


	<p align="center">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		
Aprobación: 2022/03/01	Código: GUIA-PRLE-001	Página: 1

INFORME DE LABORATORIO

INFORMACIÓN BÁSICA					
ASIGNATURA:	Fundamentos de la Programacion 2				
TÍTULO DE LA PRÁCTICA:	<i>Laboratorio 12</i>				
NÚMERO DE PRÁCTICA:	<i>12</i>	AÑO LECTIVO:	<i>1</i>	NRO. SEMESTRE:	<i>2</i>
FECHA DE PRESENTACIÓN	<i>20/12/2023</i>	HORA DE PRESENTACIÓN	<i>4:55pm</i>		
INTEGRANTE (s): Cuno Cahuari Armando Steven				NOTA:	
DOCENTE(s): Alain Revilla Christian					

SOLUCIÓN Y RESULTADOS	
<p>I. SOLUCIÓN DE EJERCICIOS/PROBLEMAS</p> <p>Para este laboratorio primero se inicio con un menú con 3 opciones las cuales fueron juegoRapido, juegoPersonalizado y Salir.</p> <pre> public static void opcionesJuego(ArrayList<Soldado> ejercito1, ArrayList<Soldado> ejercito2, char[][]tablero){ Scanner sc = new Scanner(System.in); boolean validez = true; int opcion; while(validez){ System.out.println("Ingrese una opcion"); System.out.println("Opcion 1: Juego Rapido\nOpcion 2: Juego Personalizado\nOpcion 3: Salir"); opcion = sc.nextInt(); switch(opcion){ case 1: System.out.println("Se inicio una partida rapida"); juegoRapido(ejercito1, ejercito2, tablero); break; case 2: System.out.println("Se inicio una partida personalizada"); juegoPersonalizado(ejercito1, ejercito2, tablero); break; case 3: validez = false; break; } } } </pre>	

	<p style="text-align: center;">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p style="text-align: center;">Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 2</p>

El juegoRapido es el mismo método juego del laboratorio anterior solo que en este caso nos requiere que en cada momento consultemos si queremos seguir jugando si en caso no pues se cancela el juego.

En la opción de juegoRapido:

```
public static void juegoRapido(ArrayList<Soldado> ejercito1, ArrayList<Soldado> ejercito2, char[][]tablero){
    Scanner sc = new Scanner(System.in);
    int opcion;
    boolean validez = true;
    while(validez){
        juego(ejercito1, ejercito2, tablero);
        System.out.println("Ingrese otra opcion");
        opcion = sc.nextInt();
        switch(opcion){
            case 1:
                System.out.println("Se inicio nuevamente la partida");
                juego(ejercito1, ejercito2, tablero);
                break;
            default:
                System.out.println("Redireccionando al menu principal");
                validez = false;
                break;
        }
    }
}
```

En esta opción nos dara para iniciar el juego o si bien para salir de este



```
public static void juego(ArrayList<Soldado> ejercito1, ArrayList<Soldado> ejercito2, char[][] tablero){
    Scanner sc = new Scanner(System.in);
    boolean validez = true;
    String coordenada;
    int movimiento, fila, columna;
    String jugar = "";
    int i = 0;
    while(validez){
        if(ejercito1.size()== 0 || ejercito2.size() == 0 || jugar.equals("NO")){
            validez = false;
            break;
        } else if(i % 2 == 0){
            System.out.println("Turno del Ejercito 1");
            coordenada = ingresar(ejercito1.get(0).getFigura(), tablero);
            movimiento = ingresarMovimiento(coordenada);
            movimientoJugado(coordenada, movimiento, ejercito1, ejercito2, tablero);
            impTablero(tablero);
        } else {
            System.out.println("Turno del Ejercito 2");
            coordenada = ingresar(ejercito2.get(0).getFigura(), tablero);
            movimiento = ingresarMovimiento(coordenada);
            movimientoJugado(coordenada, movimiento, ejercito2, ejercito1, tablero);
            impTablero(tablero);
        }
        System.out.println("Desea seguir jugando?");
        jugar = sc.next();
        jugar = jugar.toUpperCase();
        i++;
    }
    if(ejercito1.size() == 0){
        System.out.println("Salio victorio el ejercito 2");
        for(Soldado n: ejercito2){
            System.out.println(n);
        }
    } else if(ejercito2.size()==0){
        System.out.println("Salio victorioso el ejercito 1");
        for(Soldado m: ejercito2){
            System.out.println(m);
        }
    } else {
        System.out.println("Se cancelo el juego");
    }
}
```

Como vemos al momento de iniciar el while en el juego aparte de para romper ese while reconozca que si en caso el ArrayList de ambos ejércitos este en vacío para anular el juego pues también se agrega la opción de que se consulte si se desea seguir jugando en caso no se cancela el juego.

