

Ejercicio 1:

a)

```
public class Ejercicio1A {

    public static void main(String[] argumentos) {

        int vector[] = new int[argumentos.length-1];
        boolean ordenAsc = true;
        char letra = argumentos[argumentos.length-1].charAt(0);
        int resultado[] = new int[vector.length];
        if (vector.length > 1) {
            if (Character.isLetter(letra)) {
                if (letra == 'a') {
                    ordenAsc = true;
                } else {
                    ordenAsc = false;
                }

                for (int i = 0 ; i < vector.length ; i++) {
                    vector [i] = Integer.parseInt(argumentos[i]);
                }
                resultado = ordenado(vector, ordenAsc);

                for (int y=0; y < vector.length; y++) {
                    System.out.println(resultado[y]);
                }

            } else {
                System.out.println("El último argumento debe ser un caracter (a = ascendente, d = descendente)");
                System.out.println("El último argumento escrito fue: " + letra);
            }

        } else {
            System.out.println("Los argumentos deben ser al menos dos números y un caracter (a = ascendente, d = descendente)");
        }

    }

    private static int[] ordenado(int[] vector, boolean orden) {
        int valor = 0;
        for (int i = 0; i < vector.length; i++) {
            for (int x = i; x < vector.length; x++) {
                if (orden == true) {
                    if (vector[x] <= vector[i]) {
                        valor = vector[i];
                        vector[i] = vector[x];
                        vector[x] = valor;
                    }
                } else {
                    if (vector[x] >= vector[i]) {
                        valor = vector[i];
                        vector[i] = vector[x];
                        vector[x] = valor;
                    }
                }
            }
        }
        return vector;
    }

}
```

b)

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

```
public class Ejercicio1B {
```

```
    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        boolean ordenAsc = true;
        System.out.println("Ingrese números separados por UN espacio");
        String[] ingreso1 = scn.nextLine().split(" ");
        int numeros[] = new int [ingreso1.length];
        for (int i = 0; i < numeros.length; i++) {
            numeros[i] = Integer.parseInt(ingreso1[i]);
        }
        System.out.println("Ingrese orden (a = ascendente, d descendente)");
        String ingreso2 = scn.nextLine();
        if (ingreso2.equals("a")) {
            ordenAsc = true;
        } else if (ingreso2.equals("d")) {
            ordenAsc = false;
        }
        int[] resultado = ordenado(numeros, ordenAsc);
        for (int y=0; y < numeros.length; y++) {
            System.out.println(resultado[y]);
        }
        scn.close();
    }
}
```

```
private static int[] ordenado(int[] vector, boolean orden) {
    int valor = 0;
    for (int i = 0; i < vector.length; i++) {
        for (int x = i; x < vector.length; x++) {
            if (orden == true) {
                if (vector[x] <= vector[i]) {
                    valor = vector[i];
                    vector[i] = vector[x];
                    vector[x] = valor;
                }
            } else {
                if (vector[x] >= vector[i]) {
                    valor = vector[i];
                    vector[i] = vector[x];
                    vector[x] = valor;
                }
            }
        }
    }
    return vector;
}
```

```
}
```

c)

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

```
public class Ejercicio1C {
```

```
    public static void main(String[] argumentos) {
```

```
        if (argumentos.length > 2) {
```

```
            int numeros[] = new int[argumentos.length-1];
```

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

```
                numeros [i] = Integer.parseInt(argumentos[i]);
```

```
            }
```

```
            int resultado[] = new int[numeros.length];
```

```
            String letra = argumentos[argumentos.length-1];
```

```
            if (Character.isLetter(letra.charAt(0))) {
```

```
                boolean ordenAsc = orden(letra);
```

```
                System.out.println(ordenAsc);
```

```
                resultado = ordenado(numeros, ordenAsc);
```

```
                for (int y=0; y < resultado.length; y++) {
```

```
                    System.out.println(resultado[y]);
```

```
                }
```

```
            } else {
```

```
                System.out.println("El último argumento debe ser un caracter (a = ascendente, d
```

```
= descendente)");
```

```
                System.out.println("El último argumento escrito fue: " + letra);
```

```
            }
```

```
        } else {
```

