

ESTRUCTURAS DE DATOS Y ALGORITMOS

Jonathan Rosero

EJERCICIOS CON ARREGLOS: El siguiente grupo de ejercicios deberá ser realizado en el lenguaje JAVA.

1. Leer una matriz de $n \times m$ elementos y calcular la suma de cada una de sus filas y columnas, dejando dichos resultados en dos vectores, uno de la suma de filas y otro de columnas.

```
package ejercicio1;

public class Ejercicio1 {

    public static void main(String[] args) {
        Metodo_Ejercicio1 met = new Metodo_Ejercicio1();

        met.obtener_limites();
        met.obtener_matriz();
        met.calcular_columna();
        met.calcular_fila();
        met.presentar_matriz();
        met.presentar_resultados();

    }

}
```

```
package ejercicio1;

import java.util.Scanner;

public class Metodo_Ejercicio1 {

    Scanner scan = new Scanner(System.in);
    private int columna;
    private int fila;
    private int[][] matriz;
    private int[] suma_fila;
    private int[] suma_columna;

    public Metodo_Ejercicio1() {

    }

    public void obtener_limites() {
        System.out.println("Ingrese la cantidad de columnas de la matriz: ");
        columna = scan.nextInt();

        System.out.println("Ingrese la cantidad de filas de la matriz: ");
        fila = scan.nextInt();
    }
}
```

```

    }

    public void obtener_matriz() {
        matriz = new int[filas][columnas];
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                System.out.printf("Ingrese un numero para la posicion %s,%d: ", i +
1, j + 1);
                matriz[i][j] = scan.nextInt();
            }
            System.out.println();
        }
    }

    public void calcular_columna() {
        this.suma_columna = new int[columnas];

        for (int i = 0; i < this.matriz[0].length; i++) {
            int suma = 0;
            for (int j = 0; j < this.matriz.length; j++) {
                suma += matriz[j][i];
            }
            this.suma_columna[i] = suma;
        }
    }

    public void calcular_fila() {
        this.suma_fila = new int[filas];

        for (int i = 0; i < this.filas; i++) {
            int suma = 0;
            for (int j = 0; j < this.columnas; j++) {
                suma += matriz[i][j];
            }
            this.suma_fila[i] = suma;
        }
    }

    public void presentar_matriz() {
        for (int i = 0; i < filas; i++) {
            for (int j = 0; j < columnas; j++) {
                System.out.printf("%d ", matriz[i][j]);
            }
            System.out.println();
        }
    }

    public void presentar_resultados() {
        for (int i = 0; i < columnas; i++) {
            System.out.printf("La suma de la columna: %d es: %d\n", i + 1,
this.suma_columna[i]);
        }
        for (int i = 0; i < filas; i++) {

```

```

        System.out.printf("La suma de la fila: %d es: %d\n", i + 1,
this.suma_fila[i]);
    }
}

}

```

2. Calcular el número de elementos negativos, cero y positivos de un vector dado de n elementos.

```

package ejercicio2;

public class Ejercicio2 {

    public static void main(String[] args) {
        Metodo_Ejercicio2 met = new Metodo_Ejercicio2();

        met.establecer_Limite();
        met.establecer_vector();
        met.calcular_neg();
        met.calcular_pos();
        met.calcular_cero();
        met.presentar_vector();
        met.presentar_resultados();
    }

}

```

```

package ejercicio2;

import java.util.Scanner;

public class Metodo_Ejercicio2 {

    private int array[];
    private int lim;
    private int neg = 0;
    private int pos = 0;
    private int cero = 0;
    Scanner scan = new Scanner(System.in);

    public void establecer_Limite() {
        System.out.println("Ingrese un el tamaño del vector: ");
        lim = scan.nextInt();
    }

    public void establecer_vector() {
        array = new int[lim];
        for (int i = 0; i < array.length; i++) {
            System.out.printf("Ingrese un elemento del vector en la posicion %d: ",
i + 1);

            array[i] = scan.nextInt();
            System.out.println();
        }
    }
}

```

```

    }

    public void calcular_neg() {
        for (int i = 0; i < array.length; i++) {
            if (array[i] < 0) {
                neg++;
            }
        }
    }

    public void calcular_pos() {
        for (int i = 0; i < array.length; i++) {
            if (array[i] > 0) {
                pos++;
            }
        }
    }

    public void calcular_cero() {
        for (int i = 0; i < array.length; i++) {
            if (array[i] == 0) {
                cero++;
            }
        }
    }

    public void presentar_vector() {
        for (int i = 0; i < array.length; i++) {
            System.out.printf("%d ", array[i]);
        }
        System.out.println();
    }

    public void presentar_resultados() {
        System.out.printf("El total de numeros negativos en el vector es de: %d\n",
neg);
        System.out.printf("El total de numeros positivos en el vector es de: %d\n",
pos);
        System.out.printf("El total de numeros cero en el vector es de: %d\n",
cero);
    }
}

