

Universidad Nacional San Agustín de Arequipa

Facultad de Ingeniería de Producción y Servicios

Escuela Profesional de Ingeniería de Sistemas



Asignatura:

Laboratorio – Fundamentos de la Programación II

Tema:

Laboratorio Practica 7

Docente:

Revilla Arroyo Christian Alain

Estudiante:

Cuno Cahuari Armando Steven

Informe de Laboratorio 07

Tema: ArrayList y Arreglos Estandar

Actividad: Laboratorio 07

URL Repositorio GitHub: <https://github.com/SteveArms/LaboratorioFPII.git>

Actividades:

1. Se creó primero la clase Soldado en la cual nos importa el nombre, fila, columna y vida asimismo añadí aparte la figura representada.

```
1  public class Soldado {
2      private String nombre;
3      private int posf;
4      private char posc;
5      private int p_vida;
6      private char fig;
7      public Soldado(String n, int l, char m, int x, char w){
8          nombre = n;
9          posf = l;
10         posc = m;
11         p_vida = x;
12         fig = w;
13     }
14     public void setPosF(int n){
15         posf = n;
16     }
17     public void setPosC(char n){
18         posc = n;
19     }
20     public String getNombre(){
21         return nombre;
22     }
23     public int getPosF(){
24         return posf;
25     }
26     public char getPosC(){
27         return posc;
28     }
29     public int getVida(){
30         return p_vida;
31     }
32     public char getFigura(){
33         return fig;
34     }
35 }
```

A este lo acompañamos de métodos los cuales nos entreguen los datos que nosotros ingresamos, esto nos ayudara cuando queremos hacer un intercambio de valores en los métodos de ordenamiento que se nos pide.

2. Primero empezaremos con el nro aleatorio por cada ejercito y asimismo los datos que tendrán sus soldados para ello, se hizo uso de arreglos y la parte de listas se uso para el método de ordenamiento.

```
int n1 = (int)(Math.random() * 9 + 1);
System.out.println("La cantidad de soldados del ejercito 1 es : " + n1);
int n2 = (int)(Math.random() * 9 + 1);
System.out.println("La cantidad de soldados del ejercito 2 es : " + n2);
Soldado[] ejer1 = new Soldado[n1];
Soldado[] ejer2 = new Soldado[n2];
```

Si bien para el llenado de datos se hizo uso de metodos y asimismo se fue llenando en el tablero sus posiciones respectivas

```
public static void datSoldados(Soldado[] m, char n, int nEjer){
    for(int i = 0; i < m.length; i++){
        String nombre = "Soldado" + (i + 1) + "X" + nEjer;
        int vida = vidAl();
        int fila = posfAl();
        char columna = poscAl();
        m[i] = new Soldado(nombre, fila, columna, vida, n);
    }
}

public static char[][] llenadoTablero(Soldado[] n, char[][] t){
    for(int i = 0; i < n.length; i++){
        int f = n[i].getPosF();
        char c = n[i].getPosC();
        while(t[f][posCol(c)] != '-'){}
        f = posfAl();
        c = poscAl();
        n[i].setPosF(f);
        n[i].setPosC(c);
        t[n[i].getPosF()][posCol(n[i].getPosC())] = n[i].getFigura();
    }
    return t;
}
```

El llenadoTablero aparte de colocar la figura en su posición tiene el rol de cumplir con que la posición si en caso se vea ocupada por otra este vea la manera de como colocarla en otro.

Asimismo se tuvo el uso de metodos auxiliares para los datos aleatorios.

```
public static int vidAl(){
    return (int)(Math.random()* 4 + 1);
}

public static int posfAl(){
    return (int)(Math.random()* 10);
}

public static char poscAl(){
    int n = (int)(Math.random()* 10);
    String a = "abcdefghij";
    return a.charAt(n);
}

public static int posCol(char n){
    switch(n){
        case 'a': return 0;
        case 'b': return 1;
        case 'c': return 2;
        case 'd': return 3;
        case 'e': return 4;
        case 'f': return 5;
        case 'g': return 6;
        case 'h': return 7;
        case 'i': return 8;
        case 'j': return 9;
        default: return 0;
    }
}
```

