

# ENTORNOS DE DESARROLLO

## UNIDAD 3

### TAREA ED03

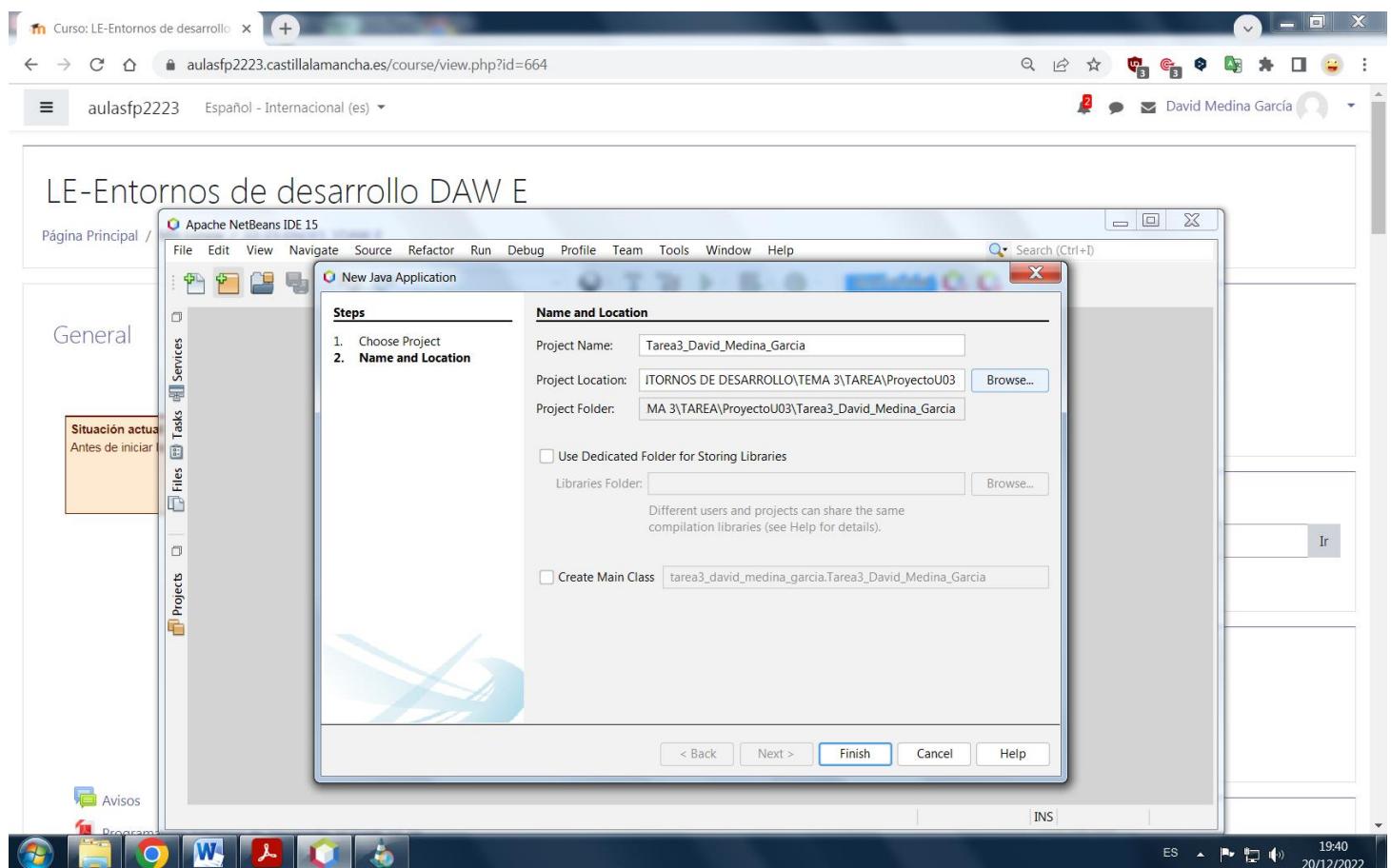


DAVID MEDINA GARCIA

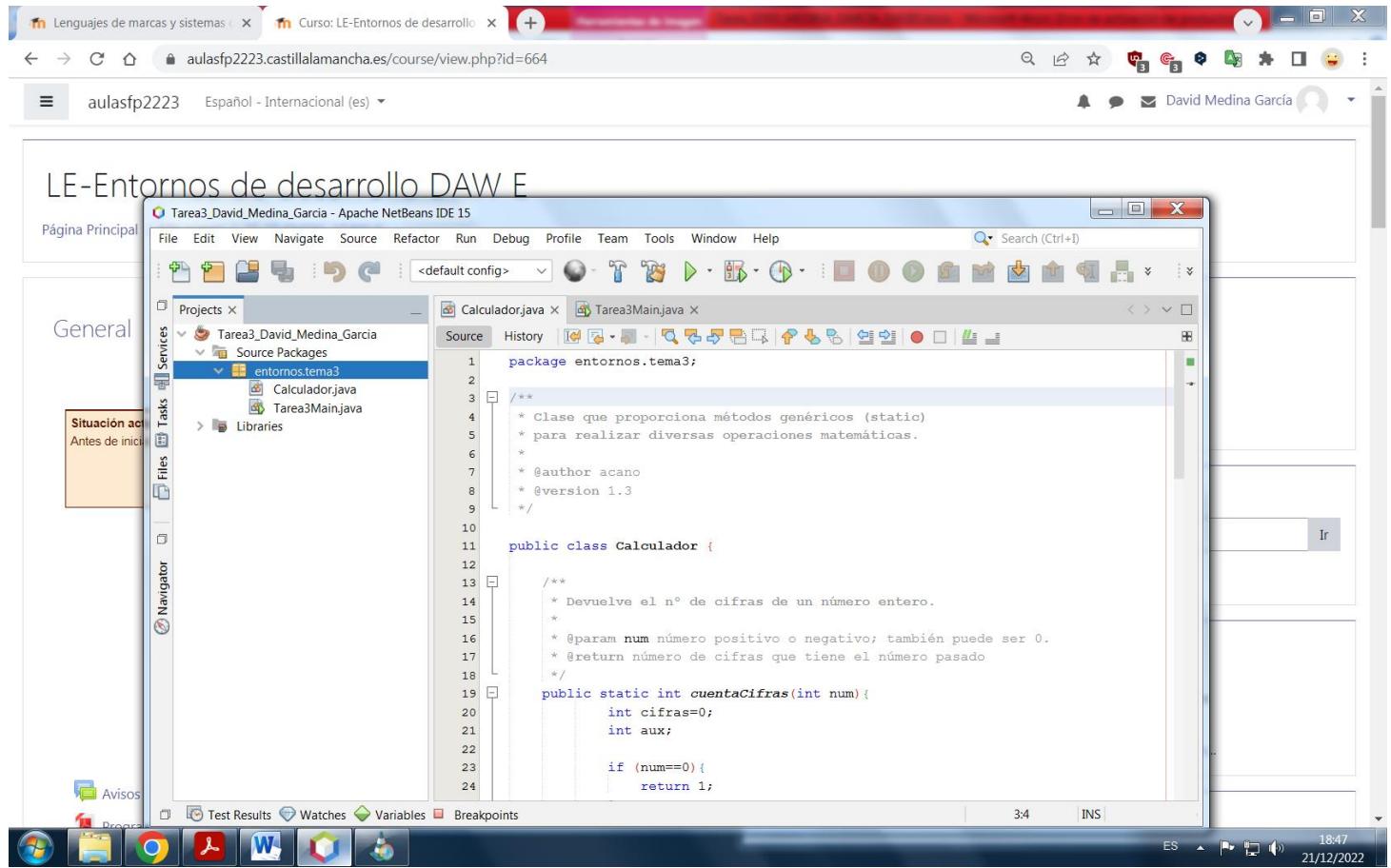
**1. Crear un proyecto en Apache NetBeans cuyo nombre sea "Tarea3\_TuNombre" e incorpora las dos clases proporcionadas en el paquete entornos.tema3 (renombra el paquete que viene en las clases que te he pasado, para que funcione)**

Hacemos click en la pestaña **File > New Project**. Dentro de la carpeta **Java Ant** seleccionamos **Java Application**.

Introducimos el nombre del proyecto y donde se va a ubicar.



Importamos las clases aportadas por el profesor al paquete creado por defecto y cambiamos el nombre del paquete con "**Refactor**" para que se aplique en todos los elementos.



## 2. Compilar y ejecutar la aplicación, viendo la salida que da por consola.

- Realiza dos o tres ejecuciones, en las que pruebas diferentes números para ver que se cuentan las cifras correctamente, y se realizan bien las potencias (la aplicación no controla excepciones, no puedes poner un número no entero ni mayor del máximo).

Ejecutamos el proyecto (**F6**) e introducimos números enteros cuando nos lo pida para ver que el código funciona correctamente. En las siguientes imágenes se ve que hemos probado con varios números, y efectivamente, no hay ningún error.

