


	<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

(formato estudiante)

INFORMACIÓN BÁSICA					
ASIGNATURA:	Laboratorio Fundamentos de la programación 2				
TÍTULO DE LA PRÁCTICA:	Arreglos de objetos				
NÚMERO DE PRÁCTICA:	3	AÑO LECTIVO:	1	NRO. SEMESTRE:	II
FECHA DE PRESENTACIÓN	5/10/2024	HORA DE PRESENTACIÓN	23/55/00 pm		
INTEGRANTE (s) Hilacondo Begazo, Emanuel David				NOTA (0-20)	
DOCENTE(s): Ing. Lino José Pinto Oppe					

RESULTADOS Y PRUEBAS	
I. EJERCICIOS RESUELTOS:	
EJERCICIO 1:	
<p>Esta clase nos servirá para nuestro main en el ejercicio 1, como podemos ver con esto nombraremos nuestra nave, le indicaremos la fila en la que se encuentra, también la columna, después tambien indicaremos si la nave funciona o no con un boolean y por último ingresaremos los puntos de nivel que posee.</p>	
<pre> 1 package Laboratorio3; 2 /*Laboratorio Nr3 - Ejercicio1 7 public class Nave { 8 // Atributos de la clase Nave 9 private String nombre, columna; 10 private int fila, puntos; 11 private boolean estado; 12 13 // Métodos mutadores 14 public void setNombre(String nom) { 15 nombre = nom; 16 } 17 public void setFila(int fil) { 18 fila = fil; 19 } 20 public void setColumna(String col) { 21 columna = col; 22 } 23 public void setEstado(boolean est) { 24 estado = est; 25 } 26 public void setPuntos(int punt) { 27 puntos = punt; 28 } </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>

Anteriormente vimos los métodos mutadores o de entrada, es donde indicaremos el valor de la variable, estos métodos accesorios son los que nos dan acceso a esas variables ya que son privadas y por último el toString devolverá la información de cada nave.

```

30 // Métodos accesorios
31 public String getNombre() {
32     return nombre;
33 }
34 public int getFila() {
35     return fila;
36 }
37 public String getColumna() {
38     return columna;
39 }
40 public boolean getEstado() {
41     return estado;
42 }
43 public int getPuntos() {
44     return puntos;
45 }
46
47 // Método toString para mostrar la información de la nave
48 public String toString() {
49     return ("Nombre: " + getNombre() + "\nFila: " + getFila() +
50           "\nColumna: " + getColumna() + "\nEstado: " + getEstado() +
51           "\nPuntos: " + getPuntos());
52 }
53 }



```

Aquí es donde usaremos la clase anterior, primero creamos misNaves de la clase Nave y después declaramos las otras variables, luego introducimos un “for” el cual se repetirá hasta llenar el arreglo con los datos declarados anteriormente.

```

9 public class DemoBatalla {
10 public static void main(String [] args) {
11     // Se crea un arreglo de objetos Nave con capacidad para 10 naves
12     Nave [] misNaves = new Nave[10];
13     Scanner sc = new Scanner(System.in);
14     String nomb, col;
15     int fil, punt;
16     boolean est;
17
18     // Ciclo para solicitar los datos de cada nave al usuario
19     for (int i = 0; i < misNaves.length; i++) {
20         System.out.println("Nave " + (i+1));
21         System.out.print("Nombre: ");
22         nomb = sc.next();
23         System.out.print("Fila: ");
24         fil = sc.nextInt();
25         System.out.print("Columna: ");
26         col = sc.next();
27         System.out.print("Estado (true = operativa, false = dañada): ");
28         est = sc.nextBoolean();
29         System.out.print("Puntos: ");
30         punt = sc.nextInt();

```

	<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: 3</p>

Aquí declaramos a todas las naves a la clase Nave y llenamos sus datos respectivamente llamando a los métodos correspondientes de esta clase.

```

32         // Se crea una nueva nave y se asignan los valores proporcionados
33         misNaves[i] = new Nave();
34         misNaves[i].setNombre(nomb);
35         misNaves[i].setFila(fil);
36         misNaves[i].setColumna(col);
37         misNaves[i].setEstado(est);
38         misNaves[i].setPuntos(punt);
39     }

```

Imprime las naves respectivamente con el método “mostrarPorNombre”, que este posee un “for” que imprimirá el toString de la clase Nave para cada nave del arreglo de Nave (estamos usando un for-each)

```

41         // Muestra las naves creadas
42         System.out.println("\nNaves creadas:");
43         mostrarNaves(misNaves);

```

```

57     // Método para mostrar todas las naves en la flota
58     public static void mostrarNaves(Nave [] flota) {
59         for(Nave n: flota)
60             System.out.println(n + "\n");
61     }

```

El primer método te pedirá el nombre de alguna nave, si existe imprimirá todas las naves con ese nombre en caso de que no; no lo hará. El segundo método te pedirá cierta cantidad de puntos e imprimirá todas las naves que tengan esa cantidad de puntos o menos

```

// Muestra las naves filtradas por nombre y puntos
mostrarPorNombre(misNaves, sc);
mostrarPorPuntos(misNaves, sc);

```

```

63     // Método para mostrar naves por nombre
64     public static void mostrarPorNombre(Nave [] flota, Scanner sc) {
65         System.out.print("Ingrese el nombre de la flota: ");
66         String nomBuscado = sc.next();
67         for(Nave n: flota)
68             if(n.getNombre().equals(nomBuscado)) // Compara el nombre de cada nave
69                 System.out.println(n + "\n");
70     }
71
72     // Método para mostrar naves con puntos menores o iguales a los indicados
73     public static void mostrarPorPuntos(Nave [] flota, Scanner sc) {
74         System.out.print("Ingrese un número de puntos: ");
75         int cantPunt = sc.nextInt();
76         for(Nave n: flota)
77             if(n.getPuntos() <= cantPunt)
78                 System.out.println(n + "\n");
79     }

```

Este método imprimirá la nave que posea la mayor cantidad de puntos que los otros. El método compara todos los puntos de la nave y guarda la posición con la mayor cantidad de puntos

```
// Muestra la nave con mayor número de puntos
System.out.println("\nNave con mayor número de puntos:\n" + mostrarMayorPuntos(misNaves));



// Muestra las naves en orden aleatorio
```

```
82 public static Nave mostrarMayorPuntos(Nave [] flota) {
83     int pos = 0;
84     int mayorPunt = flota[pos].getPuntos();
85     for(int i = 1; i < flota.length; i++) {
86         if(mayorPunt < flota[i].getPuntos()) {
87             mayorPunt = flota[i].getPuntos();
88             pos = i;
89         }
90     }
91     return flota[pos];
92 }
```

Como vemos de este método retorna un arreglo de naves, este contiene las naves, pero en un orden diferente, el método usa el arrayCopy para copiar el arreglo y después con un Math.random() va variando las posiciones de este arreglo y al salir del bucle retorna al main y el original se guarda.

```
52 // Muestra las naves en orden aleatorio
53 System.out.println("\nLas naves desordenadas:\n");
54 mostrarNaves(desordenarNaves(misNaves));
55 }
```

```
94 // Método que desordena aleatoriamente las naves en la flota
95 public static Nave[] desordenarNaves(Nave [] flota) {
96     Nave [] flota1 = new Nave[flota.length];
97     System.arraycopy(flota, 0 , flota1, 0 , flota.length);
98     for(int i = 0; i < flota1.length; i++) {
99         int posCambio1 = (int)(Math.random() * flota1.length);
100        int posCambio2 = (int)(Math.random() * flota1.length);
101        Nave nave = flota1[posCambio2];
102        flota1[posCambio2] = flota1[posCambio1];
103        flota1[posCambio1] = nave;
104    }
105    return flota1;
106 }
107 }
```