Acerca del panel de JuegoPersonalizado:

Para este se usó diversos métodos los cuales son necesarios para cumplir con el rol de Juego Personalizado mediante diversos métodos los cuales :

```
public static void juegoPersonalizado(ArrayList<Soldado> ejercito1, ArrayList<Soldado> ejercito2, char[][] tabl
Scanner sc = new Scanner(System.in);
ArrayList<Soldado> ejercitoElegido;
int opcion;
boolean validez = true;
while(validez){
    System.out.println("¿Que ejercito deseas aplicar modificacion?");
    ejercitoElegido = elegido(ejercito1, ejercito2);
    System.out.println("Opcion 1: Ingresar Soldado\nOpcion 2: Eliminar Soldado\nOpcion 3: Clonar Soldado" +
"\nOpcion 4: Modificar Soldado\nOpcion 5: Comparar Soldados\nOpcion 6: Intercambiar Soldados" +
"\nOpcion 7: Ver Soldado\nOpcion 8: Ver ejercito\nOpcion 9: Suma de Niveles\nOpcion 10: Iniciar juego"
"\nOpcion 11: Salir al menu principal");
    System.out.println("Ingrese la opcion que desea realizar:");
    opcion = sc.nextInt();
    switch(opcion){
        case 1:
            System.out.println("Ingresar soldado:");
            addSoldado(ejercitoElegido, tablero);
            break;
        case 2:
            System.out.println("Eliminar soldado:");
            eliminarSoldado(ejercitoElegido, tablero);
            break;
        case 3:
            System.out.println("Clonar soldado:");
            clonarSoldado(ejercitoElegido, tablero);
            break;
        case 4:
            System.out.println("Modificar soldado:");
            modificarSoldado(ejercitoElegido);
            break;
        case 5:
            System.out.println("Comparar soldados:");
            compararSoldados(ejercitoElegido);
            break;
        case 6:
            System.out.println("Intercambiar soldados:");
            intercambiarSoldados(ejercitoElegido);
            break;
        case 7:
            System.out.println("Ver Soldado:");
            verSoldado(ejercitoElegido);
            break;
        case 8:
            System.out.println("Ver Ejercito:");
            verEjercito(ejercitoElegido);
            break;
        case 9:
            System.out.println("Sumar niveles:");
            sumarNiveles(ejercitoElegido);
            break;
        case 10:
            juego(ejercito1, ejercito2, tablero);
```

	<p style="text-align: center;">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p style="text-align: center;">Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 5</p>

```

        case 11:
            System.out.println("Redirigiendo al menu principal");
            validez = false;
            break;
        default:
            System.out.println("Opcion invalida");
    }
}

```

Con la opción 1:

```

public static void addSoldado(ArrayList<Soldado> ejercito, char[][] tablero){
    Scanner sc = new Scanner(System.in);
    if(ejercito.size() == 10){
        System.out.println("Ya no es posible agregar un soldado a este ejercito ya que cuenta con 10 soldados")
    } else {
        System.out.println("Ingrese el nombre");
        String nombre = sc.next();
        System.out.println("Ingrese el nivel de Vida");
        int nivVida = sc.nextInt();
        Soldado nuevo = new Soldado(nombre, nivVida, ejercito.get(0).getFigura());
        ejercito.add(nuevo);
        actTablero(ejercito, tablero);
    }
}

```



Tenemos el agregarSoldado tiene la condición que el arrayList presente no rebase de los 10 soldados que este contega sino en todo caso le saldrá un mensaje asimismo si en caso no pues este le pedirá el nombre y nivel de vida para agregar. Con el método actTablero nos aseguramos que ocupe un espacio vacío.

```

public static void actTablero(ArrayList<Soldado> ejercito, char[][] tablero){
    int fila;
    char columna;
    for(int i = 0; i < ejercito.size(); i++){
        fila = ejercito.get(i).getFila();
        columna = ejercito.get(i).getColumna();
        while(tablero[fila - 1][nroColumna(columna)] != '-'){
            fila = (int)(Math.random() * Soldado.CANTMAX + 1);
            columna = Soldado.numCol();
        }
        ejercito.get(i).setFila(fila);
        ejercito.get(i).setCol(columna);
        tablero[fila - 1][nroColumna(columna)] = ejercito.get(i).getFigura();
    }
}

