

# LABORATORIO 09

## HASHMAP - VIDEOJUEGO

**APELLIDOS Y NOMBRES:** CHAMBILLA PERCA, Valentina Milagros

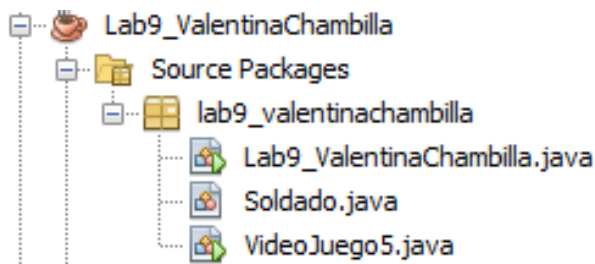
**CUI:** 20213032

**CURSO:** Fundamentos de la Programación 2

El Laboratorio se realizó en el entorno Apache NetBeans IDE 12.4



La estructura es la siguiente:



### ARCHIVO VideoJuego5.java

```
// Laboratorio Nro 9 - Ejercicio 1
```

```
// Autor: Valentina Milagros Chambilla Perca
```

```
//Considero que tuve muchas dificultades al trabajar con las clases y el Hashmap
```

```
//Ya que la forma en la que realicé antes el VideoJuego fue siempre de forma  
bidimensional
```

```
//Y consideré que trabajar con pares en la primera parte sería inadecuado ya que aún  
no vimos
```

```
//El objeto Pair, como para poder utilizarlo como clave (y así poder colocar dos  
valores)
```

```
package lab9_valentinachambilla;
```

```
import java.util.*;
```

```
public class VideoJuego5 {  
    public static void main(String[] args) {  
        int cantidad1, cantidad2;  
        HashMap<String , Soldado> ListaSol = new HashMap<>();  
        //Inicia el juego
```

```

System.out.println("*****JUEGO V.
3*****");

cantidad1 = Aleatorio(10, 1);
cantidad2 = Aleatorio(10, 1);
System.out.println("El primer equipo tendrá: " + cantidad1);
System.out.println("El segundo equipo tendrá: " + cantidad2);
/*
System.out.println(cantidad + " SOLDADOS.");
if (cantidad>5)
    System.out.println("QUE SUERTE OuO");
else
    System.out.println("Bueno..., tu suerte quiza estará en otro lado UwU");
*/
System.out.println("Se estan generando los soldados ~~");
//Generación de Soldados
GenerarSoldados(ListaSol, cantidad1, 1);
GenerarSoldados(ListaSol, cantidad2, 2);
//Mostrar Matriz Bidimensional
mostrarMatrizDeSoldados(ListaSol);
//Mostrar Soldado con más vida
System.out.println("\nMOSTRAR SOLDADO CON MAYOR CANTIDAD DE VIDA");
mostrarSoldadoMayorVida(ListaSol, 1);
mostrarSoldadoMayorVida(ListaSol, 2);
//Promedio de los puntos de Vida
System.out.println("\nMOSTRAR PROMEDIO Y NIVEL DE VIDA");
double nivVid1 = mostrarPromedioYNivelVida(ListaSol, cantidad1, 1);
double nivVid2 = mostrarPromedioYNivelVida(ListaSol, cantidad2, 2);

//mostrar en Orden de creación
System.out.println("\nSOLDADOS POR ORDEN DE CREACIÓN");
mostrarSoldadosOrd(ListaSol, 1, cantidad1);
mostrarSoldadosOrd(ListaSol, 2, cantidad2);
System.out.println("\nORDENAMIENTO POR BURBUJA DEL PRIMER GRUPO");
HashMap<Integer , Soldado> ListaSol1 = new HashMap<>();
for (int i=0; i<cantidad1;i++){
    for (String key: ListaSol.keySet()){

```

```

        if (ListaSol.get(key).getEqui() == 1 &&
ListaSol.get(key).getNom().equalsIgnoreCase("Soldado"+1+"X"+i)){
            ListaSol1.put(i,ListaSol.get(key));
            break;
        }
    }
}
System.out.println("\nEQUIPO " + 1);
ordenarPorPuntosBurbuja(ListaSol1);
mostrarSoldados(ListaSol1);