	<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>

EJERCICIO 2:

Este es una nueva clase, específicamente la clase Soldado, este tiene los atributos del nombre del soldado y su vida como se vio también anteriormente tiene tanto métodos mutadores como accesores y el toString para imprimir los datos de los soldados

```

7 public class Soldados {
8     // Atributos de la clase Soldados
9     private int vida;
10    private String nombre;
11
12    // Métodos mutadores
13    public void setNombre(String nombre_) {
14        nombre = nombre_;
15    }
16    public void setVida(int vida_) {
17        vida = vida_;
18    }
19
20    // Métodos accesores
21    public String getNombre() {
22        return nombre;
23    }
24    public int getVida() {
25        return vida;
26    }
27
28    // Método toString para mostrar la información del soldado
29    public String toString() {
30        return ("Nombre: " + getNombre() + "\nVida: " + getVida() + "\n");
31    }
32 }



```

Como se puede ver en el main se creó un arreglo de Soldados que lo llevaremos a la función rellenar, acá llenaremos tanto su nombre como su vida usando la clase Soldados para ello y después es llevado al método imprimir que como dice, imprimirá todos los datos de los soldados guardados en el arreglo.

```

9 public class ejercicio2 {
10    public static void main(String[] args) {
11        // Se crea un arreglo de 5 soldados
12        Soldados[] soldados = new Soldados[5];
13        rellenarLista(soldados); // Llenar el arreglo con datos
14        imprimirLista(soldados); // Imprimir la lista de soldados
15    }
16 }

```

	<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>

En primer método simplemente metemos el arreglo en un bucle hasta llenarlo, y llamaremos a los métodos de la clase Soldado que como se mencionó indican la vida y el nombre del soldado.

El segundo método simplemente imprime el arreglo a través de un for-each.

```

5
7 // Método para rellenar la lista de soldados con sus datos
8 public static void rellenarLista(Soldados[] lista) {
9     Scanner sc = new Scanner(System.in);
10    for(int i = 0; i < lista.length; i++) {
11        lista[i] = new Soldados();
12        System.out.print("Ingrese su nombre: ");
13        lista[i].setNombre(sc.next());
14        System.out.print("Ingrese el nivel de vida: ");
15        lista[i].setVida(sc.nextInt());
16    }
17 }
18
19 // Método para imprimir la lista de soldados
20 public static void imprimirLista(Soldados[] lista) {
21     for(Soldados n : lista)
22         System.out.println(n);
23 }
24 }

```



EJERCICIO 3:

Primeramente, declaramos las variables en la clase Ejercito, después creamos un constructor que recibe la cantidad de soldados que debe contener,

```

1 package Laboratorio3;
2 /*Laboratorio Nr3 - Ejercicio3
7 import java.util.*;
8
9 public class Ejercito {
10     private String[] ejercito;
11     private String nombreEjercito;
12
13     // Constructor que inicializa el ejército con un número determinado de soldados
14     Ejercito(int soldados) {
15         ejercito = new String[soldados];
16     }
17 }

```

	<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: 7</p>

El primer método se encarga de rellenar nuestro arreglo de "String" como sabemos estos llevan "null" por defecto es por ello que los rellenamos con soldado y su posición.

El segundo devuelve la cantidad de espacios que tiene el arreglo o sea la cantidad de soldados.

```

18 // Método para llenar el ejército con soldados numerados
19 public void completarEjercito() {
20     for(int i = 0; i < ejercito.length; i++)
21         ejercito[i] = "soldado" + i;
22 }
23
24 // Método que devuelve la cantidad de soldados en el ejército
25 public int cantidadSoldados() {
26     return ejercito.length;
27 }
28

```



El primer método recibe el nombre de cada ejército de la clase Ejercito que se haya creado y lo guardamos, además de ello creamos un "accesor" para poder llamarlo cuando queramos.

Después se creo un método para imprimir todo el arreglo, en este caso es más eficiente que el toString.

```

29 // Método para asignar un nombre al ejército
30 public void setNombreEjercito() {
31     Scanner sc = new Scanner(System.in);
32     System.out.print("Dele un nombre al ejército: ");
33     nombreEjercito = sc.nextLine();
34 }
35
36 // Método para obtener el nombre del ejército
37 public String getNombreEjercito() {
38     return nombreEjercito;
39 }
40
41 // Método para mostrar los soldados del ejército
42 public void mostrarSoldados() {
43     System.out.println("\n" + getNombreEjercito());
44     for(String n : ejercito) {
45         System.out.print(n + "\n");
46     }
47 }
48 }

```


	<p style="text-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: 8

Primero en el main creamos ejercito1 y ejercito2 de la clase Ejercito y por medio del constructor que vimos le mandamos un numero al azar entre 1 y 5.

Después llamamos los métodos de la clase Ejercito, que el primero se encarga de darle un nombre al ejército, después el método completarEjercito como ya lo vimos anteriormente no hay que profundizar que al igual que mostrarSoldados.

```

7 public class ejercicio3 {
8     public static void main(String[] args) {
9         // Se crean dos ejércitos con un número aleatorio de soldados
10        Ejercito ejercito1 = new Ejercito((int)((Math.random() * 5) + 1));
11        Ejercito ejercito2 = new Ejercito((int)((Math.random() * 5) + 1));
12
13        // Se asignan nombres y completan las listas de soldados
14        ejercito1.setNombreEjercito();
15        ejercito2.setNombreEjercito();
16
17        ejercito1.completarEjercito();
18        ejercito2.completarEjercito();
19
20        // Se muestran los soldados de cada ejército
21        ejercito1.mostrarSoldados();
22        ejercito2.mostrarSoldados();
23    }



```

Este último método del ejercicio3 simplemente define al ganador con condicionales múltiples, lo que define al ganador es el que tenga más soldados en el arreglo.

```

24        // Se determina el ejército ganador
25        mostrarEjercitoGanador(ejercito1, ejercito2);
26    }
27
28    // Método que muestra el ejército con más soldados
29    public static void mostrarEjercitoGanador(Ejercito ejercito1, Ejercito ejercito2) {
30        System.out.println("\nEl ejército ganador es:");
31        if(ejercito1.cantidadSoldados() > ejercito2.cantidadSoldados())
32            ejercito1.mostrarSoldados();
33        else if(ejercito1.cantidadSoldados() < ejercito2.cantidadSoldados())
34            ejercito2.mostrarSoldados();
35        else
36            System.out.println("Es un empate");
37    }
38 }

```


	<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: 9</p>

COMMITTS:

Estos vendrían a ser todos los “commit” que hice a mi repositorio personal.

```

emanu@DESKTOP-LT66QVT MINGW64 /e/LABORATORIO FP2/REPOSITORIO_FP2_LAB/Laboratorio
3 (Laboratorio3)
$ git log
commit c6529c8f4f3cf34a76cc70a450a289b6eddfb9e9 (HEAD -> Laboratorio3, origin/La
boratorio3)
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 20:15:03 2024 -0500

    Correccion de la version final

commit 5fff48a677d056d4d44d144b65691bad96955152
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 16:46:44 2024 -0500

    Version final del ejercicio

commit b87e401ef0a319bbb1b2dc99e25e95c77633fa15
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 16:27:22 2024 -0500

    Primer commit de Ejercito

commit 998f2cc4dd9953f3964773090ec5d27d9222e56e
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 16:26:39 2024 -0500

...skipping...
commit c6529c8f4f3cf34a76cc70a450a289b6eddfb9e9 (HEAD -> Laboratorio3, origin/Laboratorio3)
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 20:15:03 2024 -0500

    Correccion de la version final

commit 5fff48a677d056d4d44d144b65691bad96955152
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 16:46:44 2024 -0500

    Version final del ejercicio

commit b87e401ef0a319bbb1b2dc99e25e95c77633fa15
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 16:27:22 2024 -0500

    Primer commit de Ejercito

commit 998f2cc4dd9953f3964773090ec5d27d9222e56e
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 16:26:39 2024 -0500

    Primer commit del ejercicio 3

commit a89e332c4b6c5afa8bf481cf860592e7b4462d45
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 13:09:15 2024 -0500