3. Empezamos con el tablero para ello se uso un arreglo bidimensional de char :

```
char[][] tablero = {{"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"},
                    {"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"},
                    {"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"},
                    {"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"},
                    {"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"},
                    {"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"},
                    {"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"},
                    {"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"},
                    {"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"},
                    {"-", "-", "-", "-", "-", "-", "-", "-", "-", "-"}
};
```

Anteriormente se mostro el método llenarTablero pero asimismo se tiene lo que viene a ser el imprimirTablero

```
public static void impTablero(char[][] tablero){
    for(int i = 0; i < tablero.length; i++){
        for(int j = 0; j < tablero[i].length; j++){
            System.out.print(tablero[i][j]);
        }
        System.out.println();
    }
}
```

4. En algunos metodos haremos uso de ArrayList para ello:

```
ArrayList<Soldado> Soldados = ambosEjercitos(ejer1, ejer2);  
ArrayList<Soldado> orden1 = Soldados;  
ArrayList<Soldado> orden2 = Soldados;
```

El método ambosEjercitos se encarga de colocar los datos de ambos arreglos en el ArrayList

```
public static ArrayList<Soldado> ambosEjercitos(Soldado[] n1, Soldado[] n2){  
    ArrayList<Soldado> m = new ArrayList<Soldado>();  
    for(int i = 0; i < n1.length; i++){  
        m.add(n1[i]);  
    }  
    for(int j = 0; j < n2.length; j++){  
        m.add(n2[j]);  
    }  
    return m;  
}
```

5. Ahora como bien pedia en el laboratorio 6 tiene la misma mecánica mediante opciones para ello.

```
28      System.out.println("Opcion 1: Imprimir Tablero \nOpcion 2: Soldado con mayor vida del ejercito1\n" +
29                          "Opcion 3: Soldado con mayor vida del ejercito2\nOpcion 4: Promedio de puntos del ejercito 1\n"
30                          "Opcion 5: Promedio de puntos del ejercito 2\nOpcion 6: Mostrar datos de todos los soldados\n"
31                          "Opcion 7: Ranking de poder de ambos ejercito - Metodo 1\n" +
32                          "Opcion 8: Ranking de poder de ambos ejercitos - Metodo 2\nOpcion 9: Ejercito ganador\n" +
33                          "Opcion 10: Salir");
34      System.out.println("Ingrese una opcion :");
35      int v = sc.nextInt();
36      while(v != 10){
37          if(v == 1){
38              impTablero(tablero);
39              System.out.println("*****");
40          }
41          if(v == 2){
42              System.out.println("Mayor vida del ejercito 1");
43              mayVida(ejer1);
44              System.out.println("*****");
45          }
46          if(v == 3){
47              System.out.println("Mayor vida del ejercito 2");
48              mayVida(ejer2);
49              System.out.println("*****");
50          }
51          if(v == 4){
52              System.out.println("Promedio de puntos del ejercito 1");
53              promPuntos(ejer1);
54              System.out.println("*****");
55          }
56          if(v == 5){
57              System.out.println("Promedio de puntos del ejercito 2");
58              promPuntos(ejer2);
59              System.out.println("*****");
60          }
61          if(v == 6){
62              impSoldados(Soldados);
63              System.out.println("*****");
64          }
65          if(v == 7){
66              System.out.println("RANKING 1er metodo: *****");
67              rankSoldados1(orden1);
68              System.out.println("*****");
69          }
70          if(v == 8){
71              System.out.println("RANKING 2do metodo: *****");
72              rankSoldados2(orden2);
73              System.out.println("*****");
74          }
75          if(v == 9){
76              ejercitoGanador(ejer1, ejer2);
77              System.out.println("*****");
78          }
79          System.out.println("Ingrese una opcion :");
80          v = sc.nextInt();
81      }
```

6. Metodo 1 : Encargado de mostrar el tablero

```
184      public static void impTablero(char[][] tablero){
185          for(int i = 0; i < tablero.length; i++){
186              for(int j = 0; j < tablero[i].length; j++){
187                  System.out.print(tablero[i][j]);
188              }
189              System.out.println();
190          }
191      }
```