Curso: LE-Entornos de desarrollo Vista para moderadores de i

aulasfp2223.castillalamancha.es/course/view.php?id=664

aulasfp2223 Español - Internacional (es) David Medina García

## LE-Entornos de desarrollo DAW E

Página Principal / General

Situación actual: Antes de iniciar

Tarea3\_David\_Medina\_Garcia - Apache NetBeans IDE 15

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Output X Calculadora.java X Tarea3Main.java X

Debugger Console X Tarea3\_David\_Medina\_Garcia (run) X

```
run:
I N I C I O      D E      E J E C U C I O N
-----
--VAMOS A CONTAR CIFRAS--
Introduce el nñmero entero:
25
Nº dñgitos del valor 25 es 2

Ahora vamos a contar las cifras del
entero mñs grande que permite Java.

Nº dñgitos del valor 2147483647 es 10

--VAMOS A CALCULAR POTENCIAS--
Introduce el valor de la base:
5
Introduce el valor del exponente:
5
El resultado de 5 elevado a 5 = 3125

F I N      D E      E J E C U C I O N
-----
BUILD SUCCESSFUL (total time: 1 minute 12 seconds)
```

Tarea3\_David\_Medina\_Garcia (debug) running... X

Avisos Programas

Windows Taskbar: Internet Explorer, Google Chrome, Microsoft Word, Adobe Acrobat Reader, Java Development Kit

2015 20/12/2022

Curso: LE-Entornos de desarrollo Vista para moderadores de i

aulasfp2223.castillalamancha.es/course/view.php?id=664

aulasfp2223 Español - Internacional (es) David Medina García

## LE-Entornos de desarrollo DAW E

Página Principal / General

Situación actual: Antes de iniciar

Tarea3\_David\_Medina\_Garcia - Apache NetBeans IDE 15

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Output X Calculadora.java X Tarea3Main.java X

Debugger Console X Tarea3\_David\_Medina\_Garcia (run) X

```
run:
I N I C I O      D E      E J E C U C I O N
-----
--VAMOS A CONTAR CIFRAS--
Introduce el nñmero entero:
46
Nº dñgitos del valor 46 es 2

Ahora vamos a contar las cifras del
entero mñs grande que permite Java.

Nº dñgitos del valor 2147483647 es 10

--VAMOS A CALCULAR POTENCIAS--
Introduce el valor de la base:
3
Introduce el valor del exponente:
3
El resultado de 3 elevado a 3 = 27

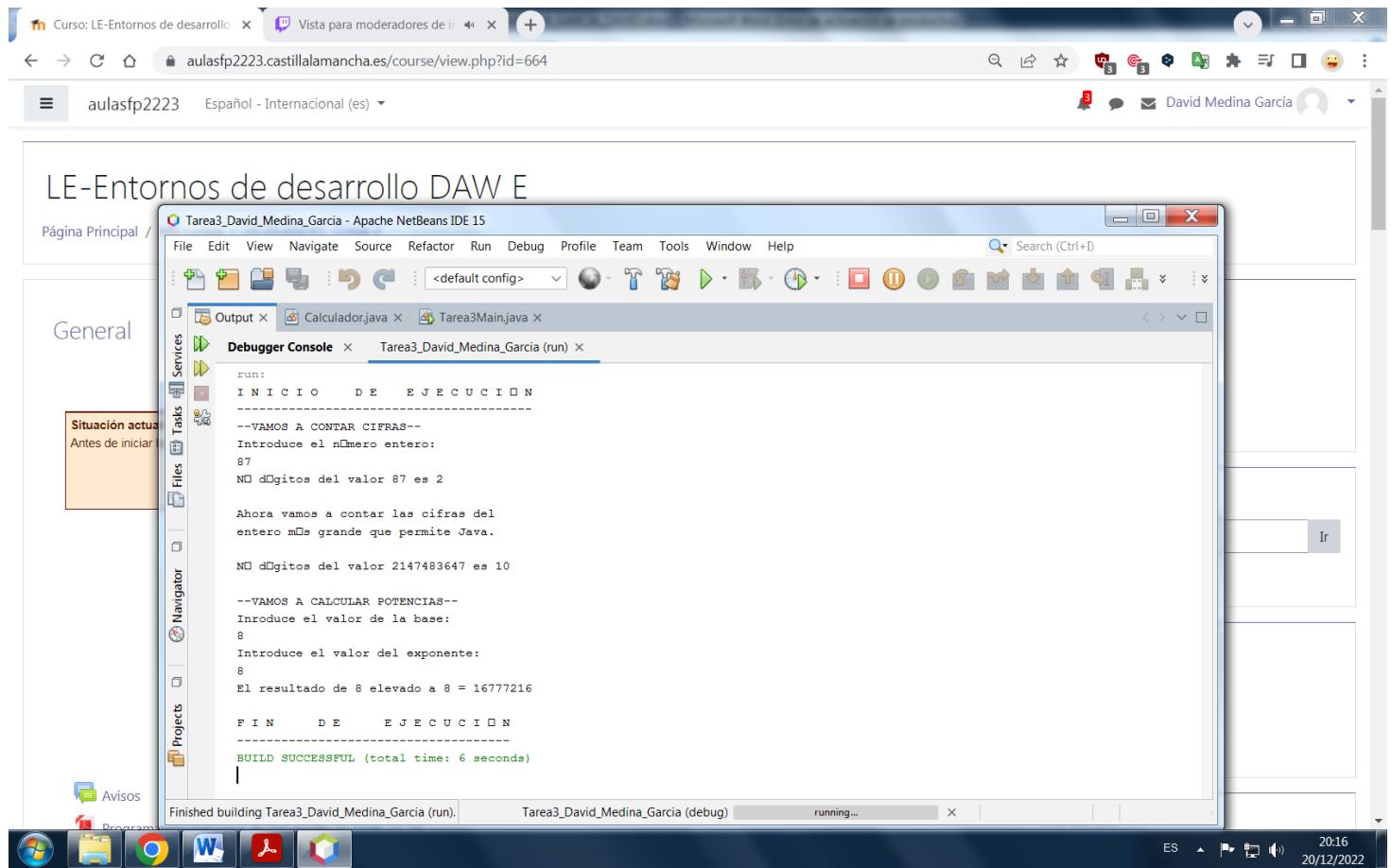
F I N      D E      E J E C U C I O N
-----
BUILD SUCCESSFUL (total time: 9 seconds)
```

Finished building Tarea3\_David\_Medina\_Garcia (run). Tarea3\_David\_Medina\_Garcia (debug) running... X

Avisos Programas

Windows Taskbar: Internet Explorer, Google Chrome, Microsoft Word, Adobe Acrobat Reader, Java Development Kit

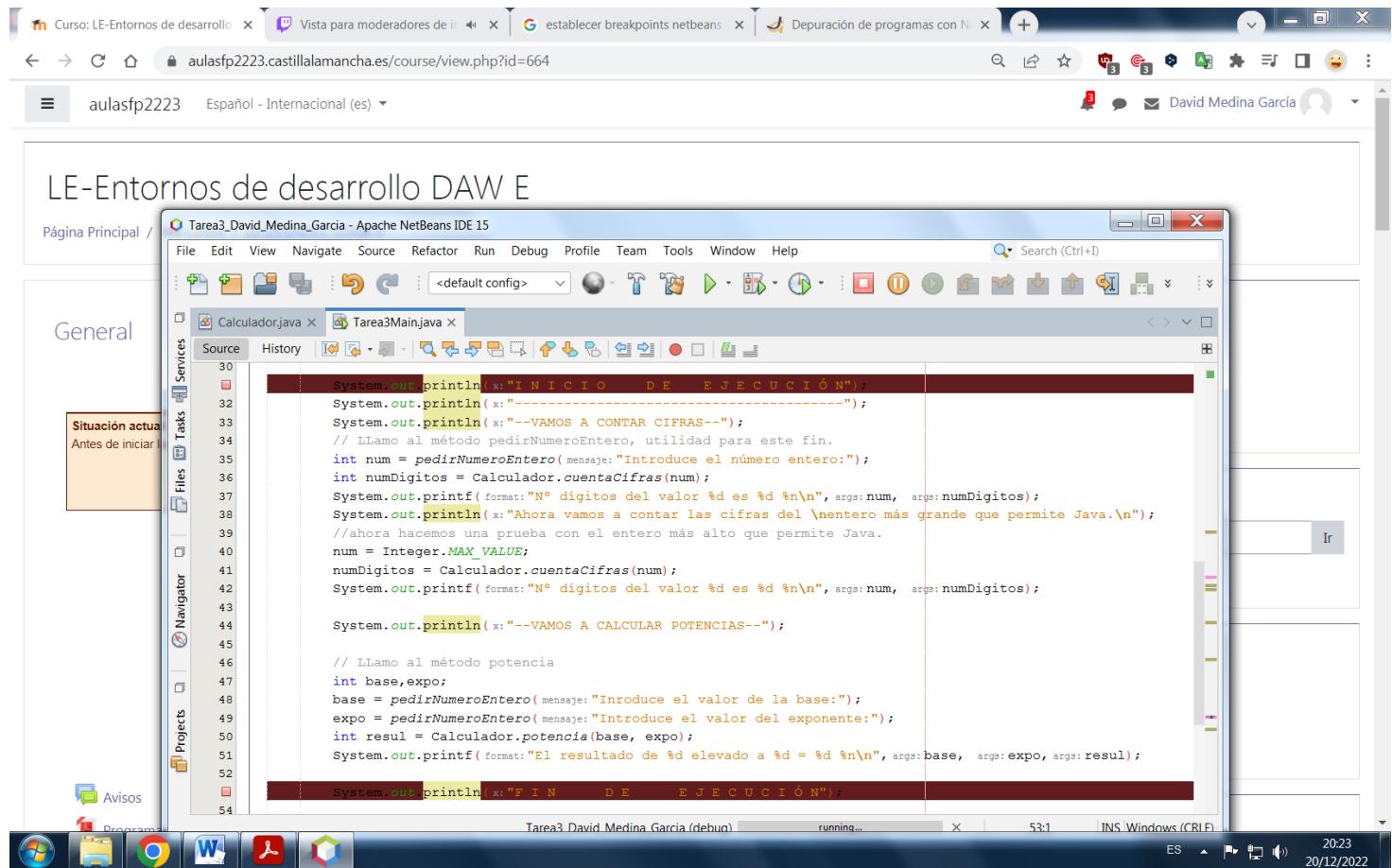
2016 20/12/2022



### 3. Establecer dos Breakpoints:

- uno en la línea 31 de Tarea3Main.java (la que pone “I N I C I O D E E J E C U C I Ó N”).
- otro en la línea 53 de Tarea3Main.java (la que pone “—F I N D E E J E C U C I Ó N—”).

Establecemos dos puntos de ruptura (Breakpoints) en las **líneas 31 y 53**. Para ello **pulsamos sobre el número** que nos indica la línea que corresponde.