```

```
commit a89e332c4b6c5afa8bf481cf860592e7b4462d45
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 13:09:15 2024 -0500

Segundo avance del codigo

commit 27ba8895865d948e1dadcd2579a90d9ba5b62cc1b
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 13:08:47 2024 -0500

Segundo avance del codigo

commit df25b0d09ed2c671830ab238668376fe25b38d40
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 13:01:33 2024 -0500

Primr commit de Soldados

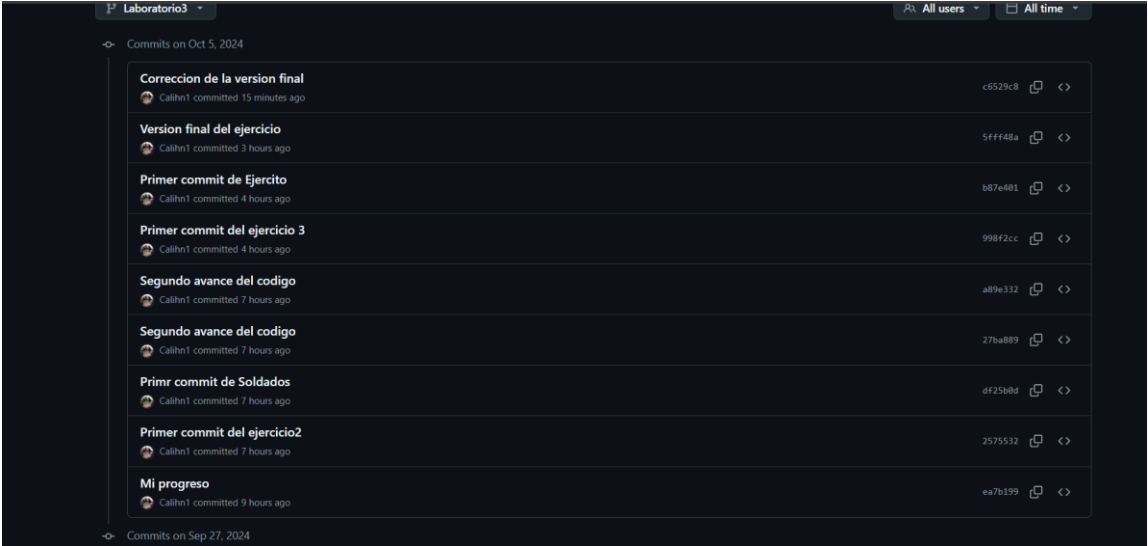
commit 2575532552e68389a333fcb8d1f5dd102c20fb4c
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 13:01:07 2024 -0500



Primer commit del ejercicio2

commit ea7b1997b25b6c563a3f8f4cc7f29c42fa1a3a4e
Author: Calihn1 <ehilacondob@unsa.edu.pe>
Date: Sat Oct 5 10:42:41 2024 -0500

Mi progreso
```

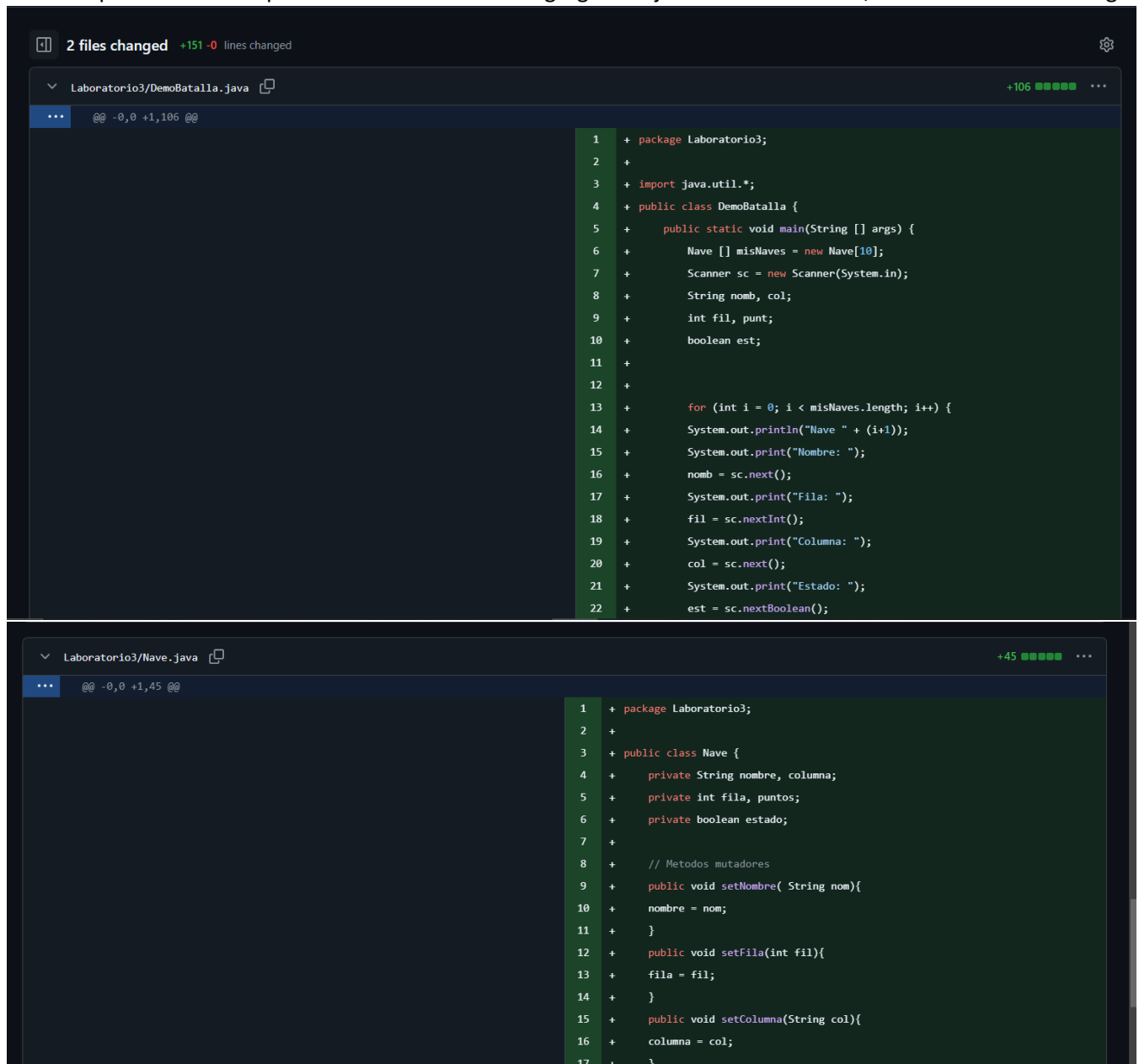
GITHUB: Todos estos “commits” fueron agregados en la rama Laboratorio3



	<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: 11</p>

EXPLICACIÓN DE CADA COMMIT:

Como se puede ver en mi primer commit solamente agregue el ejercicio1 con su clase, aun tenía fallas mi código.



```



1  + package Laboratorio3;
2  +
3  + import java.util.*;
4  + public class DemoBatalla {
5  +     public static void main(String [] args) {
6  +         Nave [] misNaves = new Nave[10];
7  +         Scanner sc = new Scanner(System.in);
8  +         String nomb, col;
9  +         int fil, punt;
10 +         boolean est;
11 +
12 +
13 +         for (int i = 0; i < misNaves.length; i++) {
14 +             System.out.println("Nave " + (i+1));
15 +             System.out.print("Nombre: ");
16 +             nomb = sc.next();
17 +             System.out.print("Fila: ");
18 +             fil = sc.nextInt();
19 +             System.out.print("Columna: ");
20 +             col = sc.next();
21 +             System.out.print("Estado: ");
22 +             est = sc.nextBoolean();

```

```

1  + package Laboratorio3;
2  +
3  + public class Nave {
4  +     private String nombre, columna;
5  +     private int fila, puntos;
6  +     private boolean estado;
7  +
8  +     // Metodos mutadores
9  +     public void setNombre( String nom){
10 +         nombre = nom;
11 +     }
12 +     public void setFila(int fil){
13 +         fila = fil;
14 +     }
15 +     public void setColumna(String col){
16 +         columna = col;
17 +     }

```

	<p style="text-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: 12

En estos segundos “commits” solo puse la clase y el ejercicio2 por ser el avance que logré hasta terminar el laboratorio del martes.

```

Laboratorio3/ejercicio2.java
... @@ -0,0 +1,5 @@

1 + package Laboratorio3;
2 +
3 + public class ejercicio2 {
4 +
5 + }

```

```

1 file changed +5 -0 lines changed

Laboratorio3/Soldados.java
... @@ -0,0 +1,5 @@

1 + package Laboratorio3;
2 +
3 + public class Soldados {
4 +
5 + }