7. Metodo 2 y 3: Encargado de mostrar los soldados con mayor vida por cada ejercito

```
public static void mayVida(Soldado[] n){
    int mayor = 0;
    for(int i = 1; i < n.length; i++){
        if(n[i].getVida() > n[mayor].getVida()){
            mayor = i;
        }
    }
    for(int j = 0; j < n.length; j++){
        if(n[j].getVida() == n[mayor].getVida()){
            System.out.println(n[j].getNombre() + " " + n[j].getVida() + " " + (n[j].getPosF() + 1) + " " + n[j].getPosC());
        }
    }
}
```

8. Metodo 4 y 5: Encargado de mostrar el promedio de vida entre cada ejercito

```
125 public static void promPuntos(Soldado[] n){
126     int sum = 0;
127     for(int i = 0; i < n.length; i++){
128         sum += n[i].getVida();
129     }
130     System.out.println("El promedio de puntos es " + (sum / n.length));
131 }
```

9. Metodo 6: Muestra los datos de todos los soldados, para ello como bien hemos creado el ArrayList con ambos ejércitos daremos uso de este:

```
179 public static void impSoldados(ArrayList<Soldado> n){
180     for(int i = 0; i < n.size(); i++){
181         System.out.println("Soldado Nro" + (i + 1) + " " + n.get(i).getNombre() + " " + n.get(i).getVida() + " " + n.get(i).getPosF() + " " + n.get(i).getPosC());
182     }
183 }
```

El “Soldado Nro” nos ayuda a contabilizar si en caso se cumple que la suma de ambas cantidades de ejércitos es igual

10. Metodo 7 y 8: Como bien creamos el ArrayList con ambos ejércitos es viable ahora crear otros ArrayList sin modificar el original y a estos aplicarles métodos de ordenamiento

Dando uso tanto al método de Selección y Burbuja

```
133 public static void rankSoldados1(ArrayList<Soldado> n){
134     for(int i = 0; i < n.size() - 1; i++){
135         for(int j = i + 1; j < n.size(); j++){
136             if(n.get(i).getVida() < n.get(j).getVida()){
137                 Soldado mayor = new Soldado(n.get(j).getNombre(), n.get(j).getPosF(), n.get(j).getPosC(), n.get(j).getVida());
138                 Soldado menor = new Soldado(n.get(i).getNombre(), n.get(i).getPosF(), n.get(i).getPosC(), n.get(i).getVida());
139                 n.set(i, mayor);
140                 n.set(j, menor);
141             }
142         }
143     }
144     impSoldados(n);
145 }
146 public static void rankSoldados2(ArrayList<Soldado> n){
147     for(int i = 0; i < n.size() - 1; i++){
148         for(int j = 0; j < n.size() - 1; j++){
149             if(n.get(j).getVida() < n.get(j + 1).getVida()){
150                 Soldado mayor = new Soldado(n.get(j + 1).getNombre(), n.get(j + 1).getPosF(), n.get(j + 1).getPosC(), n.get(j + 1).getVida());
151                 Soldado menor = new Soldado(n.get(j).getNombre(), n.get(j).getPosF(), n.get(j).getPosC(), n.get(j).getVida());
152                 n.set(j, mayor);
153                 n.set(j + 1, menor);
154             }
155         }
156     }
157     impSoldados(n);
158 }
```

En ambos al momento de intercambiar los valores por ser mayor hacemos uso de un objeto de apoyo llamado “mayor” y “menor” ayudados del método set que otorga el ArrayList hacemos el intercambio

11. Metodo 9 : Nos muestra al ganador del cual se obtiene a partir de la suma mayor por puntos de vida de cada ejercito

```

160     public static void ejercitoGanador(Soldado[] n, Soldado[] m){
161         int sum1 = 0, sum2 = 0;
162         for(int i = 0; i < n.length; i++){
163             sum1 += n[i].getVida();
164         }
165         for(int j = 0; j < m.length; j++){
166             sum2 += m[j].getVida();
167         }
168         System.out.println("Total de puntos del ejercito 1 " + sum1);
169         System.out.println("Total de puntos del ejercito 2 " + sum2);
170         if(sum1 == sum2){
171             System.out.println("Hay un empate");
172         } else if(sum1 > sum2){
173             System.out.println("El ganador es el ejercito 1");
174         } else{
175             System.out.println("El ganador es el ejercito 2");
176         }
177     }