```

3. Escriba un programa que me permita almacenar en un arreglo B el factorial de 10 números almacenados previamente en un arreglo A.

```

package ejercicio3;

import java.util.Scanner;

public class Ejercicio3 {

    public static void main(String[] args) {
        int A[] = new int[10];
    }
}

```

```

int B[] = new int[10];
Scanner scan = new Scanner(System.in);

for (int i = 0; i < A.length; i++) {
    System.out.println("Ingrese un valor: ");
    A[i] = scan.nextInt();
}

for (int i = 0; i < A.length; i++) {
    B[i] = fact(A[i]);
}

for (int i = 0; i < A.length; i++) {
    System.out.printf("El factorial de %d es: %d\n", A[i], B[i]);
}
}

public static int fact(int num) {
    if (num == 0) {
        return 1;
    }
    if (num <= 1) {
        return 1;
    }
    return num * fact(num - 1);
}
}

```

4. Calcular la suma de los elementos de la diagonal principal de una matriz de $n \times n$.

```

package ejercicio4;

import java.util.Scanner;

public class Ejercicio4 {

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        Metodo_Ejercicio4 met = new Metodo_Ejercicio4();
        int[][] matriz;

        System.out.println("Ingrese el tamaño de la matriz: ");
        int lim = scan.nextInt();

        matriz = met.ingresar_matriz(lim);
        met.presentar_matriz(matriz);
        int diagonal = met.suma_diagonal(matriz);

        System.out.printf("El total de la suma de la diagonal de la matriz es: %d\n", diagonal);
    }
}

```

```

package ejercicio4;

import java.util.Scanner;

public class Metodo_Ejercicio4 {

    Scanner scan = new Scanner(System.in);

    public int[][] ingresar_matriz(int lim) {

        int matriz[][] = new int[lim][lim];
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz.length; j++) {
                System.out.printf("Ingrese un valor en la posicion %d,%d: ", i + 1,
j + 1);

                matriz[i][j] = scan.nextInt();
                System.out.println();
            }
        }
        return matriz;
    }

    public void presentar_matriz(int[][] matriz) {
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz.length; j++) {
                System.out.printf("%d", matriz[i][j]);
            }
            System.out.println();
        }
    }

    public int suma_diagonal(int[][] matriz) {
        int suma = 0;
        for (int i = 0; i < matriz.length; i++) {
            suma = suma + matriz[i][i];
        }
        return suma;
    }
}

```

5. Se dispone de un arreglo T de n números diferentes de 0, (puede ser llenada por un procedimiento que genere números aleatorios distintos de cero); crear un nuevo arreglo en el que todos sus elementos resulten de dividir los elementos del arreglo T por el elemento $T[i]$, siendo i un valor leído por teclado.

```

package ejercicio5;

public class Ejercicio5 {

```

```

    public static void main(String[] args) {
        Metodo_Ejercicio5 met = new Metodo_Ejercicio5();

        met.establecer_Limite();
        met.establecer_vector();
        met.presentar_original();
        met.calcular_div();
        met.presentar();
    }
}

```

```

package ejercicio5;

import java.util.Scanner;

import java.util.Random;

public class Metodo_Ejercicio5 {

    private int T[];
    private int lim;
    private int num;
    private double T2[];
    Scanner scan = new Scanner(System.in);

    public void establecer_Limite() {
        System.out.println("Ingrese un el tamaño del arreglo T: ");
        lim = scan.nextInt();
    }

    public void establecer_vector() {
        T = new int[lim];
        for (int i = 0; i < T.length; i++) {
            T[i] = (int) (Math.random() * 100) + 1;
        }
    }

    public void calcular_div() {
        T2 = new double[lim];
        System.out.println("Elija la posicion del numero que sera el divisor: ");
        num = scan.nextInt() - 1;
        for (int i = 0; i < T.length; i++) {
            T2[i] = T[i] / (double) T[num];
        }
    }

    public void presentar_original() {
        System.out.printf("El arreglo original: ");
        for (int i = 0; i < T.length; i++) {
            System.out.printf("%d ", this.T[i]);
        }
        System.out.println();
    }
}

```

```

    public void presentar() {
        System.out.printf("El arreglo dividido para %d es: ", T[num]);
        for (int i = 0; i < T2.length; i++) {
            System.out.printf("%.2f ", this.T2[i]);
        }
        System.out.println();
    }
}