```



En los siguientes commit termine el código de los demás ejercicios, pero todavía faltaba agregar comentarlos e identificarlos de mejor manera.

```



1 file changed +20 -1 lines changed

Laboratorio3/ejercicio2.java
... @@ -1,5 +1,24 @@

1 package Laboratorio3;
2
3 public class ejercicio2 {
4 -
5 +
6 +
7 +
8 +
9 +
10 +
11 +
12 +
13 +
14 +
15 +
16 +
17 +
18 +
19 +
20 +
21 +
22 +
23 +
24 +
25 +
26 +
27 +
28 +
29 +
30 +
31 +
32 +
33 +
34 +
35 +
36 +
37 +
38 +
39 +
40 +
41 +
42 +
43 +
44 +
45 +
46 +
47 +
48 +
49 +
50 +
51 +
52 +
53 +
54 +
55 +
56 +
57 +
58 +
59 +
60 +
61 +
62 +
63 +
64 +
65 +
66 +
67 +
68 +
69 +
70 +
71 +
72 +
73 +
74 +
75 +
76 +
77 +
78 +
79 +
80 +
81 +
82 +
83 +
84 +
85 +
86 +
87 +
88 +
89 +
90 +
91 +
92 +
93 +
94 +
95 +
96 +
97 +
98 +
99 +
100 +
101 +
102 +
103 +
104 +
105 +
106 +
107 +
108 +
109 +
110 +
111 +
112 +
113 +
114 +
115 +
116 +
117 +
118 +
119 +
120 +
121 +
122 +
123 +
124 +
125 +
126 +
127 +
128 +
129 +
130 +
131 +
132 +
133 +
134 +
135 +
136 +
137 +
138 +
139 +
140 +
141 +
142 +
143 +
144 +
145 +
146 +
147 +
148 +
149 +
150 +
151 +
152 +
153 +
154 +
155 +
156 +
157 +
158 +
159 +
160 +
161 +
162 +
163 +
164 +
165 +
166 +
167 +
168 +
169 +
170 +
171 +
172 +
173 +
174 +
175 +
176 +
177 +
178 +
179 +
180 +
181 +
182 +
183 +
184 +
185 +
186 +
187 +
188 +
189 +
190 +
191 +
192 +
193 +
194 +
195 +
196 +
197 +
198 +
199 +
200 +
201 +
202 +
203 +
204 +
205 +
206 +
207 +
208 +
209 +
210 +
211 +
212 +
213 +
214 +
215 +
216 +
217 +
218 +
219 +
220 +
221 +
222 +
223 +
224 +
225 +
226 +
227 +
228 +
229 +
230 +
231 +
232 +
233 +
234 +
235 +
236 +
237 +
238 +
239 +
240 +
241 +
242 +
243 +
244 +
245 +
246 +
247 +
248 +
249 +
250 +
251 +
252 +
253 +
254 +
255 +
256 +
257 +
258 +
259 +
260 +
261 +
262 +
263 +
264 +
265 +
266 +
267 +
268 +
269 +
270 +
271 +
272 +
273 +
274 +
275 +
276 +
277 +
278 +
279 +
280 +
281 +
282 +
283 +
284 +
285 +
286 +
287 +
288 +
289 +
290 +
291 +
292 +
293 +
294 +
295 +
296 +
297 +
298 +
299 +
300 +
301 +
302 +
303 +
304 +
305 +
306 +
307 +
308 +
309 +
310 +
311 +
312 +
313 +
314 +
315 +
316 +
317 +
318 +
319 +
320 +
321 +
322 +
323 +
324 +
325 +
326 +
327 +
328 +
329 +
330 +
331 +
332 +
333 +
334 +
335 +
336 +
337 +
338 +
339 +
340 +
341 +
342 +
343 +
344 +
345 +
346 +
347 +
348 +
349 +
350 +
351 +
352 +
353 +
354 +
355 +
356 +
357 +
358 +
359 +
360 +
361 +
362 +
363 +
364 +
365 +
366 +
367 +
368 +
369 +
370 +
371 +
372 +
373 +
374 +
375 +
376 +
377 +
378 +
379 +
380 +
381 +
382 +
383 +
384 +
385 +
386 +
387 +
388 +
389 +
390 +
391 +
392 +
393 +
394 +
395 +
396 +
397 +
398 +
399 +
400 +
401 +
402 +
403 +
404 +
405 +
406 +
407 +
408 +
409 +
410 +
411 +
412 +
413 +
414 +
415 +
416 +
417 +
418 +
419 +
420 +
421 +
422 +
423 +
424 +
425 +
426 +
427 +
428 +
429 +
430 +
431 +
432 +
433 +
434 +
435 +
436 +
437 +
438 +
439 +
440 +
441 +
442 +
443 +
444 +
445 +
446 +
447 +
448 +
449 +
450 +
451 +
452 +
453 +
454 +
455 +
456 +
457 +
458 +
459 +
460 +
461 +
462 +
463 +
464 +
465 +
466 +
467 +
468 +
469 +
470 +
471 +
472 +
473 +
474 +
475 +
476 +
477 +
478 +
479 +
480 +
481 +
482 +
483 +
484 +
485 +
486 +
487 +
488 +
489 +
490 +
491 +
492 +
493 +
494 +
495 +
496 +
497 +
498 +
499 +
500 +
501 +
502 +
503 +
504 +
505 +
506 +
507 +
508 +
509 +
510 +
511 +
512 +
513 +
514 +
515 +
516 +
517 +
518 +
519 +
520 +
521 +
522 +
523 +
524 +
525 +
526 +
527 +
528 +
529 +
530 +
531 +
532 +
533 +
534 +
535 +
536 +
537 +
538 +
539 +
540 +
541 +
542 +
543 +
544 +
545 +
546 +
547 +
548 +
549 +
550 +
551 +
552 +
553 +
554 +
555 +
556 +
557 +
558 +
559 +
560 +
561 +
562 +
563 +
564 +
565 +
566 +
567 +
568 +
569 +
570 +
571 +
572 +
573 +
574 +
575 +
576 +
577 +
578 +
579 +
580 +
581 +
582 +
583 +
584 +
585 +
586 +
587 +
588 +
589 +
590 +
591 +
592 +
593 +
594 +
595 +
596 +
597 +
598 +
599 +
600 +
601 +
602 +
603 +
604 +
605 +
606 +
607 +
608 +
609 +
610 +
611 +
612 +
613 +
614 +
615 +
616 +
617 +
618 +
619 +
620 +
621 +
622 +
623 +
624 +
625 +
626 +
627 +
628 +
629 +
630 +
631 +
632 +
633 +
634 +
635 +
636 +
637 +
638 +
639 +
640 +
641 +
642 +
643 +
644 +
645 +
646 +
647 +
648 +
649 +
650 +
651 +
652 +
653 +
654 +
655 +
656 +
657 +
658 +
659 +
660 +
661 +
662 +
663 +
664 +
665 +
666 +
667 +
668 +
669 +
670 +
671 +
672 +
673 +
674 +
675 +
676 +
677 +
678 +
679 +
680 +
681 +
682 +
683 +
684 +
685 +
686 +
687 +
688 +
689 +
690 +
691 +
692 +
693 +
694 +
695 +
696 +
697 +
698 +
699 +
700 +
701 +
702 +
703 +
704 +
705 +
706 +
707 +
708 +
709 +
710 +
711 +
712 +
713 +
714 +
715 +
716 +
717 +
718 +
719 +
720 +
721 +
722 +
723 +
724 +
725 +
726 +
727 +
728 +
729 +
730 +
731 +
732 +
733 +
734 +
735 +
736 +
737 +
738 +
739 +
740 +
741 +
742 +
743 +
744 +
745 +
746 +
747 +
748 +
749 +
750 +
751 +
752 +
753 +
754 +
755 +
756 +
757 +
758 +
759 +
760 +
761 +
762 +
763 +
764 +
765 +
766 +
767 +
768 +
769 +
770 +
771 +
772 +
773 +
774 +
775 +
776 +
777 +
778 +
779 +
780 +
781 +
782 +
783 +
784 +
785 +
786 +
787 +
788 +
789 +
790 +
791 +
792 +
793 +
794 +
795 +
796 +
797 +
798 +
799 +
800 +
801 +
802 +
803 +
804 +
805 +
806 +
807 +
808 +
809 +
810 +
811 +
812 +
813 +
814 +
815 +
816 +
817 +
818 +
819 +
820 +
821 +
822 +
823 +
824 +
825 +
826 +
827 +
828 +
829 +
830 +
831 +
832 +
833 +
834 +
835 +
836 +
837 +
838 +
839 +
840 +
841 +
842 +
843 +
844 +
845 +
846 +
847 +
848 +
849 +
850 +
851 +
852 +
853 +
854 +
855 +
856 +
857 +
858 +
859 +
860 +
861 +
862 +
863 +
864 +
865 +
866 +
867 +
868 +
869 +
870 +
871 +
872 +
873 +
874 +
875 +
876 +
877 +
878 +
879 +
880 +
881 +
882 +
883 +
884 +
885 +
886 +
887 +
888 +
889 +
890 +
891 +
892 +
893 +
894 +
895 +
896 +
897 +
898 +
899 +
900 +
901 +
902 +
903 +
904 +
905 +
906 +
907 +
908 +
909 +
910 +
911 +
912 +
913 +
914 +
915 +
916 +
917 +
918 +
919 +
920 +
921 +
922 +
923 +
924 +
925 +
926 +
927 +
928 +
929 +
930 +
931 +
932 +
933 +
934 +
935 +
936 +
937 +
938 +
939 +
940 +
941 +
942 +
943 +
944 +
945 +
946 +
947 +
948 +
949 +
950 +
951 +
952 +
953 +
954 +
955 +
956 +
957 +
958 +
959 +
960 +
961 +
962 +
963 +
964 +
965 +
966 +
967 +
968 +
969 +
970 +
971 +
972 +
973 +
974 +
975 +
976 +
977 +
978 +
979 +
980 +
981 +
982 +
983 +
984 +
985 +
986 +
987 +
988 +
989 +
990 +
991 +
992 +
993 +
994 +
995 +
996 +
997 +
998 +
999 +
1000 +
1001 +
1002 +
1003 +
1004 +
1005 +
1006 +
1007 +
1008 +
1009 +
1010 +
1011 +
1012 +
1013 +
1014 +
1015 +
1016 +
1017 +
1018 +
1019 +
1020 +
1021 +
1022 +
1023 +
1024 +
1025 +
1026 +
1027 +
1028 +
1029 +
1030 +
1031 +
1032 +
1033 +
1034 +
1035 +
1036 +
1037 +
1038 +
1039 +
1040 +
1041 +
1042 +
1043 +
1044 +
1045 +
1046 +
1047 +
1048 +
1049 +
1050 +
1051 +
1052 +
1053 +
1054 +
1055 +
1056 +
1057 +
1058 +
1059 +
1060 +
1061 +
1062 +
1063 +
1064 +
1065 +
1066 +
1067 +
1068 +
1069 +
1070 +
1071 +
1072 +
1073 +
1074 +
1075 +
1076 +
1077 +
1078 +
1079 +
1080 +
1081 +
1082 +
1083 +
1084 +
1085 +
1086 +
1087 +
1088 +
1089 +
1090 +
1091 +
1092 +
1093 +
1094 +
1095 +
1096 +
1097 +
1098 +
1099 +
1100 +
1101 +
1102 +
1103 +
1104 +
1105 +
1106 +
1107 +
1108 +
1109 +
1110 +
1111 +
1112 +
1113 +