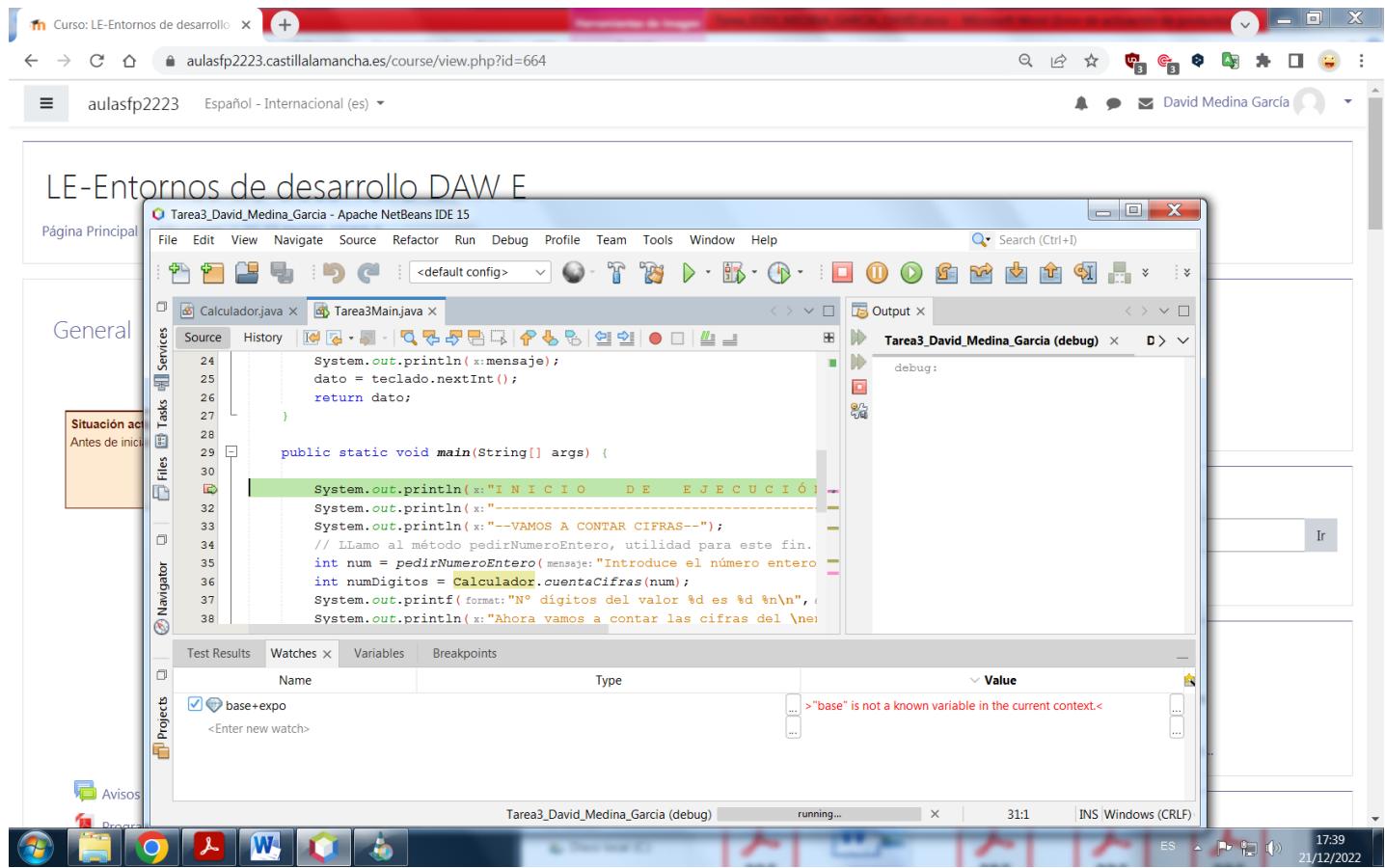
#### 4. Iniciar una sesión con el depurador para ejecutar paso a paso la aplicación.

- c) La ejecución se detendrá en el primer Breakpoints (línea 31).
- d) Una vez se empieza a depurar, se tienen que tener visibles las ventanas de depuración (*Variables* y *Watches* sobre todo).
- e) Añade a *Watches*, la siguiente expresión: `base+expo` (no es que tenga sentido sumar base y exponente, pero más adelante se utilizará para un ejemplo)

Tras ejecutar la depuración del código (**ctrl+F5**) comprobamos que se detiene en el primer **Breakpoints**.

Hacemos visibles las ventanas que nos interesen del modo debug en la pestaña **window > debugging**. En este caso he puesto las ventanas **Variables**, **Watches** y **Breakpoints**.

En la ventana **Watches** añadimos la expresión **base+expo** haciendo **doble click** donde pone "**<Enter new watch>**".

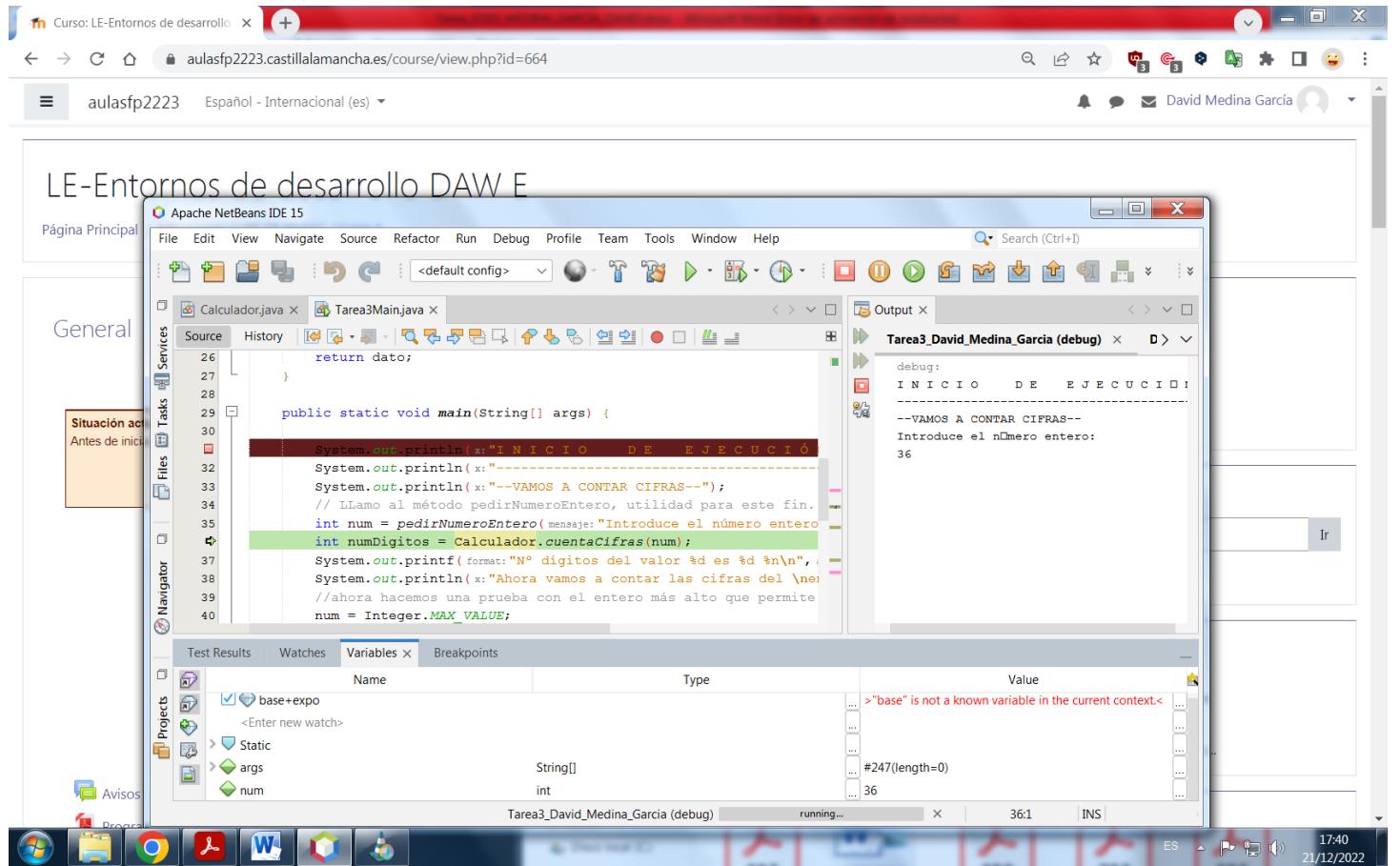


f) Avanza paso a paso hasta la línea 36 (la llamada al método **cuentaCifras**). En uno de los pasos, habrás tenido que introducir un número por teclado.

g) Inspecciona las variables en la ventana Variables (**num** tiene que tener el número introducido).

Con **F8** avanzamos en la depuración paso a paso hasta la **línea 36** donde nos piden introducir un número. Tras poner el número pulsamos **enter** para seguir con la depuración.

Comprobamos en la ventana variable que nuestro número ha sido correctamente introducido.



h) Avanza paso a paso sin entrar en el método hasta la línea 38.

i) Verás que el valor de la variable **numDigitos** ha cambiado, ya que se ha ejecutado el método que ha contado las cifras y que van apareciendo poco a poco mensajes por consola.

Si seguimos avanzando paso a paso (**F8**) hasta la **línea 38** comprobamos que la variable **numDigitos** ha cambiado de valor al ejecutarse el método (**cuentacifras**).

Este cambio se verá reflejado en la consola.

The screenshot shows the Apache NetBeans IDE interface. In the center, there's a Source editor window titled "Tarea3\_David\_Medina\_Garcia - Apache NetBeans IDE". It contains Java code for a program named "Calculador.java". The code includes a main method that prints a welcome message, prompts for a number, and then prints the number of digits of the entered number. The Output window to the right shows the execution of the program, starting with "INICIO DE EJECUCION", prompting for a number ("VAMOS A CONTAR CIFRAS-- Introduce el número entero:"), receiving the input "36", and then printing "NO dígitos del valor 36 es 2". Below the Source editor, the Variables window shows the current values of variables: args (String[], value: "#247(length=0)"), num (int, value: 36), and numDigitos (int, value: 2).

j) Avanza hasta la línea 41. El valor de la variable **num** ha cambiado (tiene que tener un valor muy grande, el correspondiente a **Integer.MAX\_VALUE**).

Continuando hasta la **línea 41** comprobamos que el valor de **num** ha cambiado porque hemos pasado por el código **num = Integer.MAX\_VALUE**.

