

Nombre Apellido:		NOTA	
Módulo:	Entornos de desarrollo	Fecha:	11/05/2025
C.F:	DAM_SM	Curso:	1º
Profesorado:	Joan Agustí Suárez		
UD3 y UD4:	Pruebas, refactorización y JavaDoc	11:15-13:10	

NOTA: En el bloque examen tercera evaluación (esta vez no es terrible) tenéis disponible el fichero .java del que se habla a continuación, además la tarea donde entregar los documentos que generéis.

```
public class Calculadora {

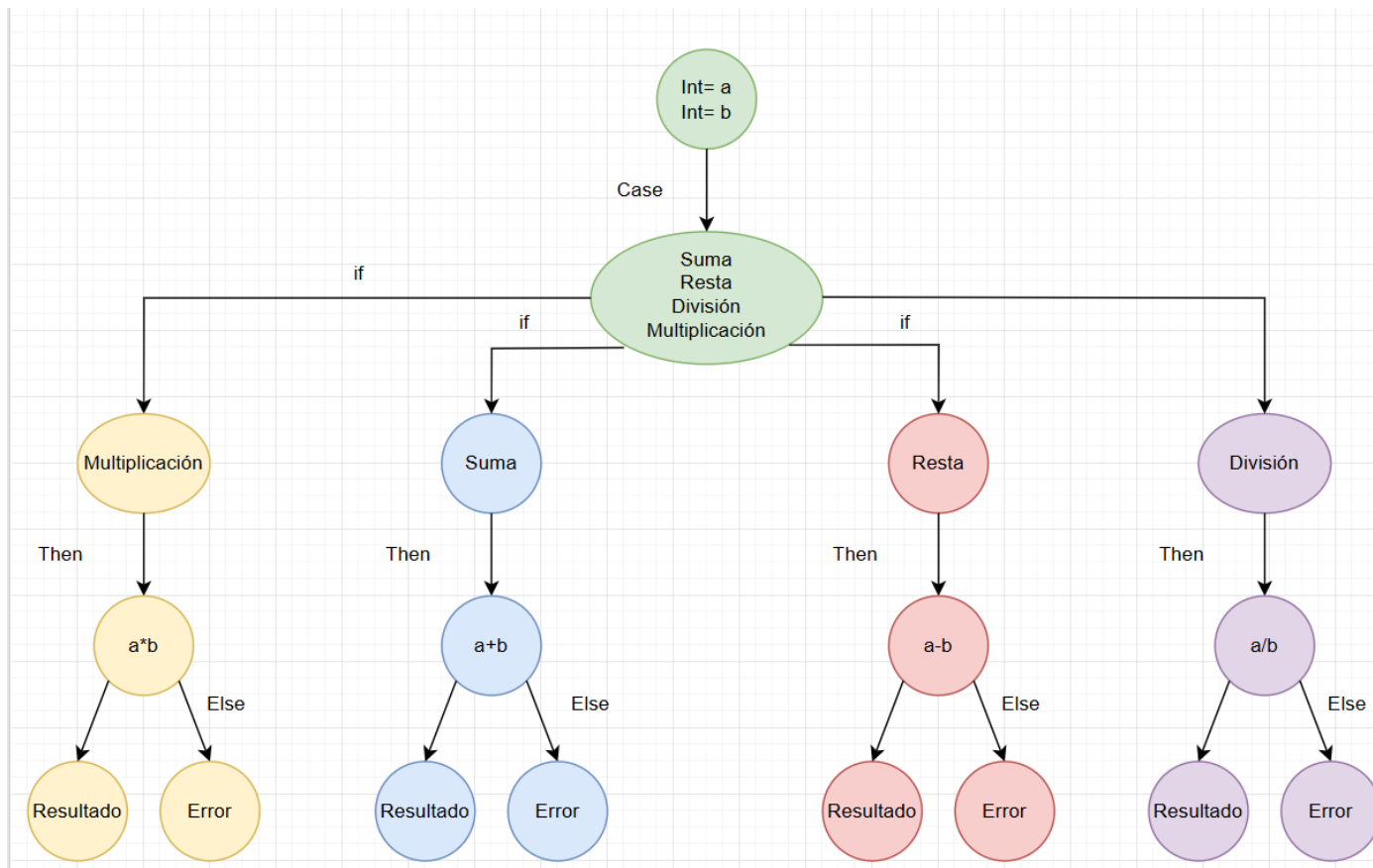
    public int operar(String operador, int a, int b) {
        if (operador.equals("sumar")) {
            System.out.println("🔧 Iniciando operación: SUMAR");
            int resultado = a + b;
            System.out.println("Resultado: " + resultado);
            return resultado;
        } else if (operador.equals("restar")) {
            System.out.println("🔧 Iniciando operación: RESTAR");
            int resultado = a - b;
            System.out.println("Resultado: " + resultado);
            return resultado;
        } else if (operador.equals("multiplicar")) {
            System.out.println("🔧 Iniciando operación: MULTIPLICAR");
            int resultado = a * b;
            System.out.println("Resultado: " + resultado);
            return resultado;
        } else if (operador.equals("dividir")) {
            System.out.println("🔧 Iniciando operación: DIVIDIR");
            if (b == 0) {
                System.out.println("⚠ Error: División por cero");
                throw new ArithmeticException("División por cero");
            }
            int resultado = a / b;
            System.out.println("Resultado: " + resultado);
            return resultado;
        } else {
            System.out.println("⚠ Error: Operación no válida");
            return 0;
        }
    }
}
```