```

La 2da opción:

	<p style="text-align: center;">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p style="text-align: center;">Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 6</p>

```

public static void eliminarSoldado(ArrayList<Soldado> ejercito, char[][] tablero){
    Scanner sc = new Scanner(System.in);
    if(ejercito.size() == 1){
        System.out.println("No es posible eliminar ya que este es el ultimo soldado de este ejercito");
    } else {
        System.out.println("Datos de los Soldados");
        imprimirDatosEjercito(ejercito);
        System.out.println("Que soldado desea eliminar");
        int indice = sc.nextInt();
        System.out.println("Se eliminara el soldado Nro " + indice);
        System.out.println("Con datos: " + ejercito.get(indice));
        ejercito.remove(indice);
        actTablero(ejercito, tablero);
    }
}

```

Este método elimina un soldado del ArrayList la condición es que este ArrayList no se elimine por completo para ello esta la condición que si el tamaño es 1 pues este no se aplicara en caso contrario si pidiendo el índice del soldado a eliminar.

La 3ra opción:



```

public static void clonarSoldado(ArrayList<Soldado> ejercito, char[][] tablero){
    Scanner sc = new Scanner(System.in);
    if(ejercito.size() == 10){
        System.out.println("No es posible clonar porque el ejercito ya cuenta con 10 soldados");
    } else {
        System.out.println("Datos de los Soldados");
        imprimirDatosEjercito(ejercito);
        System.out.println("Que soldado deseas clonar");
        int indice = sc.nextInt();
        Soldado original = ejercito.get(indice);
        Soldado copia = new Soldado(original.getNombre(), original.getAtaque(), original.getDefensa(), original.getVelocidad());
        ejercito.add(copia);
        actTablero(ejercito, tablero);
    }
}

```

El metodo clonarSoldado se encarga de lo mismo del método agregar que primero se condiciona que el ArrayList no pase de los 10 elementos en caso contrario le pedirá el índice del Soldado que quiere clonar para agregarlo al ultimo asimismo a este le aplicamos agregación hacia el arraylist presente.

La 4ta opción:

	<p align="center">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		
Aprobación: 2022/03/01	Código: GUIA-PRLE-001	Página: 7



```

public static void modificarSoldado(ArrayList<Soldado> ejercito){
    Scanner sc = new Scanner(System.in);
    int valor;
    System.out.println("Datos de los Soldados");
    imprimirDatosEjercito(ejercito);
    System.out.println("Que soldado deseas modificar");
    int indice = sc.nextInt();
    System.out.println("Opcion 1: Nivel de Ataque\nOpcion 2: Nivel de defensa\nOpcion 3: Vida Actual");
    System.out.println("Ingrese la opcion que desea modificar");
    int opcion = sc.nextInt();
    System.out.println("Ingrese la cantidad que quiere cambiar:");
    valor = sc.nextInt();
    if(opcion == 1){
        if(valor <= 5 && valor>= 1){
            System.out.println("Cambio realizado");
            ejercito.get(indice).setAtaque(valor);
        } else {
            System.out.println("No es posible realizar el cambio porque rebasa los limites");
        }
    } else if(opcion == 2){
        if(valor <= 5 && valor>= 1){
            System.out.println("Cambio realizado");
            ejercito.get(indice).setDefensa(valor);
        } else {
            System.out.println("No es posible realizar el cambio porque rebasa los limites");
        }
    } else if(opcion == 3){
        if(valor <= 5 && valor>= 1){
            System.out.println("Cambio realizado");
            ejercito.get(indice).setNivVidAct(valor);
        } else {
            System.out.println("No es posible realizar el cambio porque rebasa los limites");
        }
    } else{
        System.out.println("Opcion invalida");
    }
}

```

En este método de modificar es necesario indicar que valor va a querer modificar y el índice del elemento a modificar , una ves se solicita el valor a cambiar pues este sera evaluado si esta dentro de un parámetro del 1 al 5 en caso no botara un mensaje de opción invalida.

