

 UNIVERSIDAD DE SAN BUENAVENTURA	UNIVERSIDAD DE SAN BUENAVENTURA Estructuras de Datos Guía de Aprendizaje No.2 Ordenamiento de listas	Fecha: 12-02 de 2024 Versión 1 Página 1 de 5
--	---	---

ASIGNATURA: Estructuras de Datos.

TEMA: Arreglos (Unidimensionales).

ACTIVIDAD DE ENSEÑANZA – APRENDIZAJE – EVALUACIÓN: Aplicar la programación para la solución de problemas informáticos.

TIEMPO DE LA ACTIVIDAD DE E-A-E: 2 horas

TIEMPO DE LA GUIA DE APRENDIZAJE: 2 horas

ORIENTACIONES GENERALES: Aplicar el concepto de arreglos para dar solución a los siguientes casos.

Arreglos Unidimensionales (Vectores)

Los **arrays**, **arreglos** o **vectores** forman parte de la amplia variedad de estructuras de datos que nos los lenguajes de programación, siendo además una de las principales y más útiles estructuras que podremos tener como herramienta de programación. Por otro lado, son considerados como una variable que contiene múltiples espacios para almacenar múltiples valores. Estos valores pueden ser números, letras o cualquier tipo de variable que deseemos incluso tipos de datos propios

Método String “Length

Esta función se usa para obtener la longitud de una cadena Java. El método de longitud de cadena devuelve la cantidad de caracteres escritos en la Cadena.

Ej.1

```
public class Arreglo_dinamico
{
    int[] numeros = {1,2,3,4,5};

    public void entrada()
    {
        int tam = numeros.length;
        System.out.print("El tamaño es: " + tam);
    }

    public static void main(String arg[ ])
    {
        Arreglo_dinamico ad = new Arreglo_dinamico();
        ad.entrada();
    }
}
```

Ej.2

```
public class Ejemplo_String {  
  
    // declara la cadena como un objeto S1 S2  
    String S1 = "Hola método de cadena Java";  
    String S2 = "RockStar";  
  
    public void informacion()  
    {  
        // El método length () de String devuelve la longitud de una cadena S1.  
  
        int longitud = S1.length();  
        System.out.println("Longitud de una cadena es:" + longitud);  
        // 8 Longitud de una cadena RockStar  
        System.out.println("Longitud de una cadena es:" + S2.length());  
    }  
  
    public static void main(String arg[ ])   
    {  
        Ejemplo_String ej = new Ejemplo_String();  
        ej. informacion();  
  
    }  
}
```

Ej.3

```
public class Lista_num {  
  
    int [] num = {0,1, 2, 3, 4,5,6,7,8,9};  
  
    public void listanum()  
    {  
        for (int i = 0; i < num.length; i++)  
        {  
            System.out.print(num[i]);  
        }  
    }  
  
    public static void main(String arg[ ])   
    {  
        Lista_num ln = new Lista_num();  
        ln. listanum();  
    }  
}
```



UNIVERSIDAD DE SAN BUENAVENTURA
Estructuras de datos
Guía de Aprendizaje No.2
Ordenamiento de listas

Fecha: 12-02 de
2024

Versión 1

Página 3 de 5

Ej.4. Entrada y Salida de datos

#Librería ingreso de datos por teclado

```
import java.util.Scanner;
```

```
public class Leerdatos
```

```
{
```

```
Scanner entrada = new Scanner(System.in);
```

```
int i;
```

```
int[] numeros = new int[5];
```

#Lectura de datos

```
public void leer()
```

```
{
```

```
for(i=0; i<numeros.length; i++)
```

```
{
```

```
System.out.printf("Introduzca número: ", i+1);
```

```
numeros[i] = entrada.nextInt();
```

```
}
```

```
}
```

#Impresión de datos

```
public void imprimir()
```

```
{
```

```
for(i=0; i<numeros.length; i++)
```

```
{
```

```
System.out.println(numeros[i]);
```

```
}
```

```
}
```

```
public static void main(String[] args)
```

```
{
```

```
Leerdatos l = new Leerdatos();
```

```
l.leer();
```

```
l.imprimir ();
```

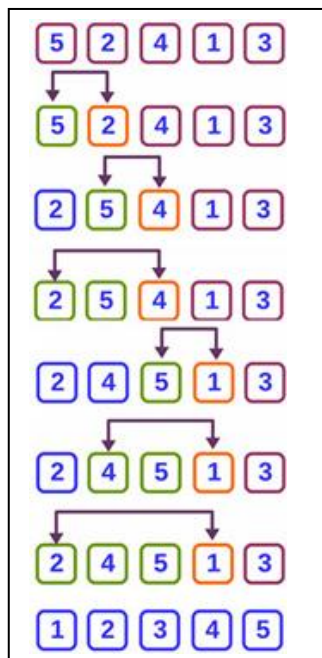
```
}
```

```
}
```

Método de ordenamiento (Burbuja).

La Ordenación de burbuja es un algoritmo de ordenamiento. Funciona revisando cada elemento de la lista que va a ser ordenada con el siguiente, intercambiándolos de posición si están en el orden equivocado.

Estructura



```
int [] lista = {9,1, 5, 4, 7,3};  
int aux = 0;  
  
for (int i = 0; i < lista.length; i++)  
{  
    for (int j = 0; j < lista.length; j++)  
    {  
        if(lista[i] > lista[j])  
        {  
            aux = lista[i];  
            lista[i] = lista[j];  
            lista[j] = aux;  
        }  
    }  
}
```

 <p>UNIVERSIDAD DE SAN BUENAVENTURA</p>	<p align="center">UNIVERSIDAD DE SAN BUENAVENTURA</p> <p align="center">Estructuras de datos</p> <p align="center">Guía de Aprendizaje No.2</p> <p align="center">Ordenamiento de listas</p>	<p>Fecha: 12-02 de 2024</p> <p>Versión 1</p> <p>Página 5 de 5</p>
--	--	---

Actividad

Crear una aplicación en java que permita cargar desde teclados una lista **N** de números. La aplicación deberá mostrar por pantalla un menú de opciones, la primera opción deberá permitir cargar la lista y la segunda opción mostrarla por pantalla. La salida deberá presentar dos listas, una con los números ingresados y la otra lista con los números ordenados de mayor a menor.

Listas

1. Cargar lista
2. Imprimir lista

Elija opción ____