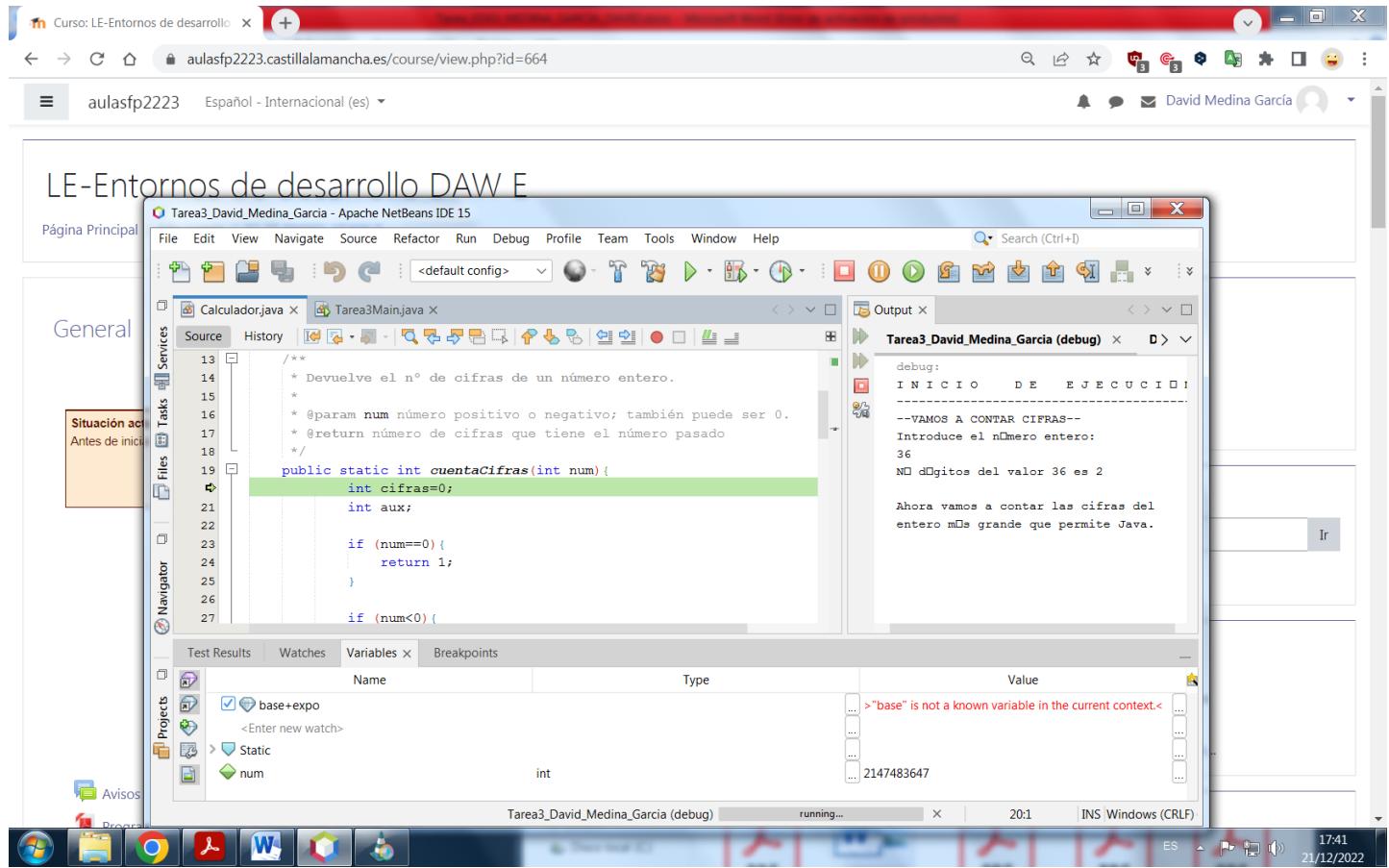
The screenshot shows the NetBeans IDE interface with the following details:

- Title Bar:** Curso: LE-Entornos de desarrollo > aulasfp2223.castillalamancha.es/course/view.php?id=664
- Toolbar:** File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Source Tab:** Displays Java code for 'Tarea3Main.java' and 'Calculador.java'. The code includes printing 'INICIO DE EJECUCIÓN', prompting for an integer, counting digits, and calculating powers.
- Output Tab:** Shows the execution log with messages like 'INICIO DE EJECUCIÓN', 'VAMOS A CONTAR CIFRAS...', and the digit count for the number 36.
- Variables Tab:** Shows the variable 'num' with a value of 2147483647, which is highlighted with a red box.
- Bottom Status Bar:** Shows the project name 'Tarea3\_David\_Medina\_Garcia (debug)', the state 'running...', the line number '41:1', and the file 'INS Windows (CRLF)'.

k) Ahora avanza paso a paso entrando en el método **cuentaCifras**.

l) Verás cómo ha cambiado el contexto de las variables. (Ahora se ven las del método en el que estás).

Para entrar en el método mientras avanzamos paso a paso pulsamos **F7**. Comprobaremos que hemos cambiado a la clase **Calculador**, pues es donde se encuentra el método. También observamos que ahora se nos muestran las variables de este método.



m) Avanza paso a paso dentro del método hasta que hayas ejecutado el bucle while todas las veces. Verás cómo cambió la variable **num**, y en cada iteración del while, van cambiando el valor de la variable **cifras** en el que se van contando las cifras del número y el de la variable **aux**.

Continuamos paso a paso (**F8**) para ejecutar el bucle while entero. Mientras lo hacemos observamos que el valor de las variables **cifras** y **aux** van cambiando, aumentando el valor de **cifras** hasta **10** y disminuyendo el de **aux** hasta **0**.

Curso: LE-Entornos de desarrollo x +

aulasfp2223.castillalamancha.es/course/view.php?id=664

aulasfp2223 Español - Internacional (es) David Medina García

## LE-Entornos de desarrollo DAW E

Página Principal General Situación actual Antes de iniciar Avisos

Tarea3\_David\_Medina\_Garcia - Apache NetBeans IDE 15

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Source History <default config> Tarea3Mainjava X

```
28 // Si es negativo, obtengo el valor absoluto.
29     aux = Math.abs( a:num );
30 }
31 else {
32     aux = num;
33 }
34
35 // Hago divisiones sucesivas entre 10
36 while (aux>0) {
37     aux = aux/10;
38     cifras++;
39 }
40
41 return cifras;
42
```

Output Tarea3\_David\_Medina\_Garcia (debug) D > v

```
debug:
I N I C I O      D E      E J E C U C I O N
--VAMOS A CONTAR CIFRAS--
Introduce el nñmero entero:
36
NO dígitos del valor 36 es 2

Ahora vamos a contar las cifras del
entero mñs grande que permite Java.
```

Test Results Watches Variables Breakpoints

Name	Type	Value
num	int	2147483647
cifras	int	0
aux	int	214748364

Thread main stopped at Calculador.java:38. Tarea3\_David\_Medina\_Garcia (debug) running... 38:1 INS Windows (CRLF)

17:42 21/12/2022

Name	Type	Value
num	int	2147483647
cifras	int	0
aux	int	214748364

Curso: LE-Entornos de desarrollo x +

aulasfp2223.castillalamancha.es/course/view.php?id=664

aulasfp2223 Español - Internacional (es) David Medina García

## LE-Entornos de desarrollo DAW E

Página Principal General Situación actual Antes de iniciar Avisos

Tarea3\_David\_Medina\_Garcia - Apache NetBeans IDE 15

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Source History <default config> Tarea3Mainjava X

```
28 // Si es negativo, obtengo el valor absoluto.
29     aux = Math.abs( a:num );
30 }
31 else {
32     aux = num;
33 }
34
35 // Hago divisiones sucesivas entre 10
36 while (aux>0) {
37     aux = aux/10;
38     cifras++;
39 }
40
41 return cifras;
42
```

Output Tarea3\_David\_Medina\_Garcia (debug) D > v

```
debug:
I N I C I O      D E      E J E C U C I O N
--VAMOS A CONTAR CIFRAS--
Introduce el nñmero entero:
36
NO dígitos del valor 36 es 2

Ahora vamos a contar las cifras del
entero mñs grande que permite Java.
```

Test Results Watches Variables Breakpoints

Name	Type	Value
num	int	2147483647
cifras	int	10
aux	int	0

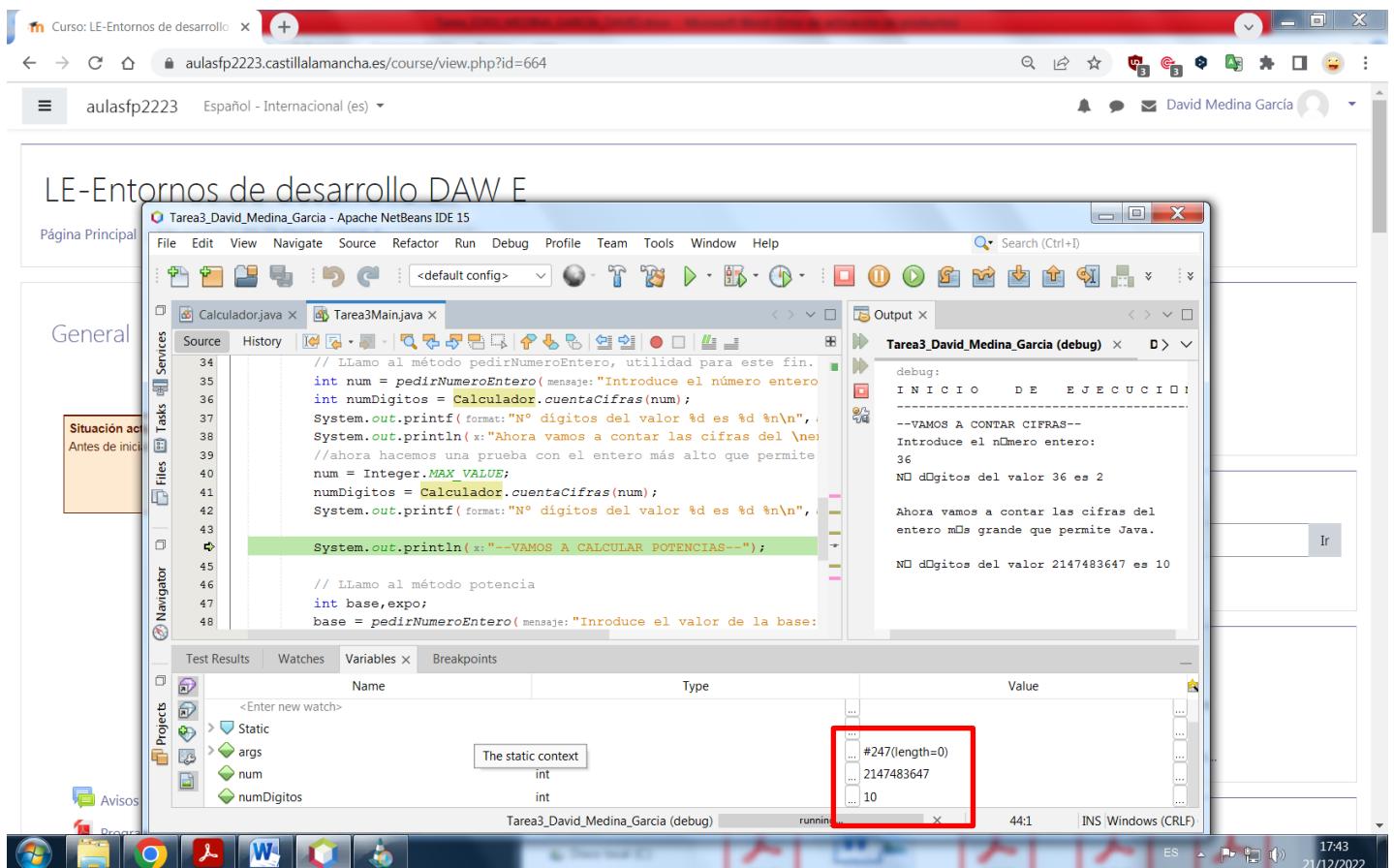
Thread main stopped at Calculador.java:38. Tarea3\_David\_Medina\_Garcia (debug) running... 36:1 INS Windows (CRLF)

