 75 %	Sobresaliente 100 %
2.0	0.5	1.0	1.5	2.0
4.0	1.0	2.0	3.0	4.0

Tabla 3: Rúbrica para contenido del Informe y demostración

	Contenido y demostración	Puntos	Checklist	Estudiante	Profesor
<b>1. GitHub</b>	Hay enlace URL activo del directorio para el laboratorio hacia su repositorio GitHub con código fuente terminado y fácil de revisar.	2	X	2	
<b>2. Commits</b>	Hay capturas de pantalla de los commits más importantes con sus explicaciones detalladas. (El profesor puede preguntar para refrendar calificación).	4	X	1	
<b>3. Código fuente</b>	Hay porciones de código fuente importantes con numeración y explicaciones detalladas de sus funciones.	2	X	1	
<b>4. Ejecución</b>	Se incluyen ejecuciones/pruebas del código fuente explicadas gradualmente.	2	X	2	
<b>5. Pregunta</b>	Se responde con completitud a la pregunta formulada en la tarea. (El profesor puede preguntar para refrendar calificación).	2	X	2	
<b>6. Fechas</b>	Las fechas de modificación del código fuente están dentro de los plazos de fecha de entrega establecidos.	2	X	2	
<b>7. Ortografía</b>	El documento no muestra errores ortográficos.	2	X	2	
<b>8. Madurez</b>	El Informe muestra de manera general una evolución de la madurez del código fuente, explicaciones puntuales pero precisas y un acabado impecable. (El profesor puede preguntar para refrendar calificación).	4	X	2	
<b>Total</b>		20		14	

## 6. Referencias

- <https://drive.google.com/file/d/18wvjXuguiRaIZ3Z0dElzC-LM9hrabhue/view>