Actividad 1. Caja negra. (La cantidad de filas es orientativa, no quiere decir que tengáis que hacer 7)

Prueba	Resultado esperado	Resultado obtenido	Conclusión
Suma: a+b	2	2	Operación válida
Suma: a-b	2	Error	La operación es inválida
Resta: a-b	2	2	Operación válida
Multiplicación: a*c	2	Error	Datos de la operación incorrectos, es inválida
División: a*b	2	Error	La operación es inválida
División: a/b	2	2	Operación válida
Multiplicación: a*b	2	2	Operación válida

Actividad 2. Caja Blanca. Dibuja el grafo (diagrama de nodos como algunos le llamáis) marcando claramente los nodos predicado y calcula la complejidad ciclomática.

Rellena una tabla con los diferentes caminos que pueden existir.



Actividad 3. JUnit. Realiza las pruebas unitarias necesarias para cubrir el 100% de este código.

Actividad 4. Aplica todos los patrones de **refactorización** que detectes. Enumera los patrones que has utilizado y el porqué.

```
import java.util.Scanner;

/**
 *
 * @author Iván Martínez Guillén
 */

public class Calculadora {

    public static void main(String[] args) {
        Calculadora calc = new Calculadora();
        Scanner scanner = new Scanner(System.in);
        String nombre = "Mi nombre";

        System.out.println(" Bienvenido a la Calculadora de " + nombre);
        System.out.print("Introduce la operación (sumar, restar, multiplicar, dividir): ");
        String operador = scanner.nextLine();

        System.out.print("Introduce el primer número: ");
        int a = scanner.nextInt();

        System.out.print("Introduce el segundo número: ");
        int b = scanner.nextInt();

        // Este cambio se ha realizado para acortar número de líneas.
        Calc_refactor(calc, operador, a, b);
    }

    private static void Calc_refactor(Calculadora calc, String operador, int a, int b) {
        try {
            int resultado = calc.operar(operador, a, b);
            System.out.println(" Resultado: " + resultado);
        } catch (Exception e) {
            System.out.println(" Error: " + e.getMessage());
        }
    }

    // He realizado este cambio en los métodos para acortar el número de líneas
    // necesarias en el código.

    public int operar(String operador, int a, int b) {
        if (operador.equals("sumar")) {
            return Resultado(" Iniciando operación: SUMAR", a + b);
        } else if (operador.equals("restar")) {
            return Resultado(" Iniciando operación: RESTAR", a - b);
        }
    }
}
```

```

    } else if (operador.equals("multiplicar")) {
        return Resultado(" Iniciando operación: MULTIPLICAR", a * b);

    } else if (operador.equals("dividir")) {
        System.out.println(" Iniciando operación: DIVIDIR");
        if (b == 0) {
            System.out.println(" Error: División por cero");
            throw new ArithmeticException("División por cero");
        }
        int resultado = a / b;
        System.out.println("Resultado: " + resultado);
        return resultado;

    } else {
        System.out.println(" ⚠ Error: Operación no válida");
        return 0;
    }
}

// Extrae los datos a partir del refactorizado realizado

private static int Resultado(String x, int a) {
    System.out.println(x);
    int resultado = a;
    System.out.println("Resultado: " + resultado);
    return resultado;
}
}

```

Actividad 5. Documenta el código que has obtenido lo máximo que puedas e indica a continuación los pasos a seguir si queremos generar el JavaDoc.