```

6. Se dispone de un arreglo de n elementos. Se desea diseñar un algoritmo que permita insertar el valor x en la posición k -ésima de la lista, debiendo desplazarse los elementos hacia la derecha para dar lugar al nuevo elemento y desechando el último elemento sobrante.

```

package ejercicio6;

public class Ejercicio6 {

    public static void main(String[] args) {
        Metodo_Ejercicio6 met = new Metodo_Ejercicio6();

        met.establecer_Limite();
        met.establecer_arreglo();
        System.out.println("Lista original");
        met.imprimir();
        met.ingresar_numero();
        System.out.println("Lista con el nuevo ingreso");
        met.imprimir();
    }

}

```

```

package ejercicio6;

import java.util.Scanner;

public class Metodo_Ejercicio6 {

    private int array[];
    private int lim;
    private int pos;
    Scanner scan = new Scanner(System.in);

    public void establecer_Limite() {
        System.out.println("Ingrese un el tamaño del arreglo: ");
        lim = scan.nextInt();
    }

    public void establecer_arreglo() {
        array = new int[lim];
        for (int i = 0; i < array.length; i++) {
            System.out.printf("Ingrese un valor para la posicion %d: \n", i + 1);
            array[i] = scan.nextInt();
        }
    }
}

```



```

    }

    }

    public void ingresar_numero() {
        System.out.println("Ingrese la posicion donde insertar un valor: ");
        pos = scan.nextInt() - 1;
        if (pos >= array.length || pos < 0) {
            System.out.println("La posicion ingresada no se encuentra en la
lista");
        } else {
            for (int i = this.array.length - 1; i > pos; i--) {
                this.array[i] = array[i - 1];
            }
            System.out.printf("Ingrese un numero para la posicion %d: \n", pos +
1);
            array[pos] = this.scan.nextInt();
        }
    }

    public void imprimir() {
        for (int i = 0; i < array.length; i++) {
            System.out.printf("%s ", array[i]);
        }
        System.out.println();
    }
}

```

7. Escribir un programa que permita calcular el cuadrado de los 100 primeros números enteros y a continuación escribir un arreglo que contenga dichos resultados.

```

package ejercicio7;

public class Ejercicio7 {

    public static void main(String[] args) {
        Metodo_Ejercicio7 met = new Metodo_Ejercicio7();
        met.establecer_arreglo();
        met.presentar();
    }

}

```

```

package ejercicio7;

public class Metodo_Ejercicio7 {

    private int array[];

    public void establecer_arreglo() {

```

```

        array = new int[100];
        for (int i = 0; i < array.length; i++) {
            array[i] = (i + 1) * (i + 1);
        }

    }

    public void presentar() {
        for (int i = 0; i < array.length; i++) {
            System.out.printf("El cuadrado de %d es: %d\n", i + 1, array[i]);
        }
    }
}

```

8. Dada una lista L de $n \times m$ elementos, diseñar un programa que calcule la suma de números pares e impares contenidos en ella.

```

package ejercicio8;

import java.util.Scanner;

public class Ejercicio8 {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
        Metodo_Ejercicio8 met = new Metodo_Ejercicio8();
        int[][] matriz;

        System.out.println("Ingrese las columnas de la matriz: ");
        int columnas = scan.nextInt();

        System.out.println("Ingrese las filas de la matriz: ");
        int filas = scan.nextInt();

        matriz = met.ingresar_matriz(filas, columnas);
        met.presentar_matriz(matriz);
        int pares = met.suma_pares(matriz);
        int impares = met.suma_impares(matriz);

        System.out.printf("El total de la suma de los pares de la matriz es: %d\n"
            + "El total de la suma de los impares de la matriz es: %d\n", pares,
            impares);

    }

}

```

```

package ejercicio8;

import java.util.Scanner;

```

```

public class Metodo_Ejercicio8 {

    Scanner scan = new Scanner(System.in);

    public int[][] ingresar_matriz(int fila, int colum) {

        int matriz[][] = new int[fila][colum];
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                System.out.printf("Ingrese un valor en la posicion %d,%d: ", i + 1,
j + 1);

                matriz[i][j] = scan.nextInt();
                System.out.println();
            }
        }
        return matriz;
    }

    public void presentar_matriz(int[][] matriz) {
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                System.out.printf("%d\t", matriz[i][j]);
            }
            System.out.println();
        }
    }

    public int suma_pares(int[][] matriz) {
        int pares = 0;
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                if (matriz[i][j] % 2 == 0) {
                    pares++;
                }
            }
        }
        return pares;
    }

    public int suma_impares(int[][] matriz) {
        int impares = 0;
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                if (matriz[i][j] % 2 != 0) {
                    impares++;
                }
            }
        }
        return impares;
    }

}

```

9. Se desea realizar un programa que permita controlar las reservas de plazas de un vuelo

Quito ☐ **Loja**, de acuerdo con las siguientes normas de la compañía aérea.