```

```

HashMap<Integer , Soldado> ListaSol2 = new HashMap<>();
for (int i=0; i<cantidad2;i++){
    for (String key: ListaSol.keySet()){
        if (ListaSol.get(key).getEqui() == 2 &&
ListaSol.get(key).getNom().equalsIgnoreCase("Soldado"+2+"X"+i)){
            ListaSol2.put(i,ListaSol.get(key));
            break;
        }
    }
}
System.out.println("\nEQUIPO " + 2);
ordenarPorPuntosSeleccion(ListaSol2);
mostrarSoldados(ListaSol2);

```

//EL QUE TIENE MAYOR NIVEL DE VIDA ES EL EQUIPO GANADOR, SI SON IGUALES,  
ENTONCES SE DEFIINE POR LA CANTIDAD DE SOLDADOS

///SI SON IGUALES TAMBIÉN, ENTONCES SERÁ EMPATE.

```

if (nivVid1 >nivVid2){
    System.out.println("\nGANÓ EL EQUIPO 1");
}
else if (nivVid2>nivVid1){
    System.out.println("\nGANÓ EL EQUIPO 2");
}

```

```

else {
    if (ListaSol1.size()>ListaSol2.size())
        System.out.println("\nGANÓ EL EQUIPO 1");
    else if (ListaSol2.size()>ListaSol1.size())
        System.out.println("\nGANÓ EL EQUIPO 2");
    else
        System.out.println("\nEMPATE ENTRE LOS DOS EQUIPOS");
}
}

```

```

public static int Aleatorio(int Mayor, int Menor){
    Random aleatorio = new Random();

    int num = aleatorio.nextInt(Mayor-Menor+1)+Menor;
    return num;
}

```

```

public static void GenerarSoldados(HashMap<String, Soldado> ListaSol, int
cantidad, int equi){
    //El código identificador es (S*Número de equipo [1 o 2]*Número de soldado).
    Ejem: S10 (Soldado Equipo 1 Número 0)

    String code;
    String nombre;
    int vida, fila, columna;
    boolean ocupado;
    //Generar
    for (int i=0; i<cantidad; i++){
        nombre = "Soldado"+equi+"X"+i;
        code = "S"+equi+i;
        vida = Aleatorio(5,1);
        while(true){
            ocupado = false;
            fila = Aleatorio(9,0);
            columna = Aleatorio(9,0);
            for (String key: ListaSol.keySet()){

```

```

        if (ListaSol.get(key).getFila()==fila &&
ListaSol.get(key).getColumna()==columna)

            ocupado = true;

    }

    if (!ocupado){
        ListaSol.put(code, new Soldado(nombre, vida, fila, columna,
equi));

        break;

    }

}

}

}

}

public static void mostrarMatrizDeSoldados(HashMap<String, Soldado> ListaSol) {
    boolean ocupado=false;

    System.out.println("\033[0m MOSTRAR MATRIZ BIDIMENSIONAL DE SOLDADOS");

    System.out.printf("\n%6s%1s%11s%1s%11s%1s%11s%1s%11s%1s%11s%1s%11s%1s%11s%1s%11s%1s%11s\n", "   F/C ", "|", "      A      ", "|", "      B      ", "|", "      C      ", "|",
        "      D      ", "|", "      E      ", "|", "      F      ", "|", "      G      ", "|",
        "      H      ", "|", "      I      ", "|", "      J      ");

    for (int i=0; i<10;i++){
        //GenerarLineaSuperior
        for(int k=0; k<126; k++){
            System.out.print("\033[30m-");
        }

        System.out.printf("\n%1s%6s", "\033[30m", "   " + (i+1) + "   ");

        for (int j = 0; j < 10; j++){
            ocupado = false;

            System.out.printf("%1s", "|");

            for (String key: ListaSol.keySet()){
                if (ListaSol.get(key).getFila()==i &&
ListaSol.get(key).getColumna()==j )
                    ocupado = true;

            }

            if (ocupado){

```

```

        for (String key: ListaSol.keySet()){
            if (ListaSol.get(key).getFila()==i &&
ListaSol.get(key).getColumna()==j )
                if (ListaSol.get(key).getEqui()==1)

System.out.printf("%1s%11s", "\033[34m", ListaSol.get(key).getNom());

                else

System.out.printf("%1s%11s", "\033[36m", ListaSol.get(key).getNom());

        }
    }
    else {
        System.out.printf("%1s%11s", "\033[30m", "          ");
    }
}
System.out.println();
}
}