```

Prueba:

```

La cantidad de soldados del ejercito 1 es : 8
La cantidad de soldados del ejercito 2 es : 3
Opcion 1: Imprimir Tablero
Opcion 2: Soldado con mayor vida del ejercito1
Opcion 3: Soldado con mayor vida del ejercito2
Opcion 4: Promedio de puntos del ejercito 1
Opcion 5: Promedio de puntos del ejercito 2
Opcion 6: Mostrar datos de todos los soldados
Opcion 7: Ranking de poder de ambos ejercito - Metodo 1
Opcion 8: Ranking de poder de ambos ejercitos - Metodo 2
Opcion 9: Ejercito ganador
Opcion 10: Salir

```

Opcion 1:

```

Ingrese una opcion :
1
-----*-*--
-----*--
-----$---
-----
$---*-----
-----*-*--
-----
--$*---*--
-----
-----
*****

```

Opcion 2 y 3:

```

Ingrese una opcion :
2
Mayor vida del ejercito 1
Soldado2X1 4 1 h *
Soldado4X1 4 8 e *
Soldado5X1 4 8 i *
*****

```

```

Ingrese una opcion :
3
Mayor vida del ejercito 2
Soldado2X2 4 8 d $
*****

```


Opcion 4 y 5:

```
Ingrese una opcion :  
4  
Promedio de puntos del ejercito 1  
El promedio de puntos es 3  
*****
```

```
Ingrese una opcion :  
5  
Promedio de puntos del ejercito 2  
El promedio de puntos es 2  
*****
```

Opcion 6:

```
Ingrese una opcion :  
6  
Soldado Nro1 Soldado1X1 3 4 e *  
Soldado Nro2 Soldado2X1 4 0 h *  
Soldado Nro3 Soldado3X1 2 5 i *  
Soldado Nro4 Soldado4X1 4 7 e *  
Soldado Nro5 Soldado5X1 4 7 i *  
Soldado Nro6 Soldado6X1 3 0 f *  
Soldado Nro7 Soldado7X1 2 5 f *  
Soldado Nro8 Soldado8X1 2 1 i *  
Soldado Nro9 Soldado1X2 1 2 g $  
Soldado Nro10 Soldado2X2 4 7 d $  
Soldado Nro11 Soldado3X2 3 4 a $  
*****
```

Opcion 7 y 8:

```
Ingrese una opcion :  
7  
RANKING 1er metodo: *****  
Soldado Nro1 Soldado2X1 4 0 h *  
Soldado Nro2 Soldado4X1 4 7 e *  
Soldado Nro3 Soldado5X1 4 7 i *  
Soldado Nro4 Soldado2X2 4 7 d $  
Soldado Nro5 Soldado6X1 3 0 f *  
Soldado Nro6 Soldado1X1 3 4 e *  
Soldado Nro7 Soldado3X2 3 4 a $  
Soldado Nro8 Soldado8X1 2 1 i *  
Soldado Nro9 Soldado3X1 2 5 i *  
Soldado Nro10 Soldado7X1 2 5 f *  
Soldado Nro11 Soldado1X2 1 2 g $  
*****
```

```
Ingrese una opcion :  
8  
RANKING 2do metodo: *****  
Soldado Nro1 Soldado2X1 4 0 h *  
Soldado Nro2 Soldado4X1 4 7 e *  
Soldado Nro3 Soldado5X1 4 7 i *  
Soldado Nro4 Soldado2X2 4 7 d $  
Soldado Nro5 Soldado6X1 3 0 f *  
Soldado Nro6 Soldado1X1 3 4 e *  
Soldado Nro7 Soldado3X2 3 4 a $  
Soldado Nro8 Soldado8X1 2 1 i *  
Soldado Nro9 Soldado3X1 2 5 i *  
Soldado Nro10 Soldado7X1 2 5 f *  
Soldado Nro11 Soldado1X2 1 2 g $  
*****
```

Opcion 9:

```
Ingrese una opcion :  
9  
Total de puntos del ejercito 1 24  
Total de puntos del ejercito 2 8  
El ganador es el ejercito 1  
*****
```