- a. Número de plazas del avión: 100.
- b. Plazas numeradas del 1 al 30: fumadores.
- c. Plazas numeradas del 31 al 100: no fumadores.

Se desea realizar la reserva a petición del pasajero y cerrar la reserva cuando no haya plazas libres o el avión esté próximo a despegar. Como ampliación de este algoritmo, considere la opción de anulaciones imprevistas de reservas.

```
package ejercicio9;

public class Ejercicio9 {

    public static void main(String[] args) {
        int opcion1;

        int[] array = new int[100];

        Metodo_Ejercicio9 obj = new Metodo_Ejercicio9();
        do {
            opcion1 = obj.MenuP();
            switch (opcion1) {
                case 1:
                    obj.leerA(array);
                    break;
                case 2:
                    obj.presentarA(array);
                    break;
                case 3:
                    array = obj.insertarA(array);
                    break;
                case 4:
                    array = obj.eliminarA(array);
                    break;
                case 5:
                    System.out.printf("El total de espacios vacios son: %d\n"
                                     + "El total de espacios ocupados son: %d\n",
obj.getLibre(), obj.getOcupado());
                    break;
                case 6:
                    array = obj.eliminarA(array);
                    break;
                case 0:
                    System.out.println("Gracias por su atencion");
                    break;
            }
        } while (opcion1 != 0);
    }
}
```

```
}
```

```
package ejercicio9;

import java.util.Scanner;

public class Metodo_Ejercicio9 {

    Scanner scan = new Scanner(System.in);
    private int libre;
    private int ocupado;

    public int MenuP() {
        int opcion1;
        System.out.printf("MENU\n1.INGRESAR RESERVAS\n2.PRESENTAR\n3.INSERTAR\n4.ELIMINAR RESERVAS\n5.Cantidad de asientos LIBRES Y OCUPADOS\n0.SALIR\nIngrese el numero del menu: \n");
        opcion1 = scan.nextInt();
        return opcion1;
    }

    public void leerA(int[] A) {
        libre = A.length;
        ocupado = 0;
        int i = 0;
        int j = 30;
        while (ocupado < A.length) {
            System.out.printf("Ingrese un una reserva:\n"
                + "1.Fumador\n"
                + "2.No Fumador\n"
                + "3.Terminar el ingreso\nIngrese el numero de la accion: \n");
            int ingreso = scan.nextInt();
            if (ingreso == 1) {
                A[i] = 1;
                libre--;
                ocupado++;
                i++;
            } else if (ingreso == 2) {
                A[j] = 1;
                libre--;
                ocupado++;
                j++;
            } else {
                break;
            }
        }
    }

    public void presentarA(int[] A) {
        System.out.println("La disposicion del avion es 1 asiento reservado 0 asiento libre: ");
        System.out.println("FUMADORES");
        for (int i = 0; i < A.length; i++) {
            if (i == 30) {
```

```

        System.out.println("\n\nNO FUMADORES");
    }
    System.out.printf("%d ", A[i]);
}
System.out.println();

}

public int[] insertarA(int[] A) {
    if (this.libre == 0) {
        System.out.println("No existe lugares vacios donde ingresar mas
numeros");
    } else {
        System.out.println("Ingrese el asiento a reservar:\n"
            + "1-30.FUMADORES\n"
            + "31-100.NO FUMADORES");
        int pos = scan.nextInt() - 1;
        if (pos > A.length) {
            System.out.println("El numero del asiento no existe.");
        } else {
            if (A[pos] != 0) {
                System.out.println("El asiento ya esta reservado");
            } else if (A[pos] == 0) {
                A[pos] = 1;
                this.libre--;
                this.ocupado++;
            }
        }
    }

    return A;
}

public int[] eliminarA(int[] A) {
    if (libre == A.length) {
        System.out.println("No existen reservas a anular");
    } else {
        System.out.println("Ingrese el asiento del pasajero a anular reserva");
        int pos = scan.nextInt() - 1;
        A[pos] = 0;
        ocupado--;
        libre++;
    }
    return A;
}

public int getLibre() {
    return libre;
}

public int getOcupado() {
    return ocupado;
}

}

```

10. Cada alumno de una clase de 20 estudiantes tiene notas correspondientes a 5 asignaturas diferentes, pudiendo no tener calificación en alguna asignatura. A cada asignatura le corresponde un determinado coeficiente. Escribir un programa que permita:

- a. Calcular la media de cada alumno
- b. Media general de la clase
- c. Media de la clase en cada asignatura
- d. Porcentaje de faltas (no presentado a examen)

```
package ejercicio10;

public class Ejercicio10 {

    public static void main(String[] args) {
        int n = 20, m = 5;

        Metodo_Ejercicio10 met = new Metodo_Ejercicio10(n, m);

        met.media_estudiante();
        met.media_total();
    }
}
```

```
package ejercicio10;

import java.util.Scanner;

public class Metodo_Ejercicio10 {

    Scanner input = new Scanner(System.in);

    private double[][] matriz;
    private int n, m;
    private double suma, media;
    private double prom, asig;
    private double media_total, asig_total;
    private double faltas;

    public Metodo_Ejercicio10(int n, int m) {
        this.n = n;
        this.m = m;
        this.matriz = new double[n][m];
    }

    public void media_estudiante() {
        for (int i = 0; i < this.matriz.length; i++) {
            this.suma = 0;
            System.out.printf("\nEstudiante %d\n", i + 1);
            for (int j = 0; j < this.matriz[0].length; j++) {
```

```

        System.out.printf("Introduzca la nota de la asignatura %d:\n", j +
1);

        this.matriz[i][j] = input.nextDouble();
        this.suma += matriz[i][j];
    }
    this.media = this.suma / this.matriz[0].length;
    this.prom += media;
    System.out.printf("La media del estudiante %d es: %.2f\n", i + 1,
this.media);
    }
}

public void media_total() {
    this.media_total = prom / this.matriz.length;
    System.out.printf("\nEl promedio total de la clase es: %.2f\n\n",
this.media_total);

    for (int j = 0; j < this.matriz[0].length; j++) {
        for (int i = 0; i < 20; i++) {
            this.asig += matriz[i][j];
            if (matriz[i][j] == 0) {
                this.faltas++;
            }
        }
        this.asig_total = asig / this.matriz.length;
        System.out.printf("El promedio de la asignatura %d es: %.2f\n", j + 1,
this.asig_total);
    }
    faltas = (faltas * 100) / (this.matriz.length*this.matriz[0].length);
    System.out.printf("\nEl porcentaje de faltas es: %.2f%s\n", this.faltas,
"%");
}
}

```

11. Una empresa tiene diez almacenes y desea un programa que lea sus ventas mensuales, calcule la media de las ventas y obtener un listado de los almacenes cuyas ventas mensuales son superiores a la media.

```

package ejercicio11;

public class Ejercicio11 {

    public static void main(String[] args) {
        Metodo_Ejercicio11 met = new Metodo_Ejercicio11();
        met.ingresar_ventas();
        met.obtener_medias();
        met.obtener_mayores();
        met.presentar();
    }

}

```



```

package ejercicio11;

import java.util.Scanner;

public class Metodo_Ejercicio11 {

    Scanner scan = new Scanner(System.in);
    private double[][] ventas = new double[10][12];
    private double[] media = new double[10];
    private double media_total;
    private int[] mayores;

    public void ingresar_ventas() {
        for (int i = 0; i < this.ventas.length; i++) {
            for (int j = 0; j < this.ventas[0].length; j++) {
                System.out.printf("Ingrese las Ventas del ALMACEN %d del mes %d:
\n", i + 1, j + 1);
                ventas[i][j] = scan.nextDouble();
            }
        }
    }

    public void obtener_medias() {
        for (int i = 0; i < this.ventas.length; i++) {
            double suma = 0;
            for (int j = 0; j < this.ventas[0].length; j++) {
                suma += this.ventas[i][j];
            }
            this.media[i] = suma / this.ventas[0].length;
        }
    }

    public double obtener_media_total(double[] array) {
        double suma = 0;
        for (int i = 0; i < array.length; i++) {
            suma = suma + array[i];
        }
        double prom = suma / array.length;
        return prom;
    }

    public void obtener_mayores() {
        int cont = 0;
        media_total = this.obtener_media_total(media);
        for (int i = 0; i < this.ventas.length; i++) {
            if (this.media[i] > this.media_total) {
                cont++;
            }
        }
        this.mayores = new int[cont];
        int k = 0;
        for (int i = 0; i < this.ventas.length; i++) {
            if (this.media[i] > this.media_total) {
                this.mayores[k] = i + 1;
                k++;
            }
        }
    }
}