```

```

public static void mostrarSoldadoMayorVida(HashMap<String, Soldado> ListaSol, int
equi) {
    System.out.println("\nEQUIPO " + equi);
    int mayor = 0;
    for (String key: ListaSol.keySet()){
        if (ListaSol.get(key).getEqui() == equi &&
(ListaSol.get(key).getPuntoVida()>mayor))
            mayor = ListaSol.get(key).getPuntoVida();
    }
    //MOSTRAR LOS SOLDADOS CON LA MAYOR CANTIDAD DE VIDA
    System.out.printf( "%10s%8s%10s%15s%8s\n", "Nombre", "Fila", "Columna", "Puntos
Vida", "Equipo");
    for (String key: ListaSol.keySet()){
        if (ListaSol.get(key).getEqui() == equi &&
(ListaSol.get(key).getPuntoVida()==mayor))
            System.out.printf( "%10s%8s%10s%15d%8d\n", ListaSol.get(key).getNom(),
ListaSol.get(key).getFila(),
ListaSol.get(key).getColumna(), ListaSol.get(key).getPuntoVida(),
ListaSol.get(key).getEqui());
    }
}

```

```
}
```

```
    public static double mostrarPromedioYNivelVida(HashMap<String, Soldado> ListaSol,
int num, int equi) {
        System.out.println("\nEQUIPO " + equi);
        double suma=0;
        for (String key: ListaSol.keySet()){
            if (ListaSol.get(key).getEqui() == equi)
                suma = suma + ListaSol.get(key).getPuntoVida();
        }
        double promedio = suma/num;
        //MOSTRAR LOS SOLDADOS CON LA MAYOR CANTIDAD DE VIDA
        System.out.println( "EL NIVEL DE VIDA ES: " + suma);
        System.out.println( "EL PROMEDIO DE VIDA ES: " + promedio);
        return suma;
    }
}
```

```
    public static void mostrarSoldadosOrd(HashMap<String, Soldado> ListaSol, int equi,
int cant) {
        System.out.println("\nEQUIPO " + equi);
        System.out.printf( "%10s%8s%10s%15s%8s\n", "Nombre", "Fila", "Columna","Puntos
Vida", "Equipo");
        for (int i=0; i<cant;i++){
            for (String key: ListaSol.keySet()){
                if (ListaSol.get(key).getEqui() == equi &&
ListaSol.get(key).getNom().equalsIgnoreCase("Soldado"+equi+"X"+i)){
                    System.out.printf( "%10s%8s%10s%15d%8d\n",
ListaSol.get(key).getNom(), ListaSol.get(key).getFila(),
ListaSol.get(key).getColumna(),ListaSol.get(key).getPuntoVida(),
ListaSol.get(key).getEqui());
                    break;
                }
            }
        }
    }
}
```

```
public static void mostrarSoldados(HashMap<Integer, Soldado> ListaSol) {
```

```

        System.out.printf( "%10s%8s%10s%15s%8s\n", "Nombre", "Fila", "Columna","Puntos
Vida", "Equipo");

        for (int i=0; i<ListaSol.size();i++){

            System.out.printf( "%10s%8s%10s%15d%8d\n", ListaSol.get(i).getNom(),
ListaSol.get(i).getFila(),
ListaSol.get(i).getColumna(),ListaSol.get(i).getPuntoVida(),
ListaSol.get(i).getEqui());

        }

    }

    public static void ordenarPorPuntosBurbuja(HashMap<Integer, Soldado> ListaSol){

        System.out.println("ORDENAR POR PUNTOS: Burbuja");

        for (int i=1; i<ListaSol.size();i++)

            for(int j=0;j<ListaSol.size()-i;j++)

                if((ListaSol.get(j).getPuntoVida())<=(ListaSol.get(j+1).getPuntoVida()))

                    intercambiar(ListaSol, j, j+1);

    }

    private static void intercambiar(HashMap<Integer, Soldado> ListaSol, int i, int
j){

        Soldado temp = new Soldado("", 0, 0, 0, 0);

        temp = ListaSol.get(i);

        ListaSol.put(i,ListaSol.get(j));

        ListaSol.put(j, temp);

    }

    public static void ordenarPorPuntosSeleccion(HashMap<Integer, Soldado> ListaSol){

        System.out.println("ORDENAR POR PUNTOS: Selección");

        //Encontrar el menor

        Soldado mayor = new Soldado("", 0, 0, 0, 0);

        int indice=0;

        for (int i=0; i<ListaSol.size();i++){

            //Determinar el menor

            mayor = ListaSol.get(i);

            for (int j=i; j<ListaSol.size();j++)

                if (mayor.getPuntoVida() <= ListaSol.get(j).getPuntoVida()){

                    mayor = ListaSol.get(j);

                }

            }

        }

    }