17:42 21/12/2022

Name	Type	Value
num	int	2147483647
cifras	int	10
aux	int	0

**n)** Sigue paso a paso. Volverás al contexto del **main**. Avanza paso a paso hasta la línea 44, y verás cómo han aparecido mensajes por consola y las variables del contexto del **main**, ya se han actualizado con las que han obtenido del **cuentaCifras**.

Continuando paso a paso (**F8**) volvemos a la clase **Main** y llegamos a la línea 44. Comprobamos en la consola que los valores de **num** y **numDigitos** se han actualizado por el método **cuentaCifras**.



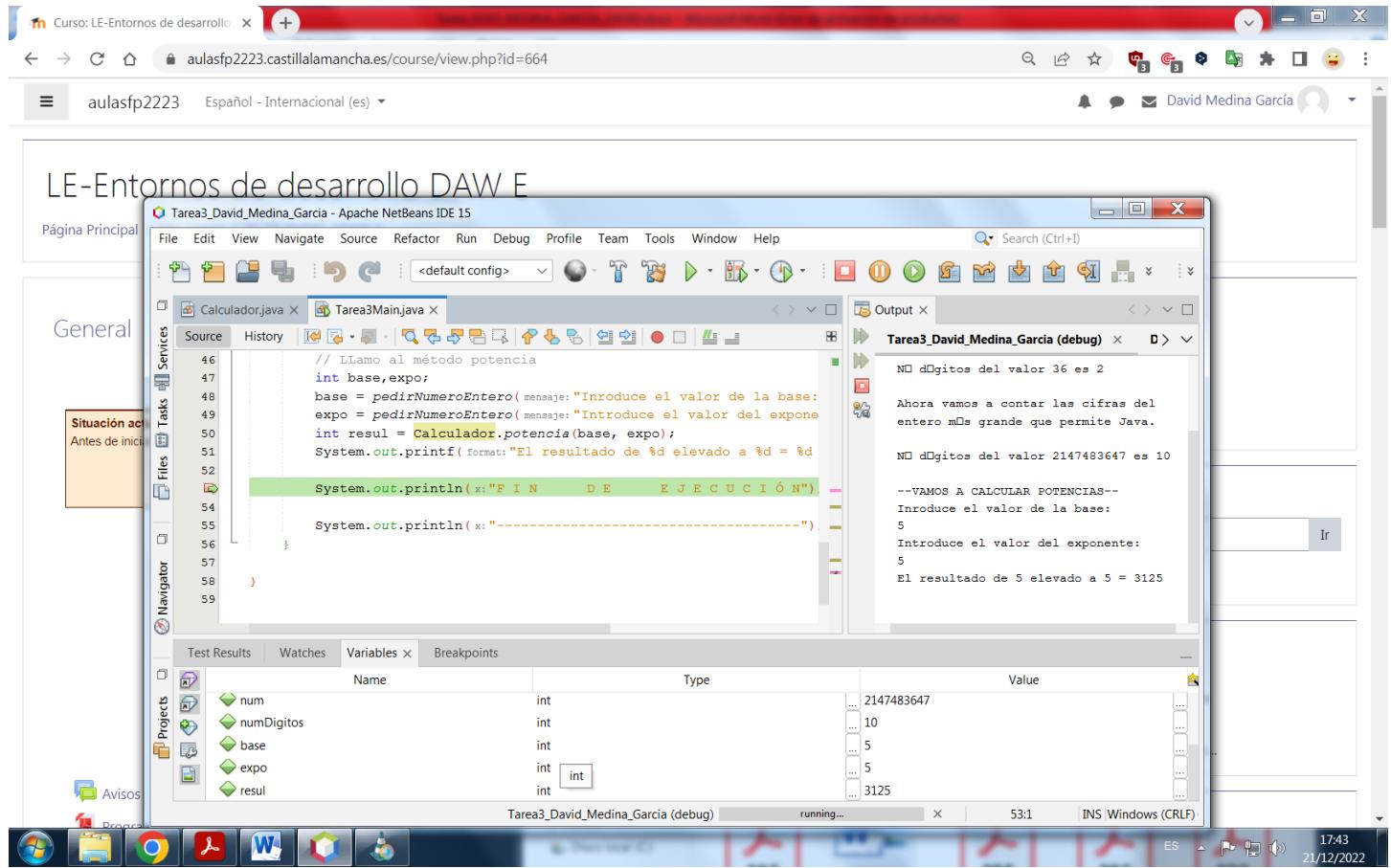
**o)** Reanuda la ejecución hasta llegar al 2º Breakpoints (ya no vayas paso a paso, pulsa en la opción **Continue** (**F5** o circulito verde con triángulo interior blanco)).

**p)** Verás que por consola se te pide el valor de la base y del exponente, y una vez los introduzcas continúa la ejecución, deteniéndose en el otro punto de ruptura que pusimos, no has tenido que ir ya paso a paso. (Eso muchas veces es útil, para avanzar en la ejecución si estás evaluando variables, pero no quieras salirte del todo de la depuración).

**q)** Entre tanto, se habrá calculado ya, el valor de la potencia y lo habrá mostrado por consola.

Continuamos la depuración (esta vez con **F5**, pues no iremos paso a paso). Por consola se nos pedirá que introduzcamos el valor de **base** y **exponente** para seguir con la depuración.

Una vez hecho esto se detendrá en el otro punto de ruptura (**Línea 53**) y observaremos en consola que se ha hecho el cálculo de la potencia.



r) Ahora cambiaremos el valor de las variables en tiempo de ejecución.

- No significa que estés cambiando el código, pero sí el valor en esa ejecución para hacer pruebas.
- Para ello observa el valor que tiene la expresión que añadimos a Watches (base + exponente). Ahora ve a la ventana Variables, y cambia el valor de la variable base.
- Verás ahora que el valor de la expresión en Watches ha cambiado.

En la ventana **variables** cambiamos el valor de la variable **base** y en la ventana **Watches** comprobamos como el resultado de la expresión que pusimos antes (**base+expo**) se ha actualizado.

Curso: LE-Entornos de desarrollo x +

aulasfp2223.castillalamancha.es/course/view.php?id=664

aulasfp2223 Español - Internacional (es) ▾

David Medina García

## LE-Entornos de desarrollo DAW E

Página Principal

General

Situación actual Antes de iniciar

Files Tasks Navigator Avisos Programas

Apache NetBeans IDE 15

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Calculator.java X Tarea3Main.java X

Source History Services

46 // Llamo al método potencia  
47 int base, expo;  
48 base = pedirNúmeroEntero( mensaje: "Introduce el valor de la base:  
49 expo = pedirNúmeroEntero( mensaje: "Introduce el valor del exponente:  
50 int resul = Calculador.potencia(base, expo);  
51 System.out.printf( formato: "El resultado de %d elevado a %d = %d  
52 System.out.println( x: "F I N D E E J E C U C I Ó N" );  
53 System.out.println( x: "-----" );  
54  
55 }  
56  
57 }

Output X

Tarea3\_David\_Medina\_Garcia (debug) x D > v

ND dígitos del valor 36 es 2  
Ahora vamos a contar las cifras del entero más grande que permite Java.  
ND dígitos del valor 2147483647 es 10  
--VAMOS A CALCULAR POTENCIAS--  
Introduce el valor de la base:  
5  
Introduce el valor del exponente:  
5  
El resultado de 5 elevado a 5 = 3125

Test Results Watches Variables Breakpoints

Name Type Value

base+expo int 10

Projects Avisos Programas

17:43 21/12/2022

The screenshot shows a Java application running in debug mode within the Apache NetBeans IDE. The code in the source editor calculates 5 to the power of 5. The 'Watches' tab in the debugger is active, displaying the variable 'base+expo' with a value of 10. The 'Output' tab shows the application's console output, which includes a message about digit counts and the result of the calculation.