```

```

    }
}

public void presentar() {
    for (int i = 0; i < this.ventas.length; i++) {
        System.out.printf("\nALMACEN %d\n", i + 1);
        for (int j = 0; j < this.ventas[0].length; j++) {
            System.out.printf("Las ventas en el mes %d son: %.2f\n", j + 1,
this.ventas[i][j]);
        }
        System.out.printf("Su media en venta anual es de: %.2f\n",
this.media[i]);
    }
    System.out.printf("\nLa media total en ventas de los almacenes es: %.2f\n"
        + "Los almacenes que superan la media son:\n", this.media_total);
    for (int i = 0; i < this.mayores.length; i++) {
        System.out.printf(" ALMACEN %d \n", this.mayores[i]);
    }
}
}

```

12. Se dispone de una lista de n nombres de alumnos. Escribir un programa que solicite el nombre de un alumno y busque en la lista si el nombre está o no.

```

package ejercicio12;

import java.util.Scanner;

public class Ejercicio12 {

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int lim;

        Metodo_Ejercicio12 met = new Metodo_Ejercicio12();
        System.out.println("Ingrese el tamaño de la lista: ");
        lim = scan.nextInt();
        met.ingresar_nombres(lim);
        met.presentar_nombres();
        met.buscar_nombre();
    }

}

```

```

package ejercicio12;

import java.util.Scanner;

public class Metodo_Ejercicio12 {

    Scanner scan = new Scanner(System.in);

```

```

private String[] nombres;
private String buscar;

public void ingresar_nombres(int lim) {
    nombres = new String[lim];
    for (int i = 0; i < this.nombres.length; i++) {
        System.out.printf("Ingrese un nombre para la posicion %d:\n", i + 1);
        nombres[i] = scan.nextLine().toUpperCase();
    }
}

public void presentar_nombres() {
    for (int i = 0; i < this.nombres.length; i++) {
        System.out.println(nombres[i]);
    }
}

public void buscar_nombre() {
    System.out.println("Ingrese el nombre a buscar: ");
    boolean bandera = false;
    buscar = scan.nextLine().toUpperCase();
    for (int i = 0; i < this.nombres.length; i++) {
        if (buscar == nombres[i]) {
            System.out.printf("El nombre de %s se encuentra en la posicion:
%d\n", buscar, i + 1);
            bandera = true;
        } else {
            bandera = false;
        }
    }
    if (bandera = false) {
        System.out.printf("El nombre de %s no se encuentra en la lista\n",
buscar);
    }
}
}

```

13. Realice un programa que me permita realizar (según un menú presentado al usuario):

- a. La multiplicación de dos matrices bidimensionales.
- b. Suma de matrices.
- c. Resta de matrices.
- d. Multiplicación por un escalar
- e. Traspuesta de una matriz

```

package ejercicio13;

import java.util.Scanner;

```

```

public class Ejercicio13 {

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int opcion1;
        int opcion2;

        Metodo_Ejercicio13 obj = new Metodo_Ejercicio13();
        do {
            opcion1 = obj.MenuP();
            switch (opcion1) {
                case 1:
                    obj.leerA();
                    break;
                case 2:
                    obj.leerB();
                    break;
                case 3:
                    obj.presentar();
                    break;
                case 4:
                    obj.multiplicar();
                    break;
                case 5:
                    obj.sumar();
                    break;
                case 6:
                    obj.restar();
                    break;
                case 7:
                    opcion2 = obj.Submenu();
                    obj.multiplicar_escalar(opcion2);
                    break;
                case 8:
                    opcion2 = obj.Submenu();
                    obj.transpuesta(opcion2);
                case 0:
                    System.out.println("Gracias por su atencion");
                    break;
            }
        } while (opcion1 != 0);
    }
}

```

```

package ejercicio13;

import java.util.Scanner;

public class Metodo_Ejercicio13 {

    Scanner scan = new Scanner(System.in);
    private int[][] A;