1114 +
1115 +
1116 +
1117 +
1118 +
1119 +
1120 +
1121 +
1122 +
1123 +
1124 +
1125 +
1126 +
1127 +
1128 +
1129 +
1130 +
1131 +
1132 +
1133 +
1134 +
1135 +
1136 +
1137 +
1138 +
1139 +
1140 +
1141 +
1142 +
1143 +
1144 +
1145 +
1146 +
1147 +
1148 +
1149 +
1150 +
1151 +
1152 +
1153 +
1154 +
1155 +
1156 +
1157 +
1158 +
1159 +
1160 +
1161 +
1162 +
1163 +
1164 +
1165 +
1166 +
1167 +
1168 +
1169 +
1170 +
1171 +
1172 +
1173 +
1174 +
1175 +
1176 +
1177 +
1178 +
1179 +
1180 +
1181 +
1182 +
1183 +
1184 +
1185 +
1186 +
1187 +
1188 +
1189 +
1190 +
1191 +
1192 +
1193 +
1194 +
1195 +
1196 +
1197 +
1198 +
1199 +
1200 +
1201 +
1202 +
1203 +
1204 +
1205 +
1206 +
1207 +
1208 +
1209 +
1210 +
1211 +
1212 +
1213 +
1214 +
1215 +
1216 +
1217 +
1218 +
1219 +
1220 +
1221 +
1222 +
1223 +
1224 +
1225 +
1226 +
1227 +
1228 +
1229 +
1230 +
1231 +
1232 +
1233 +
1234 +
1235 +
1236 +
1237 +
1238 +
1239 +
1240 +
1241 +
1242 +
1243 +
1244 +
1245 +
1246 +
1247 +
1248 +
1249 +
1250 +
1251 +
1252 +
1253 +
1254 +
1255 +
1256 +
1257 +
1258 +
1259 +
1260 +
1261 +
1262 +
1263 +
1264 +
1265 +
1266 +
1267 +
1268 +
1269 +
1270 +
1271 +
1272 +
1273 +
1274 +
1275 +
1276 +
1277 +
1278 +
1279 +
1280 +
1281 +
1282 +
1283 +
1284 +
1285 +
1286 +
1287 +
1288 +
1289 +
1290 +
1291 +
1292 +
1293 +
1294 +
1295 +
1296 +
1297 +
1298 +
1299 +
1300 +
1301 +
1302 +
1303 +
1304 +
1305 +
1306 +
1307 +
1308 +
1309 +
1310 +
1311 +
1312 +
1313 +
1314 +
1315 +
1316 +
1317 +
1318 +
1319 +
1320 +
1321 +
1322 +
1323 +
1324 +
1325 +
1326 +
1327 +
1328 +
1329 +
1330 +
1331 +
1332 +
1333 +
1334 +
1335 +
1336 +
1337 +
1338 +
1339 +
1340 +
1341 +
1342 +
1343 +
1344 +
1345 +
1346 +
1347 +
1348 +
1349 +
1350 +
1351 +
1352 +
1353 +
1354 +
1355 +
1356 +
1357 +
1358 +
1359 +
1360 +
1361 +
1362 +
1363 +
1364 +
1365 +
1366 +
1367 +
1368 +
1369 +
1370 +
1371 +
1372 +
1373 +
1374 +
1375 +
1376 +
1377 +
1378 +
1379 +
1380 +
1381 +
1382 +
1383 +
1384 +
1385 +
1386 +
1387 +
1388 +
1389 +
1390 +
1391 +
1392 +
1393 +
1394 +
1395 +
1396 +
1397 +
1398 +
1399 +
1400 +
1401 +
1402 +
1403 +
1404 +
1405 +
1406 +
1407 +
1408 +
1409 +
1410 +
1411 +
1412 +
1413 +
1414 +
1415 +
1416 +
1417 +
1418 +
1419 +
1420 +
1421 +
1422 +
1423 +
1424 +
1425 +
1426 +
1427 +
1428 +
1429 +
1430 +
1431 +
1432 +
1433 +
1434 +
1435 +
1436 +
1437 +
1438 +
1439 +
1440 +
1441 +
1442 +
1443 +
1444 +
1445 +
1446 +
1447 +
1448 +
1449 +
1450 +
1451 +
1452 +
1453 +
1454 +
1455 +
1456 +
1457 +
1458 +
1459 +
1460 +
1461 +
1462 +
1463 +
1464 +
1465 +
1466 +
1467 +
1468 +
1469 +
1470 +
1471 +
1472 +
1473 +
1474 +
1475 +
1476 +
1477 +
1478 +
1479 +
1480 +
1481 +
1482 +
1483 +
1484 +
1485 +
1486 +
1487 +
1488 +
1489 +
1490 +
1491 +
1492 +
1493 +
1494 +
1495 +
1496 +
1497 +
1498 +
1499 +
1500 +
1501 +
1502 +
1503 +
1504 +
1505 +
1506 +
1507 +
1508 +
1509 +
1510 +
1511 +
1512 +
1513 +
1514 +
1515 +
1516 +
1517 +
1518 +
1519 +
1520 +
1521 +
1522 +
1523 +
1524 +
1525 +
1526 +
1527 +
1528 +
1529 +
1530 +
1531 +
1532 +
1533 +
1534 +
1535 +
1536 +
1537 +
1538 +
1539 +
1540 +
1541 +
1542 +
1543 +
1544 +
1545 +
1546 +
1547 +
1548 +
1549 +
1550 +
1551 +
1552 +
1553 +
1554 +
1555 +
1556 +
1557 +
1558 +
1559 +
1560 +
1561 +
1562 +
1563 +
1564 +
1565 +
1566 +
1567 +
1568 +
1569 +
1570 +
1571 +
1572 +
1573 +
1574 +
1575 +
1576 +
1577 +
1578 +
1579 +
1580 +
1581 +
1582 +
1583 +
1584 +
1585 +
1586 +
1587 +
1588 +
1589 +
1590 +
1591 +
1592 +
1593 +
1594 +
1595 +
1596 +
1597 +
1598 +
1599 +
1600 +
1601 +
1602 +
1603 +
1604 +
1605 +
1606 +
1607 +
1608 +
1609 +
1610 +
1611 +
1612 +
1613 +
1614 +
1615 +
1616 +
1617 +
1618 +
1619 +
1620 +
1621 +
1622 +
1623 +
1624 +
1625 +
1626 +
1627 +
1628 +
1629 +
1630 +
1631 +
1632 +
1633 +
1634 +
1635 +
1636 +
1637 +
1638 +
1639 +
1640 +
1641 +
1642 +
1643 +
1644 +
1645 +
1646 +
1647 +
1648 +
1649 +
1650 +
1651 +
1652 +
1653 +
1654 +
1655 +
1656 +
1657 +
1658 +
1659 +
1660 +
1661 +
1662 +
1663 +
1664 +
1665 +
1666 +
1667 +
1668 +
1669 +
1670 +
1671 +
1672 +
1673 +
1674 +
1675 +
1676 +
1677 +
1678 +
1679 +
1680 +
1681 +
1682 +
1683 +
1684 +
1685 +
1686 +
1687 +
1688 +
1689 +
1690 +
1691 +
1692 +
1693 +
1694 +
1695 +
1696 +
1697 +
1698 +
1699 +
1700 +
1701 +
1702 +
1703 +
1704 +
1705 +
1706 +
1707 +
1708 +
1709 +
1710 +
1711 +
1712 +
1713 +
1714 +
1715 +
1716 +
1717 +
1718 +
1719 +
1720 +
1721 +
1722 +
1723 +
1724 +
1725 +
1726 +
1727 +
1728 +
1729 +
1730 +
1731 +
1732 +
1733 +
1734 +
1735 +
1736 +
1737 +
1738 +
1739 +
1740 +
1741 +
1742 +
1743 +
1744 +
1745 +
1746 +
1747 +
1748 +
1749 +
1750 +
1751 +
1752 +
1753 +
1754 +
1755 +
1756 +
1757 +
1758 +
1759 +
1760 +
1761 +
1762 +
1763 +
1764 +
1765 +
1766 +
1767 +
1768 +
1769 +
1770 +
1771 +
1772 +
1773 +
1774 +
1775 +
1776 +
1777 +
1778 +
1779 +
1780 +
1781 +
1782 +
1783 +
1784 +
1785 +
1786 +
1787 +
1788 +
1789 +
1790 +
1791 +
1792 +
1793 +
1794 +
1795 +
1796 +
1797 +
1798 +
1799 +
1800 +
1801 +
1802 +
1803 +
1804 +
1805 +
1806 +
1807 +
1808 +
1809 +
1810 +
1811 +
1812 +
1813 +
1814 +
1815 +
1816 +
1817 +
1818 +
1819 +
1820 +
1821 +
1822 +
1823 +
1824 +
1825 +
1826 +
1827 +
1828 +
1829 +
1830 +
1831 +
1832 +
1833 +
1834 +
1835 +
1836 +
1837 +
1838 +
1839 +
1840 +
1841 +
1842 +
1843 +
1844 +
1845 +
1846 +
1847 +
1848 +
1849 +
1850 +
1851 +
1852 +
1853 +
1854 +
1855 +
1856 +
1857 +
1858 +
1859 +
1860 +
1861 +
1862 +
1863 +
1864 +
1865 +
1866 +
1867 +
1868 +
1869 +
1870 +
1871 +
1872 +
1873 +
1874 +
1875 +
1876 +
1877 +
1878 +
1879 +
1880 +
1881 +
1882 +
1883 +
1884 +
1885 +
1886 +
1887 +
1888 +
1889 +
1890 +
1891 +
1892 +
1893 +
1894 +
1895 +
1896 +
1897 +
1898 +
1899 +
1900 +
1901 +
1902 +
1903 +
1904 +
1905 +
1906 +
1907 +
1908 +
1909 +
1910 +
1911 +
1912 +
1913 +
1914 +
1915 +
1916 +
1917 +
1918 +
1919 +
1920 +
1921 +
1922 +
1923 +
1924 +
1925 +
1926 +
1927 +
1928 +
1929 +
1930 +
1931 +
1932 +
1933 +
1934 +
1935 +
1936 +
1937 +
1938 +
1939 +
1940 +
1941 +
1942 +
1943 +
1944 +
1945 +
1946 +
1947 +
1948 +
1949 +
1950 +
1951 +
1952 +
1953 +
1954 +
1955 +
1956 +
1957 +
1958 +
1959 +
1960 +
1961 +
1962 +
1963 +
1964 +
1965 +
1966 +
1967 +
1968 +
1969 +
1970 +
1971 +
1972 +
1973 +
1974 +
1975 +
1976 +
1977 +
1978 +
1979 +
1980 +
1981 +
1982 +
1983 +
1984 +
1985 +
1986 +
1987 +
1988 +
1989 +
1990 +
1991 +
1992 +
1993 +
1994 +
1995 +
1996 +
1997 +
1998 +
1999 +
2000 +
2001 +
2002 +
2003 +
2004 +
2005 +
2006 +
2007 +
2008 +
2009 +
2010 +
2011 +
2012 +
2013 +
2014 +
2015 +
2016 +
2017 +
2018 +
2019 +
2020 +
2021 +
2022 +
2023 +
2024 +
2025 +
2026 +
2027 +
2028 +
2029 +
2030 +
2031 +
2032 +
2033 +
2034 +
2035 +
2036 +
2037 +
2038 +
2039 +
2040 +
2041 +
2042 +
2043 +
2044 +
2045 +
2046 +
2047 +
2048 +
2049 +
2050 +
2051 +
2052 +
2053 +
2054 +
2055 +
2056 +
2057 +
2058 +
2059 +
2060 +
2061 +
2062 +
2063 +
2064 +
2065 +
2066 +
2067 +
2068 +
2069 +
2070 +
2071 +
2072 +
2073 +
2074 +
2075 +
2076 +
2077 +
2078 +
2079 +
2080 +
2081 +
2082 +
2083 +
2084 +
2085 +
2086 +
2087 +
2088 +
2089 +
2090 +
2091 +
2092 +
2093 +
2094 +
2095 +
2096 +
2097 +
2098 +
2099 +
2100 +
2101 +
2102 +
2103 +
2104 +
2105 +
2106 +
2107 +
2108 +
2109 +
```