```



```

        indice = j;
    }
    intercambiar(ListaSol, indice, i);
}
}
}

```

## ARCHIVO Soldado.java

// Laboratorio Nro 9 - Ejercicio 1

// Autor: Valentina Milagros Chambilla Perca

```
package lab9_valentinachambilla;
```

```

public class Soldado {
    private String Nom="";
    private int puntoVida=0;
    private int Fila=0;
    private int Columna=0;
    private int Equipo;
    //Equipo 1: Primer equipo
    //Equipo 2: Segundo Equipo

    public Soldado(String nombre, int Vida, int fila, int columna, int equi){
        Nom = nombre;
        puntoVida = Vida;
        Fila = fila;
        Columna = columna;
        Equipo = equi;
    }

    public String getNom(){
        return Nom;
    }

    public int getPuntoVida(){
        return puntoVida;
    }
}

```

```

}

public int getFila(){
    return Fila;
}

public int getColumna(){
    return Columna;
}

public int getEqui(){
    return Equipo;
}
}

```

## Ejemplos de Aplicación

### EJEMPLO 1

```

run:
*****JUEGO V. 3*****
El primer equipo tendrá: 10
El segundo equipo tendrá: 8
Se estan generando los soldados ~~
MOSTRAR MATRIZ BIDIMENSIONAL DE SOLDADOS

```

F/C	A	B	C	D	E	F	G	H	I	J
1							Soldado2X0	Soldado2X3		Soldado1X0
2						Soldado2X6		Soldado1X2		
3										
4								Soldado2X7		
5				Soldado2X2				Soldado2X4		
6	Soldado2X5									
7			Soldado1X4	Soldado1X3		Soldado1X7				
8			Soldado1X5							
9										
10	Soldado1X9	Soldado1X8				Soldado1X6		Soldado1X1		Soldado2X1

```

MOSTRAR SOLDADO CON MAYOR CANTIDAD DE VIDA
EQUIPO 1
Nombre      Fila  Columna  Puntos Vida  Equipo
Soldado1X6   9      5         5           1
EQUIPO 2
Nombre      Fila  Columna  Puntos Vida  Equipo

```

Soldado1X0	0	6	5	2
Soldado1X4	4	7	5	2

MOSTRAR PROMEDIO Y NIVEL DE VIDA

EQUIPO 1

EL NIVEL DE VIDA ES: 29.0

EL PROMEDIO DE VIDA ES: 2.9

EQUIPO 2

EL NIVEL DE VIDA ES: 24.0

EL PROMEDIO DE VIDA ES: 3.133

SOLDADOS POR ORDEN DE CREACIÓN

EQUIPO 1

Nombre	Fila	Columna	Puntos Vida	Equipo
Soldado1X0	0	9	2	1
Soldado1X1	9	7	1	1
Soldado1X2	1	7	4	1
Soldado1X3	6	3	4	1
Soldado1X4	6	2	4	1
Soldado1X5	7	2	1	1
Soldado1X6	9	5	5	1
Soldado1X7	6	5	1	1
Soldado1X8	9	1	4	1
Soldado1X9	9	0	3	1

EQUIPO 2

Nombre	Fila	Columna	Puntos Vida	Equipo
Soldado2X0	0	6	5	2
Soldado2X1	9	9	1	2
Soldado2X2	4	3	4	2
Soldado2X3	0	7	2	2
Soldado2X4	4	7	5	2
Soldado2X5	5	0	1	2
Soldado2X6	1	5	3	2
Soldado2X7	2	7	4	2