```

```

private int[][] B;

public int MenuP() {
    int opcion1;
    System.out.printf("MENU\n1.INGRESAR MATRIZ A\n2.INGRESAR MATRIZ B\n3.PRESENTAR MATRICES\n4.MULTIPLICAR MATRICES BIDIMENSIONALES\n5.SUMA DE MATRICES\n6.DIFERENCIA DE MATRICES\n7.MULTIPLICACION DE UNA MATRIZ POR UN ESCALAR\n8.TRANSPUESTA DE UNA MATRIZ\n0.SALIR\nIngrese el numero del menu: \n");
    opcion1 = scan.nextInt();
    return opcion1;
}

public int Submenu() {
    int opcion2;
    System.out.printf("1.MATRIZ A\n2.MATRIZ B\nIngrese el numero con la matriz que quiera trabajar:\n");
    opcion2 = scan.nextInt();
    return opcion2;
}

public void leerA() {
    System.out.println("Matriz A");
    System.out.println("Ingrese la cantidad de filas de la matriz A:");
    int filas = scan.nextInt();
    System.out.println("Ingrese la cantidad de columnas de la matriz A:");
    int columnas = scan.nextInt();
    A = new int[filas][columnas];
    for (int i = 0; i < A.length; i++) {
        for (int j = 0; j < A[0].length; j++) {
            System.out.printf("Ingrese un numero para la posicion %d,%d de la Matriz A:\n", i + 1, j + 1);
            A[i][j] = scan.nextInt();
        }
    }
}

public void leerB() {
    System.out.println("Matriz B");
    System.out.println("Ingrese la cantidad de filas de la matriz B:");
    int filas = scan.nextInt();
    System.out.println("Ingrese la cantidad de columnas de la matriz B:");
    int columnas = scan.nextInt();
    B = new int[filas][columnas];
    for (int i = 0; i < B.length; i++) {
        for (int j = 0; j < B[0].length; j++) {
            System.out.printf("Ingrese un numero para la posicion %d,%d de la Matriz B:\n", i + 1, j + 1);
            B[i][j] = scan.nextInt();
        }
    }
}

public void presentar() {
    System.out.println("MATRIZ A");
    for (int i = 0; i < A.length; i++) {
        for (int j = 0; j < A[0].length; j++) {
            System.out.printf("%d ", A[i][j]);

```

```

        }
        System.out.println();
    }
    System.out.println("MATRIZ B");
    for (int i = 0; i < B.length; i++) {
        for (int j = 0; j < B[0].length; j++) {
            System.out.printf("%d ", B[i][j]);
        }
        System.out.println();
    }
}

public void multiplicar() {
    int[][] mult;
    if ((A.length == 2) && (A[0].length == 2)) && ((B.length == 2) &&
(B[0].length == 2))) {
        System.out.println("La matriz resultante de AxB es: ");
        mult = new int[A.length][B[0].length];
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < B[0].length; j++) {
                for (int k = 0; k < A[0].length; k++) {
                    mult[i][j] += A[i][k] * B[k][j];
                }
                System.out.print(mult[i][j] + " ");
            }
            System.out.println();
        }
    } else {
        System.out.println("Las matrices no pueden ser multiplicadas");
    }
}

public void sumar() {
    if ((A.length == B.length) && (B[0].length == A[0].length)) {
        System.out.println("La matriz resultante de A+B es: ");
        int[][] suma = new int[A.length][A[0].length];
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A.length; j++) {
                suma[i][j] = A[i][j] + B[i][j];
                System.out.print(suma[i][j] + " ");
            }
            System.out.println();
        }
    } else {
        System.out.println("Las matrices no pueden ser sumadas");
    }
}

public void restar() {
    if ((A.length == B.length) && (B[0].length == A[0].length)) {
        System.out.println("La matriz resultante de A-B es: ");
        int[][] suma = new int[A.length][A[0].length];
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A.length; j++) {
                suma[i][j] = A[i][j] - B[i][j];
                System.out.print(suma[i][j] + " ");
            }
        }
    }
}

```

```

        System.out.println();
    }
    } else {
        System.out.println("Las matrices no pueden ser restadas");
    }
}

public void multiplicar_escalar(int opcion) {
    System.out.println("Ingrese un numero escalar: ");
    int num = scan.nextInt();
    if (opcion == 1) {
        System.out.printf("El resultado de %d por la Matriz A es: \n", num);
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A[0].length; j++) {
                A[i][j] = A[i][j] * num;
                System.out.print(A[i][j] + " ");
            }
            System.out.println();
        }
    } else if (opcion == 2) {
        System.out.printf("El resultado de %d por la Matriz B es: \n", num);
        for (int i = 0; i < B.length; i++) {
            for (int j = 0; j < B[0].length; j++) {
                B[i][j] = B[i][j] * num;
                System.out.print(B[i][j] + " ");
            }
            System.out.println();
        }
    }
}

public void transpuesta(int opcion) {
    if (opcion == 1) {
        System.out.println("La transpuesta de la matriz A es:");
        for (int i = 0; i < A[0].length; i++) {
            for (int j = 0; j < A.length; j++) {
                System.out.printf("%d ", A[i][j]);
            }
            System.out.println();
        }
    } else if (opcion == 2) {
        System.out.println("La transpuesta de la matriz B es:");
        for (int i = 0; i < B[0].length; i++) {
            for (int j = 0; j < B.length; j++) {
                System.out.printf("%d ", B[i][j]);
            }
            System.out.println();
        }
    }
}
}
}