	<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>

<pre> ... @@ -1,5 +1,23 @@ 1 package Laboratorio3; 2 3 public class Soldados { </pre>	<pre> 1 package Laboratorio3; 2 3 public class Soldados { 4 + private int vida; 5 + private String nombre; 6 + 7 + public void setNombre(String nombre_) { 8 + nombre=nombre_; 9 + } 10 + public void setVida(int vida_) { 11 + vida=vida_; 12 + } 13 + public String getNombre() { 14 + return nombre; 15 + } 16 + public int getVida() { 17 + return vida; 18 + } 19 + public String toString() { 20 + return ("Nombre: "+getNombre()+"\nVida: "+getVida()+"\n"); 21 + } 22 </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: 14</p>

```

Laboratorio3/ejercicio3.java
... @@ -0,0 +1,29 @@

1 + package Laboratorio3;
2 +
3 + public class ejercicio3 {
4 +     public static void main(String[] args){
5 +         Ejercito ejercito1=new Ejercito( (int)((Math.random()*5)+1) );
6 +         Ejercito ejercito2=new Ejercito( (int)((Math.random()*5)+1) );
7 +         ejercito1.setNombreEjercito();
8 +         ejercito2.setNombreEjercito();
9 +
10 +         ejercito1.completarEjercito();
11 +         ejercito2.completarEjercito();
12 +
13 +         ejercito1.mostrarSoldados();
14 +         ejercito2.mostrarSoldados();
15 +
16 +         mostrarEjercitoGanador(ejercito1,ejercito2);
17 +     }
18 +     public static void mostrarEjercitoGanador(Ejercito ejercito1, Ejercito ejercito2) {
19 +         System.out.println("\nEl ejercito ganador es:");
20 +         if(ejercito1.cantidadSoldados()>ejercito2.cantidadSoldados())
21 +             ejercito1.mostrarSoldados();
22 +
23 +         else if(ejercito1.cantidadSoldados()<ejercito2.cantidadSoldados())
24 +             ejercito2.mostrarSoldados();
25 +
26 +         else
27 +             System.out.println("Es un empate");
28 +     }
29 + }


Laboratorio3/Ejercito.java
... @@ -0,0 +1,32 @@



1 + package Laboratorio3;
2 + import java.util.*;
3 +
4 + public class Ejercito {
5 +     private String[] ejercito;
6 +     private String nombreEjercito;
7 +
8 +     Ejercito(int soldados) {
9 +         ejercito=new String[soldados];
10 +     }
11 +     public void completarEjercito() {
12 +         for(int i=0; i<ejercito.length; i++)
13 +             ejercito[i]="soldado"+i;
14 +     }
15 +     public int cantidadSoldados() {
16 +         return(ejercito.length);
17 +     }
18 +     public void setNombreEjercito() {
19 +         Scanner sc=new Scanner(System.in);
20 +         System.out.print("Dele un nombre al ejercito: ");
21 +         nombreEjercito=sc.nextLine();
22 +     }
23 +     public String getNombreEjercito() {
24 +         return nombreEjercito;
25 +     }
26 +     public void mostrarSoldados() {
27 +         System.out.println("\n"+getNombreEjercito());
28 +         for( String n:ejercito) {
29 +             System.out.print(n+"\n");
30 +         }
31 +     }
32 + }

```

En esta parte ingreso la parte final de unos códigos que ya estaban “identados” y con comentarios.

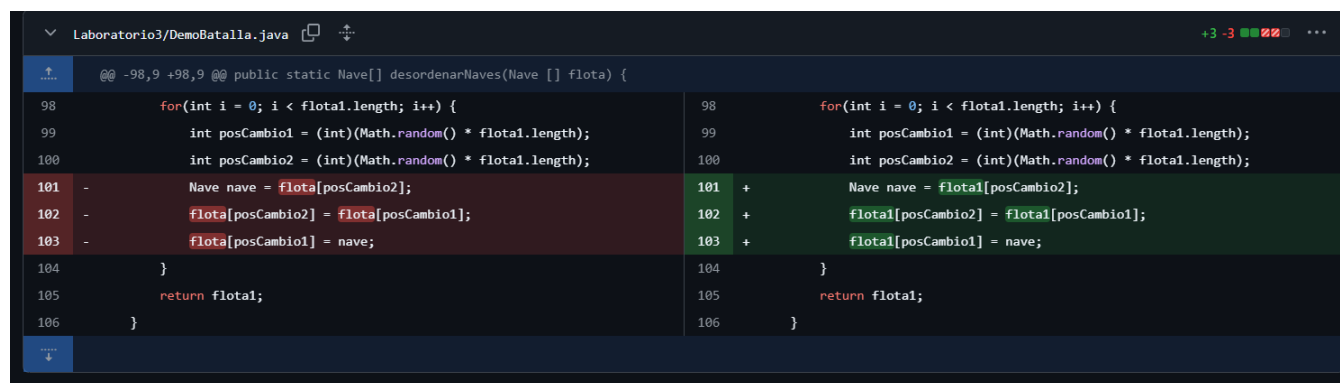
Version final del ejercicio

 Calihn1 committed 5 hours ago

5fff48a  

Este último commit lo utilizo después de testear todos los códigos y corregir un código que tuvo un problema en el desarrollo del algoritmo del ejercicio.

