

Nombre Apellido:	Rubén Gálvez Álvarez	NOTA	
Módulo:	Entornos de desarrollo	Fecha:	11/05/2025
C.F:	DAW_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 {

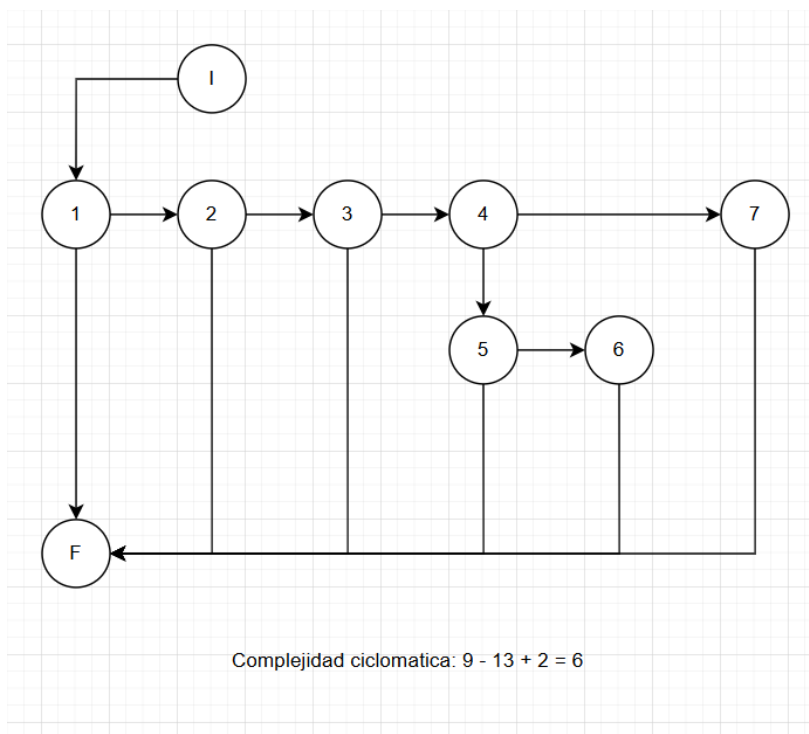
    public int operar(String operador, int a, int b) {
        if (operador.equals("sumar")) {
            System.out.println("🔧 Iniciando operación: SUMAR");
            int resultado = a + b;
            System.out.println("Resultado: " + resultado);
            return resultado;
        } else if (operador.equals("restar")) {
            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
Operador = sumar Valor a = 4 Valor b = 2	6	6	Correcto
Operador = restar Valor a = 4 Valor b = 2	2	2	Correcto
Operador = multiplicar Valor a = 4 Valor b = 2	8	8	Correcto
Operador = dividir Valor a = 4 Valor b = 2	2	2	Correcto
Operador = dividir Valor a = 4 Valor b = 0	No se puede	Error: división por 0	Correcto (esta bien que de error)
Operador = (cualquier cosa diferente a sumar, restar, multiplicar o dividir) Valor a = 4 Valor b = 2	No se puede	Error: operación no válida resultado = 0	Correcto (esta bien que de error) y según el programa, si el operador está mal, el resultado es = 0
Operador = (cualquiera válida) Valor de a o b diferente a un número entero	No se puede	InputMismatchException	No se ejecuta el programa, pero es un error por defecto

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.



Para que lo

entiendas mejor, 1 es un if; 2, 3 y 4 son else if; 7 es la opción default, el 5 es el if dentro de "dividir" y 6 si el valor es "0". He puesto directamente las funciones lógicas, no he puesto el contenido de cada cosa, por lo tanto, ya de por sí es el "esqueleto" entonces no hay predicado, ya que en sí "todas" son.

```
If (loquesea){ // solo pongo esto
    hagoalgo()
}
```

}

CASO	CAMINO	VALOR VARIABLE "OPERADOR"	EXPLICACIÓN	RESULTADO
1	I, 1, 2, 3, 4, 5, 6, F	dividir	Se hace una división por 0	Error: división por 0
2	I, 1, 2, 3, 4, 5, F	dividir	Se hace una división válida	$a/b = x$
3	I, 1, 2, 3, 4, 7, F	Joan ponme un 10 (o cualquier cosa diferente a sumar, restar, multiplicar o dividir)	No se ha escrito un operador válido	Error: operación no válida resultado = 0
4	I, 1, 2, 3, F	multiplicar	Se hace una multiplicación	$a*b = x$
5	I, 1, 2, F	restar	Se hace una resta	$a-b = x$
6	I, 1, F	sumar	Se hace una suma	$a+b = x$

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.

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.

Tres puntos, tools, generate javadoc, generate

He hecho 4 y 5 en el propio .java