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

```
            System.out.println("Ingrese números separados por UN espacio");
```

```
            String[] ingreso1 = scn.nextLine().split(" ");
```

```
            int numeros[] = asignarValores(ingreso1);
```

```
            System.out.println("Ingrese orden (a = ascendente, d descendente)");
```

```
            String ingreso2 = scn.nextLine();
```

```
            boolean ordenAsc = orden(ingreso2);
```

```
            int[] resultado = ordenado(numeros, ordenAsc);
```

```
            for (int y=0; y < numeros.length; y++) {
```

```
                System.out.println(resultado[y]);
```

```
            }
```

```
            scn.close();
```

```
        }
```

```
    }
```

```
    private static boolean orden(String letra) {
```

```
        boolean ordenAsc;
```

```
        if (letra.equals("a")) {
```

```
            ordenAsc = true;
```

```
        } else {
```

```
            ordenAsc = false;
```

```
        }
```

```
        return ordenAsc;
```

```
    }
```

```

private static int[] asignarValores(String[] vector){
    int respuesta[] = new int [vector.length];
    for (int i = 0 ; i < vector.length ; i++) {
        respuesta[i] = Integer.parseInt(vector[i]);
    }
    return respuesta;
}

private static int[] ordenado(int[] vector, boolean orden) {
    int valor = 0;
    for (int i = 0; i < vector.length; i++) {
        for (int x = i; x < vector.length; x++) {
            if (orden == true) {
                if (vector[x] <= vector[i]) {
                    valor = vector[i];
                    vector[i] = vector[x];
                    vector[x] = valor;
                }
            } else {
                if (vector[x] >= vector[i]) {
                    valor = vector[i];
                    vector[i] = vector[x];
                    vector[x] = valor;
                }
            }
        }
    }
    return vector;
}
}

```

Ejercicio 2:

```

import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.util.Scanner;

```

```

public class Ejercicio2 {

    public static void main(String[] args) {
        int sumatoria = 0;
        Scanner scn = new Scanner(System.in);
        System.out.println("Ingresa la dirección de un archivo:");
        String url = scn.nextLine();
        Path archivo = Paths.get(url);
        if (url.length() < 1) {
            archivo = Paths.get("src\\Archivos\\numeros.txt");
        }
        System.out.println("Desea sumar (s) o multiplicar (m) los números?");
        String accion = scn.nextLine();
        scn.close();
    }
}

```

```

try {
    if (accion.equals("s")) {
        for (String numeros: Files.readAllLines(archivo)) {
            sumatoria += Integer.parseInt(numeros);
        }
        System.out.println("La suma de los numeros del archivo es: "+sumatoria);
    } else {
        sumatoria = 1;
        for (String numeros: Files.readAllLines(archivo)) {
            sumatoria = sumatoria * Integer.parseInt(numeros);
        }
        System.out.println("El producto de los numeros del archivo es: "+sumatoria);
    }
} catch (IOException e) {

    e.printStackTrace();
}
}
}

```

Ejercicio 3:

```

import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.util.Scanner;

public class Ejercicio3 {

    public static void main(String[] args) throws IOException {
        String texto = "";
        int desplazamiento = 0;
        Path entrada = Paths.get("src\\Archivos\\Clase4Entrada.txt");
        Path salida = Paths.get("src\\Archivos\\Clase4Salida.txt");
        Scanner scn = new Scanner(System.in);
        System.out.println("Ingrese si desea codificar (c) o decodificar (d) un texto:");
        String opcion = scn.nextLine();
        System.out.println("Ingrese la cantidad de desplazamientos que tendrá su codificación:");
        String despl = scn.nextLine();

        System.out.println("Ingrese la ruta del archivo de entrada (solo una linea:");
        String url1 = scn.nextLine();
        if (url1.length() > 1) {
            entrada = Paths.get(url1);
        }
        System.out.println("Ingrese la ruta del archivo de salida:");
        String url2 = scn.nextLine();
        if (url2.length() > 1) {
            salida = Paths.get(url2);
        }
    }
}

```

```

        scn.close();

        for (String lineas: Files.readAllLines(entrada)) {
            texto = lineas;
        }

        try {
            desplazamiento = Integer.parseInt(despl);
        } catch (NumberFormatException e) {
            e.printStackTrace();
        }
        if (opcion.equals("c")) {
            String textoCodificado = codificar(texto, desplazamiento);
            Files.writeString(salida, textoCodificado);
        } else if (opcion.equals("d")) {
            String textoDecodificado = decodificar(texto, desplazamiento);
            Files.writeString(salida, textoDecodificado);
        }

    }

    private static String codificar(String texto, int desplazamiento) {

        String abc = "abcdefghijklmnñopqrstuvwxyz ";

        String resultado = "";

        for (int i = 0; i < texto.length(); i++) {
            char caracter = texto.charAt(i);

            for (int x = 0; x < abc.length(); x++) {
                if (abc.charAt(x) == caracter) {
                    resultado += abc.charAt((x + desplazamiento) % abc.length());
                }
            }
        }
        return resultado;
    }

    private static String decodificar(String texto, int desplazamiento) {
        return codificar(texto, 28 - desplazamiento);
    }
}

```

Ejercicio 4:

Subí todo el proyecto de Eclipse a mi Github, los códigos de esta clase los podés encontrar en:
<https://github.com/IvanSerrudo/curso-java/tree/main/CursoJava/src/Clase4>