Curso: LE-Entornos de desarrollo x +

aulasfp2223.castillalamancha.es/course/view.php?id=664

aulasfp2223 Español - Internacional (es) ▾

David Medina García

## LE-Entornos de desarrollo DAW E

Página Principal

General

Situación actual Antes de iniciar

Files Tasks Navigator Avisos Programas

Apache NetBeans IDE 15

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Calculator.java X Tarea3Main.java X

Source History Services

46 // Llamo al método potencia  
47 int base, expo;  
48 base = pedirNúmeroEntero( mensaje: "Introduce el valor de la base:  
49 expo = pedirNúmeroEntero( mensaje: "Introduce el valor del exponente:  
50 int resul = Calculador.potencia(base, expo);  
51 System.out.printf( formato: "El resultado de %d elevado a %d = %d  
52 System.out.println( x: "F I N D E E J E C U C I Ó N" );  
53 System.out.println( x: "-----" );  
54  
55 }  
56  
57 }

Output X

Tarea3\_David\_Medina\_Garcia (debug) x D > v

ND dígitos del valor 36 es 2  
Ahora vamos a contar las cifras del entero más grande que permite Java.  
ND dígitos del valor 2147483647 es 10  
--VAMOS A CALCULAR POTENCIAS--  
Introduce el valor de la base:  
5  
Introduce el valor del exponente:  
5  
El resultado de 5 elevado a 5 = 3125

Test Results Watches Variables Breakpoints

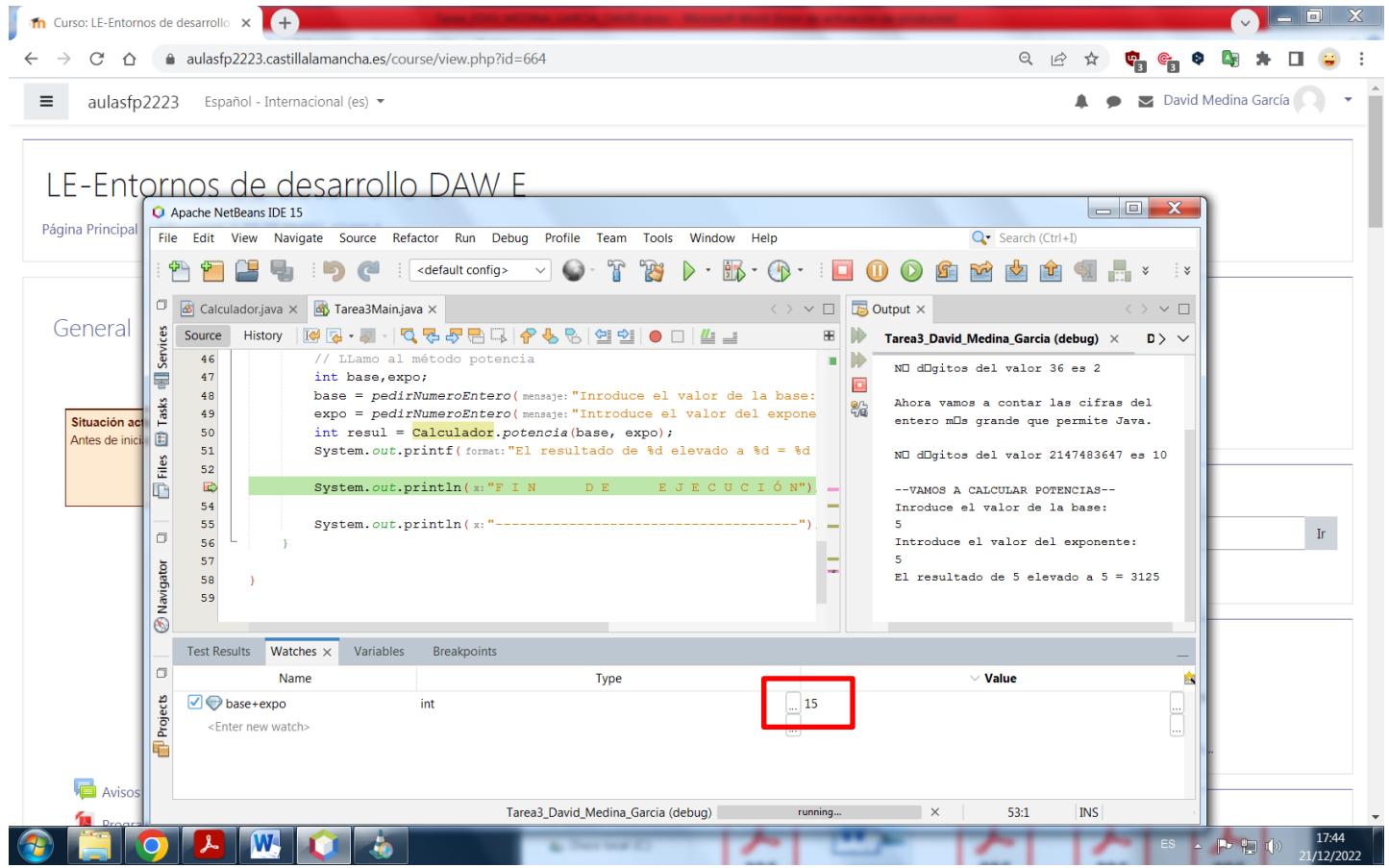
Name Type Value

num int 2147483647  
numDigitos int 10  
base int 10  
expo int 5  
resul int 3125

Projects Avisos Programas

17:44 21/12/2022

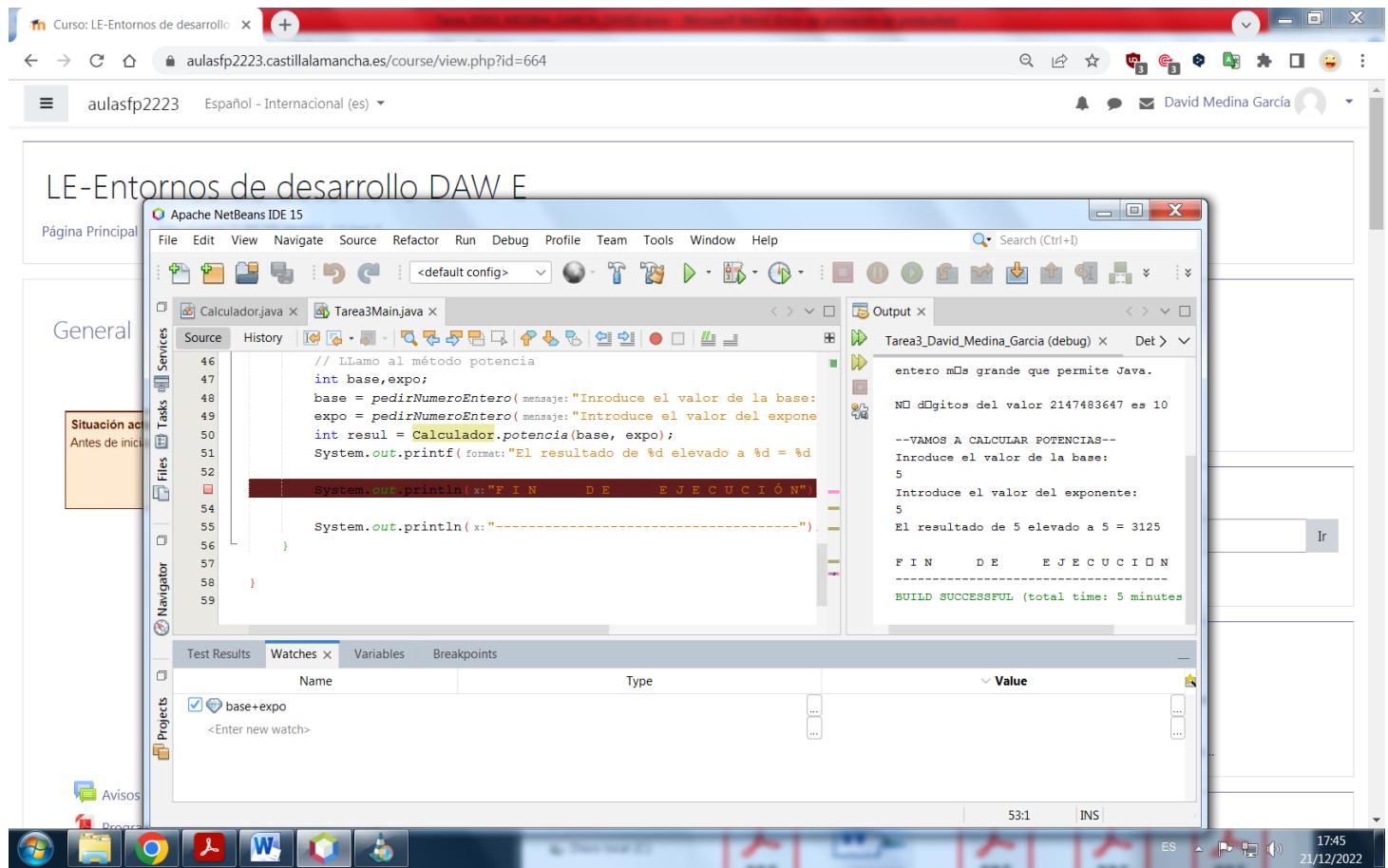
The screenshot shows a Java application running in debug mode within the Apache NetBeans IDE. The code in the source editor calculates 5 to the power of 5. The 'Variables' tab in the debugger is active, displaying local variables: 'num' (2147483647), 'numDigitos' (10), 'base' (10), 'expo' (5), and 'resul' (3125). The 'Output' tab shows the application's console output, which includes a message about digit counts and the result of the calculation.



s) Ya estamos llegando al final. Para terminar una depuración, podrías seguir paso a paso, hasta que termine, darle a continuar, o directamente darle a finalizar la depuración (cuadradito rojo con cuadradito blanco interno). Según sea el caso, es más útil hacer una u otra cosa, dependiendo de si queremos seguir inspeccionando, o si ya no nos interesa en cuyo caso lo mejor es finalizar.

t) Finaliza la sesión de depuración de la manera que deseas de las explicadas anteriormente.

Para terminar finalizamos la depuración de la manera que consideremos más óptima. En este caso como ya estábamos al final del código y solo quedaban un par de líneas daría igual el método. Yo he elegido la opción **continue (F5)**.



5. Crear una Aplicación JUnit para la clase Calculador. Utiliza la estrategia de **caja blanca** para diseñar los casos de prueba. (Para este ejercicio, además de los CONTENIDOS interactivos de la plataforma, conviene estudiar bien el anexo que se ha puesto en el apartado CONTENIDOS de la unidad)

#### Piensa en casos de prueba significativos.

- Escribe **cuatro casos de prueba** para el método **cuentaCifras**. En el que pruebes distintas cosas (rangos, negativos, cero etc.)
- Escribe **tres casos de prueba** para el método **potencia**. (Igualmente que haya variedad).
- Ejecuta la aplicación de prueba**, visualizando la consola y el **Panel de Resultados**. No deberían aparecer defectos en la implementación de la clase **Calculador**. (Todo **verde**)

## PRIMER CASO CUENTACIFRAS (NUMERO ENTERO)

The screenshot shows the Apache NetBeans IDE 15 interface. The top bar displays the course "LE-Entornos de desarrollo" and the URL "aulasfp2223.castillalamancha.es/course/view.php?id=664". The status bar at the bottom right shows "16:56 05/01/2023".