Específicamente esta parte donde en vez de alterar el orden del nuevo arreglo creado, alteraba el orden del arreglo principal



```

Laboratorio3/DemoBatalla.java
@@ -98,9 +98,9 @@ public static Nave[] desordenarNaves(Nave [] flota) {
98     for(int i = 0; i < flota1.length; i++) {
99         int posCambio1 = (int)(Math.random() * flota1.length);
100        int posCambio2 = (int)(Math.random() * flota1.length);
101        - Nave nave = flota[posCambio2];
102        - flota[posCambio2] = flota[posCambio1];
103        - flota[posCambio1] = nave;
104    }
105    return flota1;
106 }



+3 -3
98     for(int i = 0; i < flotal.length; i++) {
99         int posCambio1 = (int)(Math.random() * flotal.length);
100        int posCambio2 = (int)(Math.random() * flotal.length);
101        + Nave nave = flotal[posCambio2];
102        + flotal[posCambio2] = flotal[posCambio1];
103        + flotal[posCambio1] = nave;
104    }
105    return flotal;
106 }
  
```

PRUEBAS:

EJERCICIO 1: Aca como se vio en el código ingresaremos los datos de la nave de la nave 1 a la 10

```

Nave 1
Nombre: A1
Fila: 1
Columna: 2
Estado (true = operativa, false = dañada): true
Puntos: 12
Nave 2
Nombre: B1
Fila: 2
Columna: 3
Estado (true = operativa, false = dañada): TRUE
Puntos: 23
Nave 3
Nombre: A2
Fila: 3
Columna: 4
Estado (true = operativa, false = dañada): false
Puntos: 34
Nave 4
Nombre: B2
Fila: 4
Columna: 5
Estado (true = operativa, false = dañada): false
Puntos: 45
Nave 5
Nombre: A3
Fila: 56
Columna: 6
Estado (true = operativa, false = dañada): true
Puntos: 566
  
```


	<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: 16</p>

Después como también se vio en el código, se imprimió los datos de las 10 naves

Naves creadas:

Nombre: A1

Fila: 1

Columna: 2

Estado: true

Puntos: 12

Nombre: B1

Fila: 2

Columna: 3

Estado: true

Puntos: 23

Nombre: A2

Fila: 3

Columna: 4

Estado: false

Puntos: 34

Nombre: B2

Fila: 4

Columna: 5

Estado: false

Puntos: 45

Nombre: A3

Fila: 56

Columna: 6

Estado: true

Puntos: 566



Nombre: B3

Fila: 6

Columna: 7

Estado: true

Puntos: 67



	<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: 17</p>

Acá se imprime todas las flotas que tengan el nombre de A1

```
Ingrese el nombre de la flota: A1
Nombre: A1
Fila: 1
Columna: 2
Estado: true
Puntos: 12

Nombre: A1
Fila: 34
Columna: 56
Estado: true
```

Se imprime todas las naves que tengan esa cantidad ingresada o menor

	<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: 18</p>

Ingrese un número de puntos: 56

Nombre: A1

Fila: 1

Columna: 2

Estado: true

Puntos: 12

Nombre: B1

Fila: 2

Columna: 3

Estado: true

Puntos: 23

Nombre: A2

Fila: 3

Columna: 4

Estado: false

Puntos: 34

Nombre: B2

Fila: 4

Columna: 5

Estado: false

Puntos: 45

Imprime la nave con mayor cantidad de puntos

Nave con mayor número de puntos:

Nombre: A1



Fila: 34

Columna: 56

Estado: true

Puntos: 567

Aca se imprime en un orden desordenado las naves

	<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: 19</p>

Las naves desordenadas:

Nombre: B1
Fila: 2
Columna: 3
Estado: true
Puntos: 23

Nombre: B2
Fila: 4
Columna: 5
Estado: false
Puntos: 45

Nombre: A1
Fila: 34
Columna: 56
Estado: true
Puntos: 567

Nombre: A3
Fila: 56
Columna: 6
Estado: true
Puntos: 566



Nombre: A1
Fila: 1
Columna: 2
Estado: true
Puntos: 12

Nombre: A4
Fila: 7
Columna: 8
Estado: false
Puntos: 78

EJERCICIO 2:

Ingresar el nombre y la vida de los

```
Ingrese su nombre: Matias
Ingrese el nivel de vida: 23
Ingrese su nombre: Jorge
Ingrese el nivel de vida: 12
Ingrese su nombre: Sergio
Ingrese el nivel de vida: 56
Ingrese su nombre: Fabricio
Ingrese el nivel de vida: 45
Ingrese su nombre: Mejepu
Ingrese el nivel de vida: 23
```

	<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: 20</p>

Imprimir el arreglo de cada soldado sobre sus métodos correspondientes como nombre o vida

```
Nombre: Matias
Vida: 23

Nombre: Jorge
Vida: 12

Nombre: Sergio
Vida: 56

Nombre: Fabricio
Vida: 45

Nombre: Mejepu
Vida: 23
```

EJERCICIO 3:

Ingresamos primeramente el nombre de ambos ejércitos con los métodos respectivamente ya mencionados, después con el Math.random() genera la cantidad de soldados que tendrá cada ejercito y lo imprime y al final imprime el equipo que tenga más soldados.

```
Dele un nombre al ejército: Beba army
Dele un nombre al ejército: Tren de aragua

Beba army
soldado0

Tren de aragua
soldado0
soldado1
soldado2



El ejército ganador es:

Tren de aragua
soldado0
soldado1
soldado2
```

IV. CUESTIONARIO

Cree un Proyecto llamado Laboratorio3 Usted deberá agregar las clases Nave.java y DemoBatalla.java.

Se creo un proyecto con el nombre referido y además de ello también se arreglo las clases Nave.java y DemoBatalla.java

	<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: 21</p>

--

CONCLUSIONES

Con los arreglos de objetos nos podemos permitir agregar muchas más cosas que con un arreglo estándar, y además desarrollar un mejor algoritmo una cantidad reducida de código al poder dividir las funciones en clases lo que facilita el trabajo de main y demás del problema a resolver principal.

METODOLOGÍA DE TRABAJO



Primero entender el problema que plantea el ejercicio.
-Diseñar el algoritmo más corto y eficiente para solucionar el problema que tenían algunos métodos.
-Escribir el código en el lenguaje de programación java para ejecutar el algoritmo.
-Recibir los valores esperados del código realizado.

REFERENCIAS Y BIBLIOGRAFÍA

En este caso no use ninguna referencia para realizar el código.

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

Contenido y demostración		Puntos	Checklis t	Estudiant e	Profeso r
1. GitHub	Hay enlace URL activo del directorio para el laboratorio hacia su repositorio GitHub con código fuente terminado y fácil de revisar.	2	X	2	
2. Commits	Hay capturas de pantalla de los commits más importantes con sus explicaciones detalladas. (El profesor puede preguntar para refrendar calificación).	4	X	3	
3. Código fuente	Hay porciones de código fuente importantes con numeración y explicaciones detalladas de sus funciones.	2	X	2	
4. Ejecución	Se incluyen ejecuciones/pruebas del código fuente explicadas gradualmente.	2	X	2	
5. Pregunta	Se responde con completitud a la pregunta formulada en la tarea. (El profesor puede preguntar para refrendar calificación).	2	X	2	
6. Fechas	Las fechas de modificación del código fuente están dentro de los plazos de fecha de entrega establecidos.	2	X	2	
7. Ortografía	El documento no muestra errores ortográficos.	2	X	2	
8. Madurez	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	XX	2	

	<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: 23</p>

TOTAL	20	17		
-------	----	----	--	--

Tabla 2: Rúbrica para contenido del Informe y demostración