ORDENAMIENTO POR BURBUJA DEL PRIMER GRUPO

EQUIPO 1

ORDENAR POR PUNTOS: Burbuja

Nombre	Fila	Columna	Puntos Vida	Equipo
Soldado1X6	9	5	5	1
Soldado1X4	6	2	4	1
Soldado1X8	9	1	4	1
Soldado1X3	6	3	4	1
Soldado1X2	1	7	4	1
Soldado1X9	9	0	3	1
Soldado1X0	0	9	2	1
Soldado1X7	6	5	1	1
Soldado1X5	7	2	1	1
Soldado1X1	9	7	1	1

EQUIPO 2

ORDENAR POR PUNTOS: Selección

Nombre	Fila	Columna	Puntos Vida	Equipo
Soldado2X4	4	7	5	2
Soldado2X0	0	6	5	2
Soldado2X7	3	7	4	2
Soldado2X2	4	3	4	2
Soldado2X6	1	5	3	2
Soldado2X3	0	7	2	2
Soldado2X5	5	0	1	2
Soldado2X1	9	9	1	2

GANÓ EL EQUIPO 1

BUILD SUCCESSFUL (total time: 0 seconds)

EJEMPLO 2

```
Full:
*****JUEGO V. 3*****
El primer equipo tendrá: 2
El segundo equipo tendrá: 8
Se estan generando los soldados ---
MOSTRAR MATRIZ BIDIMENSIONAL DE SOLDADOS

F/C |  A  |  B  |  C  |  D  |  E  |  F  |  G  |  H  |  I  |  J
-----
1 |  |  |  |  |  |  |  |  |  | Soldado1X4 |
-----
2 |  |  |  |  |  |  |  |  |  |  |  |
-----
3 |  |  |  |  |  |  |  |  |  |  |  |
-----
4 |  |  |  |  |  |  | Soldado1X0 |  | Soldado1X3 |
-----
5 |  |  |  |  |  |  |  |  |  |  |  |
-----
6 |  |  |  |  |  |  |  |  |  |  | Soldado1X1 |
-----
7 |  |  |  |  |  |  |  | Soldado1X2 |  |  |
-----
8 |  |  |  |  |  |  |  |  |  |  |  |
-----
9 |  |  |  |  |  |  |  |  |  |  |  |
-----
10 |  | Soldado1X1 |  |  | Soldado1X0 |  |  |  |  |
-----

MOSTRAR SOLDADO CON MAYOR CANTIDAD DE VIDA

EQUIPO 1
Nombre  Fila  Columna  Puntos Vida  Equipo
Soldado1X1  9      1          3          1
Soldado1X0  9      4          3          1

EQUIPO 2
Nombre  Fila  Columna  Puntos Vida  Equipo
Soldado1X4  0      8          4          2
Soldado1X3  3      8          4          2

MOSTRAR PROMEDIO Y NIVEL DE VIDA

EQUIPO 1
EL NIVEL DE VIDA ES: 6.0
EL PROMEDIO DE VIDA ES: 3.0

EQUIPO 2
EL NIVEL DE VIDA ES: 14.0
EL PROMEDIO DE VIDA ES: 3.8

SOLDADOS POR ORDEN DE CREACIÓN

EQUIPO 1
Nombre  Fila  Columna  Puntos Vida  Equipo
Soldado1X0  9      4          3          1
Soldado1X1  9      1          3          1

EQUIPO 2
Nombre  Fila  Columna  Puntos Vida  Equipo
Soldado1X0  3      5          2          2
Soldado1X1  5      9          3          2
Soldado1X2  6      6          1          2
Soldado1X3  3      8          4          2
Soldado1X4  0      8          4          2

ORDENAMIENTO POR BURBUJA DEL PRIMER GRUPO

EQUIPO 1
ORDENAR POR PUNTOS: Burbuja
Nombre  Fila  Columna  Puntos Vida  Equipo
Soldado1X1  9      1          3          1
Soldado1X0  9      4          3          1

EQUIPO 2
ORDENAR POR PUNTOS: Selección
Nombre  Fila  Columna  Puntos Vida  Equipo
Soldado1X4  0      8          4          2
Soldado1X3  3      8          4          2
Soldado1X1  5      9          3          2
Soldado1X0  3      5          2          2
Soldado1X2  6      6          1          2

GANÓ EL EQUIPO 2
BUILD SUCCESSFUL (total time: 0 seconds)
```