La 5ta opción:

	<p style="text-align: center;">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p style="text-align: center;">Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 8</p>

```

public static void compararSoldados(ArrayList<Soldado> ejercito){
    Scanner sc = new Scanner(System.in);
    System.out.println("Datos de los soldados");
    imprimirDatosEjercito(ejercito);
    System.out.println("Escoja el indice del 1er soldado");
    int primero = sc.nextInt();
    System.out.println("Escoja el indice del 2do soldado");
    int segundo = sc.nextInt();
    Soldado uno = ejercito.get(primero);
    Soldado dos = ejercito.get(segundo);
    if(comparar(uno, dos)){
        System.out.println("Son iguales");
    } else {
        System.out.println("No son iguales");
    }
}

public static boolean comparar(Soldado uno, Soldado dos){
    return uno.getNombre().equals(dos.getNombre()) && uno.getAtaque() == dos.getAtaque() && uno.getDefensa() == dos.getDefensa();
}

```

En este método se comparara para lo cual se requiere 2 indices necesarios para comparar tanto el nombre, ataque, defensa , vive y nivActual. Con ayuda de otro método comparar lo cual nos retornara un boolean.

La 6ta opción:

```

public static void intercambiarSoldados(ArrayList<Soldado> ejercito){
    Scanner sc = new Scanner(System.in);
    System.out.println("Datos de los soldados");
    imprimirDatosEjercito(ejercito);
    System.out.println("Escoja el indice del 1er soldado");
    int primero = sc.nextInt();
    System.out.println("Escoja el indice del 2do soldado");
    int segundo = sc.nextInt();
    Soldado original = ejercito.get(primero);
    Soldado copia = new Soldado(original.getNombre(), original.getAtaque(), original.getDefensa(), original.getVive(), original.getNivActual());
    ejercito.set(primero, ejercito.get(segundo));
    ejercito.set(segundo, copia);
    System.out.println("Intercambio realizado");
}

```

El 6to método nos pide intercambiar soldados de ubicación dentro de la estrucutra del ArrayList para ello es necesario requerir de 2 indices con ayuda de los métodos que nos ofrecen el ArrayList hacemos el intercambio mediante sets

La 7ma opción:


```
public static void verSoldado(ArrayList<Soldado> lista){
    Scanner sc = new Scanner(System.in);
    int indice;
    System.out.println("Ingrese el nombre que esta buscando");
    String nombre = sc.next();
    if(nombreUbicado(lista, nombre)){
        for(int i = 0; i < lista.size(); i++){
            if(lista.get(i).getNombre().equals(nombre)){
                indice = i;
                System.out.println(lista.get(i));
                break;
            }
        }
    } else {
        System.out.println("No se encontro a nadie con ese nombre");
    }
}
```

Para el 7mo método de verSoldado se requerirá un String con el nombre haciendo una búsqueda lineal para buscar a lo largo del ArrayList mediante índices e imprimir el valor de este en caso contrario este botara que no se encontró a nadie.

La 8va opción:

```
public static void verEjercito(ArrayList<Soldado> ejercito){
    System.out.println("Datos de los soldados");
    imprimirDatosEjercito(ejercito);
}
```

Mediante el apoyo del método imprimirDatosEjercito pues se imprimirá elemento por elemento



```
public static void imprimirDatosEjercito(ArrayList<Soldado> ejercito){
    for(int i = 0; i < ejercito.size(); i++){
        System.out.println((i + 1) + " " +ejercito.get(i));
    }
}
```

La 9na opción es:

```
public static void sumarNiveles(ArrayList<Soldado> ejercito){
    int valorVidaT = 0;
    int valorAtaqueT = 0;
    int valorDefensaT = 0;
    int valorVelocidadT = 0;
    for(Soldado m: ejercito){
        valorVidaT += m.getVida();
        valorAtaqueT += m.getAtaque();
        valorDefensaT += m.getDefensa();
        valorVelocidadT += m.getVelocidad();
    }
    System.out.println("El valor de vida total es : " + valorVidaT + "\nEl valor de ataque total es: " + valorAtac
    "\nEl valor de defensa total es: " + valorDefensaT + "\n El valor de velocidad total es: " + valorVelocidadT);
}
```

La suma de niveles de todos los objetos del ArrayList mediante la acumulación del foreach por 4 variables.