The main window title is "LE-Entornos de desarrollo DAW F". The left sidebar includes "Projects", "Files", "Navigator", and "Services". The central workspace contains three tabs: "Calculadora.java", "CalculadoraTest.java", and "Tarea3Main.java". The "CalculadoraTest.java" tab is active, showing the following code:

```
10  /*
11  * @Test
12  * public void testCuentaCifras() {
13  *
14  *     System.out.println("Primer caso cuentaCifras.");
15  *     int num = 5787;
16  *     int expResult = 4;
17  *     int result = Calculadora.cuentaCifras(num);
18  *     assertEquals(expected:expResult, actual:result);
19  *     if (result!=expResult) {
20  *         fail(message:"El test ha fallado");
21  *     }
22  *
23  *     System.out.println("Segundo caso cuentaCifras.");
24  *     num = -875330;
```

The "Output" panel on the right shows the test results:

```
Testsuite: entornos.tema3.CalculadoraTest
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0,1
----- Standard Output -----
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
-----
test:
Deleting: C:\Users\USUARIO\AppData\Local\Temp\TEST-entornos.tema3.C
BUILD SUCCESSFUL (total time: 0 seconds)
```

The "Test Results" panel shows "Tests passed: 100,00 %" and "Both tests passed. (0,109 s)".

## SEGUNDO CASO CUENTACIFRAS (NUMERO NEGATIVO)

The screenshot shows the Apache NetBeans IDE 15 interface, identical to the previous one but with different code in the "CalculadoraTest.java" tab.

The "CalculadoraTest.java" tab contains the following code:

```
20  fail(message:"El test ha fallado");
21
22
23  System.out.println("Segundo caso cuentaCifras.");
24  num = -875330;
25  expResult = 6;
26  result = Calculadora.cuentaCifras(num);
27  assertEquals(expected:expResult, actual:result);
28  if (result!=expResult) {
29      fail(message:"El test ha fallado");
30  }
31
32  System.out.println("Tercer caso cuentaCifras.");
```

The "Output" panel shows the test results:

```
Testsuite: entornos.tema3.CalculadoraTest
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0,1
----- Standard Output -----
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
-----
test:
Deleting: C:\Users\USUARIO\AppData\Local\Temp\TEST-entornos.tema3.C
BUILD SUCCESSFUL (total time: 0 seconds)
```

The "Test Results" panel shows "Tests passed: 100,00 %" and "Both tests passed. (0,109 s)".

### TERCER CASO CUENTACIFRAS (NUMERO 0)

The screenshot shows the Apache NetBeans IDE 15 interface. The top bar displays the course "Curso: LE-Entornos de desarrollo" and the URL "aulasfp2223.castillalamancha.es/course/view.php?id=664". The main window title is "LE-Entornos de desarrollo DAW E". The left sidebar includes "Files", "Tasks", "Services", "Navigator", and "Projects" sections. The central workspace contains three files: "Calculador.java", "CalculadorTest.java", and "Tarea3Main.java". The "CalculadorTest.java" file is open, showing code for testing digit counting. The "Output - Tarea3\_David\_Medina\_Garcia (test)" window shows the execution results for the third test case, which counts digits in the number 0. The "Test Results" window indicates 100.00% passed tests. The status bar at the bottom right shows the date and time as 05/01/2023 16:57.

```
    raii(message:"El test ha fallado");

    System.out.println(x:"Tercer caso cuentaCifras.");
    num = 0;
    expResult = 1;
    result = Calculador.cuentaCifras(num);
    assertEquals(expected:expResult, actual:result);
    if (result!=expResult) {
        fail(message:"El test ha fallado");
    }
}
```

```
Testsuite: entornos.tema3.CalculadorTest
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0,1
----- Standard Output -----
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
-----
test:
Deleting: C:\Users\USUARIO\AppData\Local\Temp\TEST-entornos.tema3.C
BUILD SUCCESSFUL (total time: 0 seconds)
```

### CUARTO CASO CUENTACIFRAS (NUMERO NEGATIVO 2 CIFRAS)

The screenshot shows the Apache NetBeans IDE 15 interface. The top bar displays the course "Curso: LE-Entornos de desarrollo" and the URL "aulasfp2223.castillalamancha.es/course/view.php?id=664". The main window title is "LE-Entornos de desarrollo DAW E". The left sidebar includes "Files", "Tasks", "Services", "Navigator", and "Projects" sections. The central workspace contains three files: "Calculador.java", "Tarea3Main.java", and "CalculadorTest.java". The "CalculadorTest.java" file is open, showing code for testing digit counting. The "Output - Tarea3\_David\_Medina\_Garcia (test)" window shows the execution results for the fourth test case, which counts digits in the number -875330. The "Test Results" window indicates 100.00% passed tests. The status bar at the bottom right shows the date and time as 15/01/2023 19:11.

```
    System.out.println(x:"Segundo caso cuentaCifras.");
    num = -875330;
    expResult = 6;
    result = Calculador.cuentaCifras(num);
    assertEquals(expected:expResult, actual:result);
    if (result!=expResult) {
        fail(message:"El test ha fallado");
    }

    System.out.println(x:"Tercer caso cuentaCifras.");
    num = 0;
    expResult = 1;
    result = Calculador.cuentaCifras(num);
```

```
Testsuite: entornos.tema3.CalculadorTest
Primer caso cuentaCifrasS.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0,091 sec
----- Standard Output -----
Primer caso cuentaCifrasS.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
-----
test:
Deleting: C:\Users\USUARIO\AppData\Local\Temp\TEST-entornos.tema3.CalculadorTest..
BUILD SUCCESSFUL (total time: 0 seconds)
```

## PRIMER CASO POTENCIA (EXPONENTE 0)

The screenshot shows the Apache NetBeans IDE 15 interface. The top navigation bar includes 'File', 'Edit', 'View', 'Navigate', 'Source', 'Refactor', 'Run', 'Debug', 'Profile', 'Team', 'Tools', 'Window', and 'Help'. The title bar says 'LE-Entornos de desarrollo DAW F' and 'aulasfp2223.castillalamancha.es/course/view.php?id=664'. The status bar at the bottom right shows '16:57 05/01/2023'. The main workspace contains several tabs: 'Calculador.java', 'CalculadorTest.java', and 'Tarea3Main.java'. The 'CalculadorTest.java' tab is active, displaying the following Java code:

```
56
57     @Test
58     public void testPotencia() {
59         System.out.println("Primer caso potencia.");
60         int base = 5;
61         int expo = 0;
62         int expResult = 1;
63         int result = Calculador.potencia(base, expo);
64         assertEquals(expected:expResult, actual:result);
65         if (result!=expResult) {
66             fail(message:"El test ha fallado.");
67         }
68         System.out.println("Segundo caso potencia.");
69     }
```

The 'Output' window shows the test results:

```
Testsuite: entornos.tema3.CalculadorTest
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0,1
----- Standard Output -----
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
----- 
test:
Deleting: C:\Users\USUARIO\AppData\Local\Temp\TEST-entornos.tema3.C
BUILD SUCCESSFUL (total time: 0 seconds)
```

The 'Test Results' window in the bottom left shows 'Tests passed: 100.00 %' and 'Both tests passed. (0.109 s) > entornos.tema3.CalculadorTest passed'.

## SEGUNDO CASO POTENCIA (EXPONENTE NEGATIVO)

The screenshot shows the Apache NetBeans IDE 15 interface. The top navigation bar includes 'File', 'Edit', 'View', 'Navigate', 'Source', 'Refactor', 'Run', 'Debug', 'Profile', 'Team', 'Tools', 'Window', and 'Help'. The title bar says 'LE-Entornos de desarrollo DAW F' and 'aulasfp2223.castillalamancha.es/course/view.php?id=664'. The status bar at the bottom right shows '16:57 05/01/2023'. The main workspace contains several tabs: 'Calculador.java', 'CalculadorTest.java', and 'Tarea3Main.java'. The 'CalculadorTest.java' tab is active, displaying the following Java code:

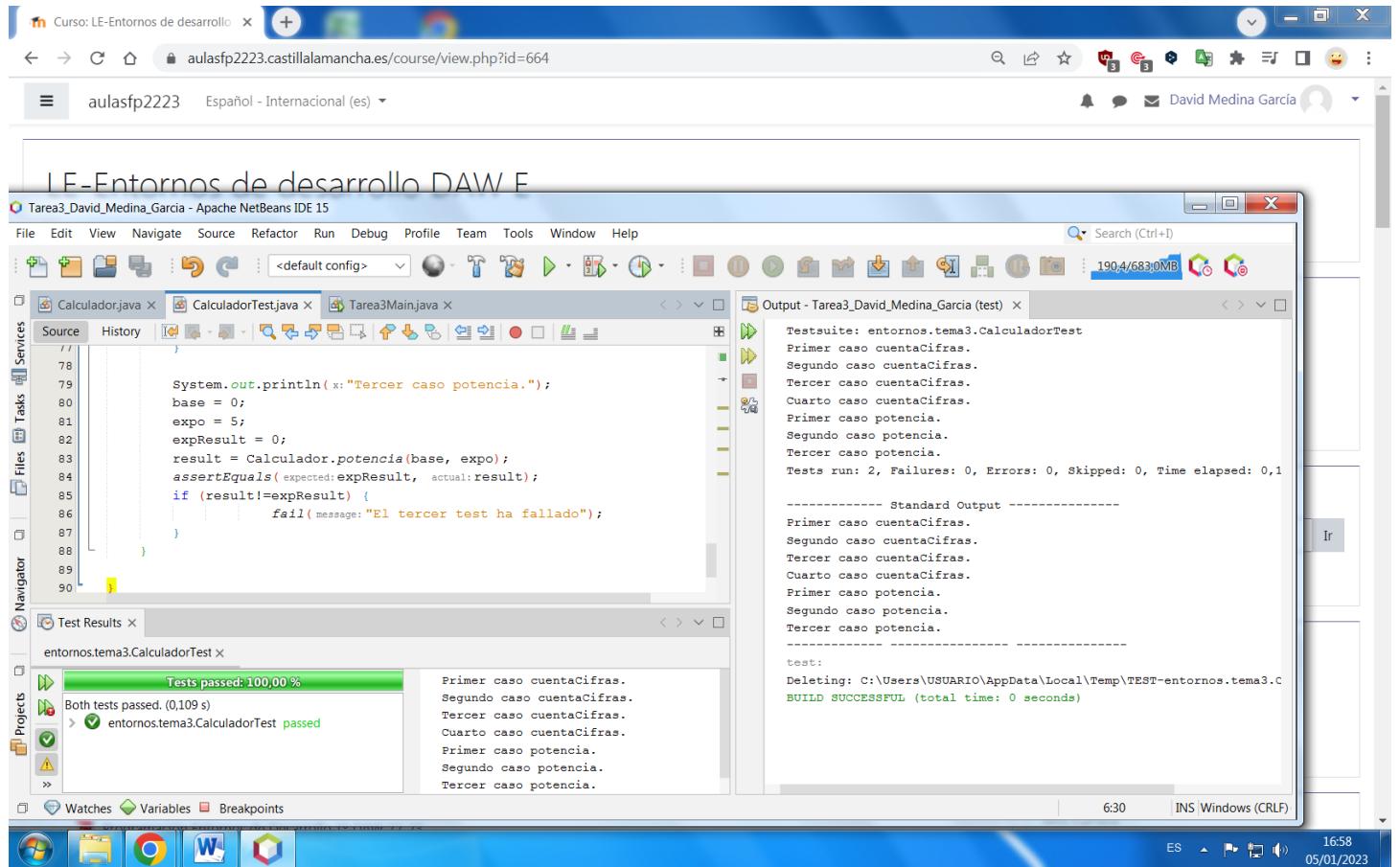
```
65
66     if (result!=expResult) {
67         fail(message:"El test ha fallado.");
68     }
69
70     System.out.println("Segundo caso potencia.");
71     base = 5;
72     expo = -10;
73     expResult = 1;
74     result = Calculador.potencia(base, expo);
75     assertEquals(expected:expResult, actual:result);
76     if (result!=expResult) {
77         fail(message:"El test ha fallado.");
78     }
```