```

14. Realice un programa que me permita realizar el ordenamiento de una matriz bidireccional (el ordenamiento debe ser realizado en la misma matriz)

```

package ejercicio14;

public class Ejercicio14 {

    public static void main(String[] args) {
        Metodo_Ejercicio14 met = new Metodo_Ejercicio14();

        met.ingresar_matriz();
        System.out.println("La matriz original es:");
        met.imprimir();
        met.ordenar();
        System.out.println("La matriz ordenada es:");
        met.imprimir();
    }

}

```

```

package ejercicio14;

import java.util.Scanner;

public class Metodo_Ejercicio14 {

    Scanner scan = new Scanner(System.in);
    private int[][] matriz;

    public void ingresar_matriz() {
        System.out.println("Introduzca las filas:");
        int filas = scan.nextInt();
        System.out.println("Introduzca las columnas:");
        int columnas = scan.nextInt();
        matriz = new int[filas][columnas];
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                System.out.printf("Introduzca el numero en la pocision %d,%d:\n", i
+ 1, j + 1);
                matriz[i][j] = scan.nextInt();
            }
        }
    }

    public void imprimir() {
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                System.out.printf("%d ", matriz[i][j]);
            }
            System.out.println();
        }
    }

    public void ordenar() {
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                for (int x = 0; x < matriz.length; x++) {

```


15. Diseñe un programa que me permita implementar el algoritmo del “cuadrado mágico” de un número de filas y columnas ingresado por teclado.

```
package ejercicio15;

public class Metodo_Ejercicio15 {
    public int[][] cuadroMagico(int n){
        int matriz[][]=new int[n][n];
        int contador=1;
        int c_ult=0;
        int f_ult=0;
        int c_sig=n/2;
        int f_sig=0;
        for (int i = 0; i < n; i++){
            for (int j = 0; j < n; j++){
                matriz[f_sig][c_sig] = contador;
            }
        }
    }
}
```

```

        f_ult = f_sig;
        c_ult = c_sig;

        f_sig=f_ult-1;
        c_sig = c_ult+1;

        if(f_sig < 0){
            f_sig = n-1;
        }
        if(c_sig >= n){
            f_sig = f_ult-1;
            c_sig = 0;
        }

        if(f_sig == -1 || matriz[f_sig][c_sig] != 0){
            f_sig = f_ult+1;
            c_sig = c_ult;
        }
        contador++;
    }
}
return matriz;
}

public void presentar(int [][]matriz){
    for(int i = 0; i < matriz.length; i++){
        for(int j = 0; j < matriz[0].length; j++){
            System.out.printf("%d  ", matriz[i][j]);
        }
        System.out.println();
    }
}
}

```

16. Diseñe un programa que permita encontrar el determinante de una matriz.

```

package ejercicio16;

import java.util.Scanner;

public class Ejercicio16 {

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        Metodo_Ejercicio16 met = new Metodo_Ejercicio16();
        System.out.println("Ingrese el tamaño de la matriz: ");
        int tamano = scan.nextInt();
        int matriz[][] = new int[tamano][tamano];
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                System.out.printf("Ingrese un numero para la posicio %d,%d: \n", i
+ 1, j + 1);
                matriz[i][j] = scan.nextInt();
            }
        }
    }
}

```

```

    }
}

met.printmat(matriz);
System.out.println(met.determin(matriz));
}
}

```

```

package ejercicio16;

public class Metodo_Ejercicio16 {

    public void printmat(int matriz[][]) {
        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz.length; j++) {
                System.out.print(String.valueOf(matriz[i][j]) + "\t");
            }
            System.out.println();
        }
        System.out.println();
    }

    public int[][] submatrize(int matriz[][], int x, int y) {
        int submatriz[][] = new int[matriz.length - 1][matriz.length - 1];
        int cur_x = 0;
        int cur_y = 0;
        for (int i = 0; i < matriz.length; i++) {
            if (i != x) {
                cur_y = 0;
                for (int j = 0; j < matriz.length; j++) {
                    if (j != y) {
                        submatriz[cur_x][cur_y] = matriz[i][j];
                        cur_y++;
                    }
                }
                cur_x++;
            }
        }
        return submatriz;
    }

    public int determin(int matriz[][]) {
        int deter = 0;
        int mult = 1;
        if (matriz.length > 2) {
            for (int i = 0; i < matriz.length; i++) {
                deter += mult * matriz[i][0] * determin(submatrize(matriz, i, 0));
                mult *= -1;
            }
            return deter;
        } else {
            return matriz[0][0] * matriz[1][1] - matriz[0][1] * matriz[1][0];
        }
    }
}

```

}