La 10ma opción:

	<p>UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p>Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 10</p>

Inicia el juego con los valores ya modificados

La 11va opción:



Redirige al menú principal

Ejemplo:



```

Inicio de Juego
El tablero es :
--*-----
-----
-----
-----
--**-----
*_*-----
-----
-----
--*--&--&-
-----
Ingrese una opcion
Opcion 1: Juego Rapido
Opcion 2: Juego Personalizado
Opcion 3: Salir
1
Se inicio una partida rapida
Turno del Ejercito 1
1 2 3
8 x 4
7 6 5
Ingrese otra coordenada
C1
1      2
Ingresar Movimiento:
4
0 Movimiento aplicado    3
El tablero es :
--*-----
-----
-----
-----
--**-----
*_*-----
-----

```

	<p>UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p>Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 11</p>



<p>Desea seguir jugando? NO Se cancelo el juego Ingrese otra opcion 12 Redireccionando al menu principal Ingrese una opcion Opcion 1: Juego Rapido Opcion 2: Juego Personalizado Opcion 3: Salir 3 Saliendo del programa</p>	
--	--

	<p>UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p>Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 12</p>

```

Ingrese una opcion
Opcion 1: Juego Rapido
Opcion 2: Juego Personalizado
Opcion 3: Salir
2
Se inicio una partida personalizada
¿Que ejercito deseas aplicar modificacion?
Opcion 1 : Ejercito *
Opcion 2: Ejercico $
1
Opcion 1: Ingresar Soldado
Opcion 2: Eliminar Soldado
Opcion 3: Clonar Soldado
Opcion 4: Modificar Soldado
Opcion 5: Comparar Soldados
Opcion 6: Intercambiar Soldados
Opcion 7: Ver Soldado
Opcion 8: Ver ejercito
Opcion 9: Suma de Niveles
Opcion 10: Iniciar juego
Opcion 11: Salir al menu principal
Ingrese la opcion que desea realizar:
1
Ingresar soldado:
Ingrese el nombre
FERNANDO
Ingrese el nivel de Vida
5
¿Que ejercito deseas aplicar modificacion?
Opcion 1 : Ejercito *
Opcion 2: Ejercico $
2
Opcion 1: Ingresar Soldado
Opcion 2: Eliminar Soldado
Opcion 3: Clonar Soldado
Opcion 4: Modificar Soldado
Opcion 5: Comparar Soldados



```

	<p style="text-align: center;">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p style="text-align: center;">Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 13</p>

```
Opcion 6: Intercambiar Soldados
Opcion 7: Ver Soldado
Opcion 8: Ver ejercito
Opcion 9: Suma de Niveles
Opcion 10: Iniciar juego
Opcion 11: Salir al menu principal
Ingrese la opcion que desea realizar:
8
Ver Ejercito:
Datos de los soldados
1 Nombre: Soldado 1X2 Vida:5 Fila:5 Columna:E
2 Nombre: Soldado 2X2 Vida:2 Fila:3 Columna:F
3 Nombre: Soldado 3X2 Vida:1 Fila:5 Columna:I
4 Nombre: Soldado 4X2 Vida:4 Fila:6 Columna:H
5 Nombre: Soldado 5X2 Vida:4 Fila:6 Columna:I
¿Que ejercito deseas aplicar modificacion?
Opcion 1 : Ejercito *
Opcion 2: Ejercicio $
1
Opcion 1: Ingresar Soldado
Opcion 2: Eliminar Soldado
Opcion 3: Clonar Soldado
Opcion 4: Modificar Soldado
Opcion 5: Comparar Soldados
Opcion 6: Intercambiar Soldados
Opcion 7: Ver Soldado
Opcion 8: Ver ejercito
Opcion 9: Suma de Niveles
Opcion 10: Iniciar juego
Opcion 11: Salir al menu principal
Ingrese la opcion que desea realizar:
11
Redirigiendo al menu principal
Ingrese una opcion
Opcion 1: Juego Rapido
Opcion 2: Juego Personalizado
Opcion 3: Salir
```

II. CONCLUSIONES

El programa debe tener un poco mas de orden asimismo la lógica de la programación va mejorando laboratorio tras laboratorio es necesario tener ideas y variables de apoyo para el intercambio de valores.

	<p align="center">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p align="center">Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p align="center">Código: GUIA-PRLE-001</p>	<p align="right">Página: 14</p>

RETROALIMENTACIÓN GENERAL

Debo mejorar aún más el orden de los códigos asimismo la asignación de variables que empleo como el orden entre otros factores.

REFERENCIAS Y BIBLIOGRAFÍA