The 'Output' window shows the test results:

```
Testsuite: entornos.tema3.CalculadorTest
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0,1
----- Standard Output -----
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Cuarto caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
----- 
test:
Deleting: C:\Users\USUARIO\AppData\Local\Temp\TEST-entornos.tema3.C
BUILD SUCCESSFUL (total time: 0 seconds)
```

The 'Test Results' window in the bottom left shows 'Tests passed: 100.00 %' and 'Both tests passed. (0.109 s) > entornos.tema3.CalculadorTest passed'.

### **TERCER CASO POTENCIA (BASE 0)**

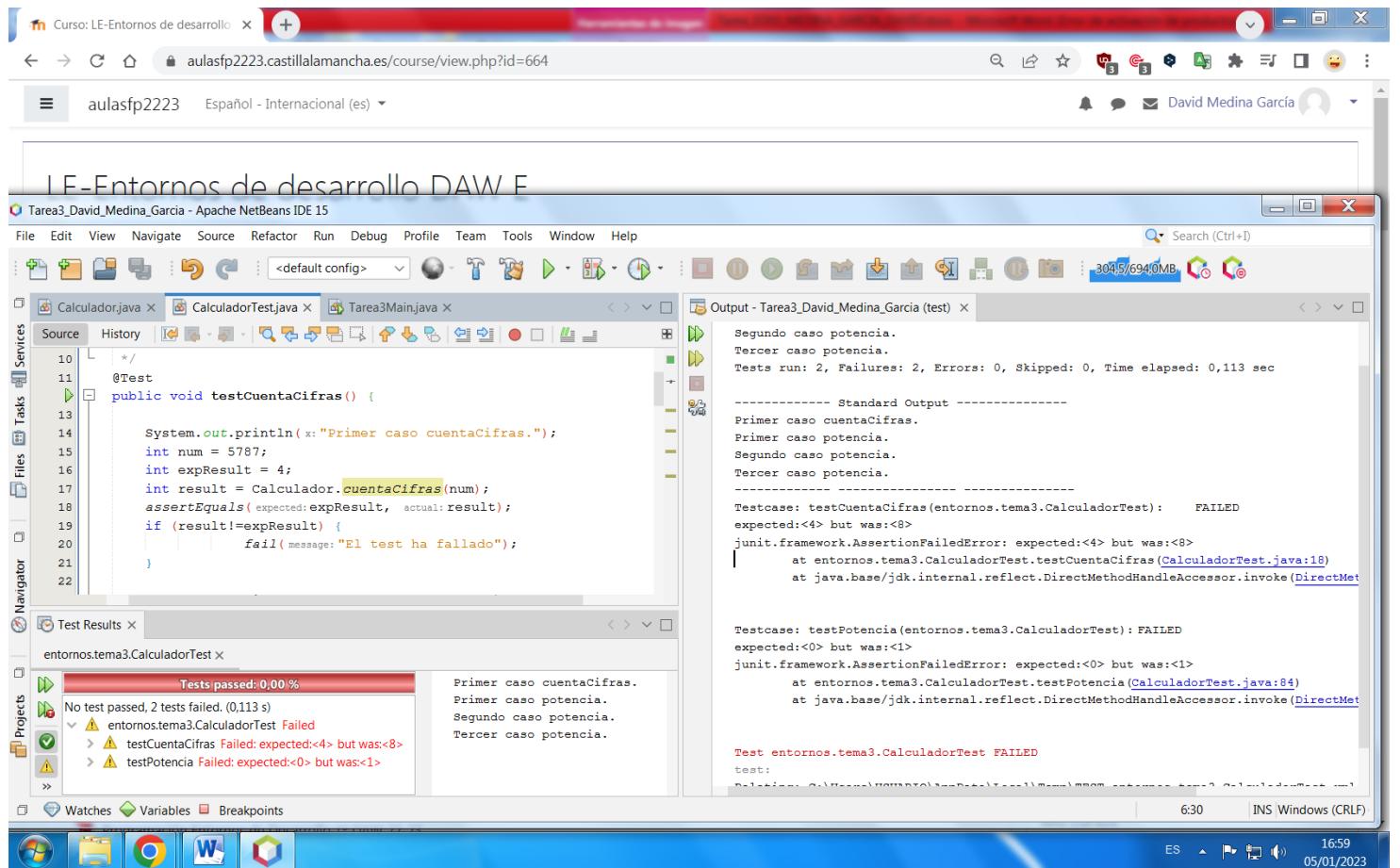


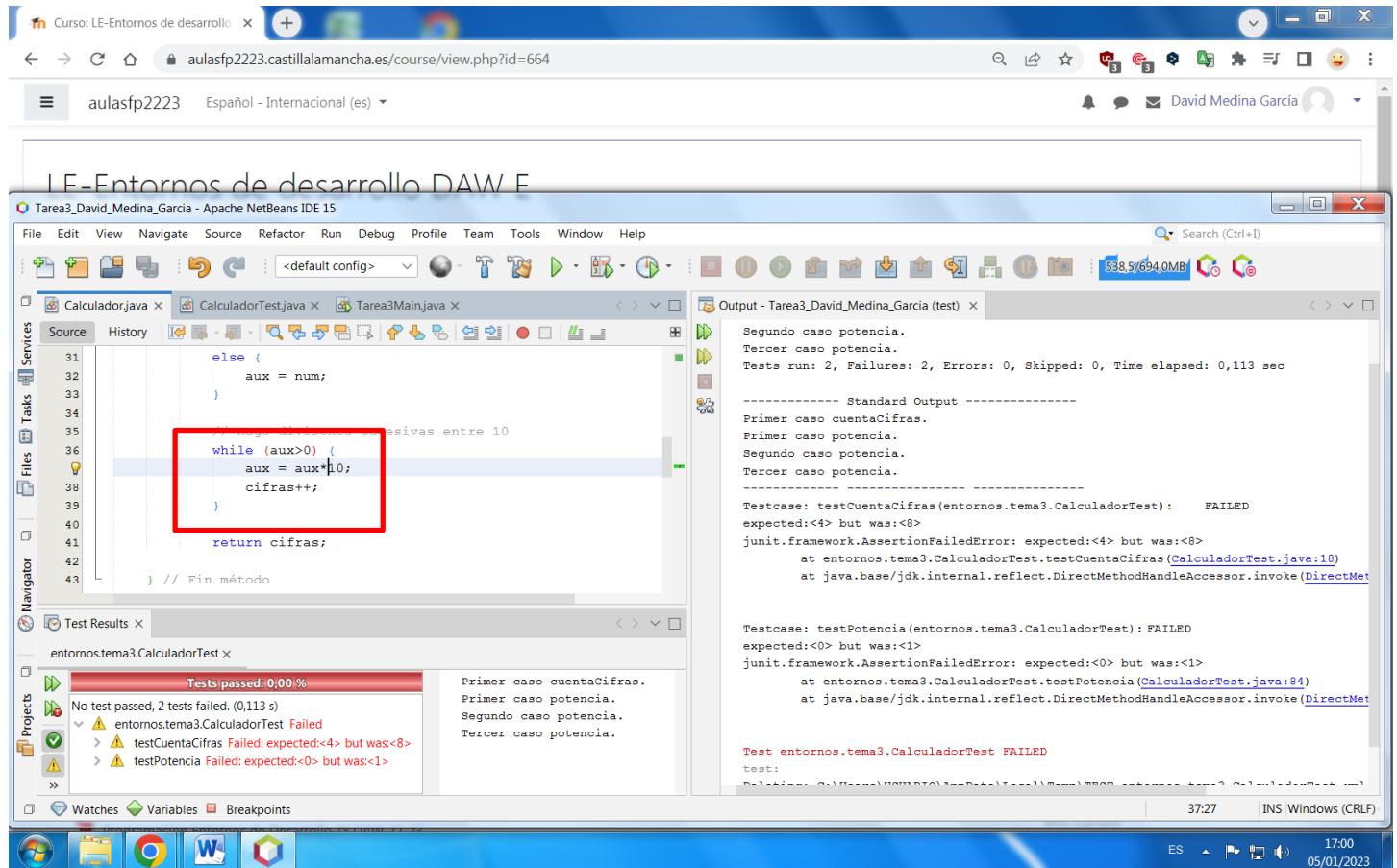
d) Ahora **fuerza algún error** en el método **cuentaCifras** y en el método **potencia** para que den mal el resultado.

e) **Ejecuta de nuevo los casos** de prueba en el que se vea que detecta el error. (Saldrá rojo e indica el mensaje si despliegas el panel del error)

## PRIMER CASO CUENTACIFRAS (NUMERO ENTERO)

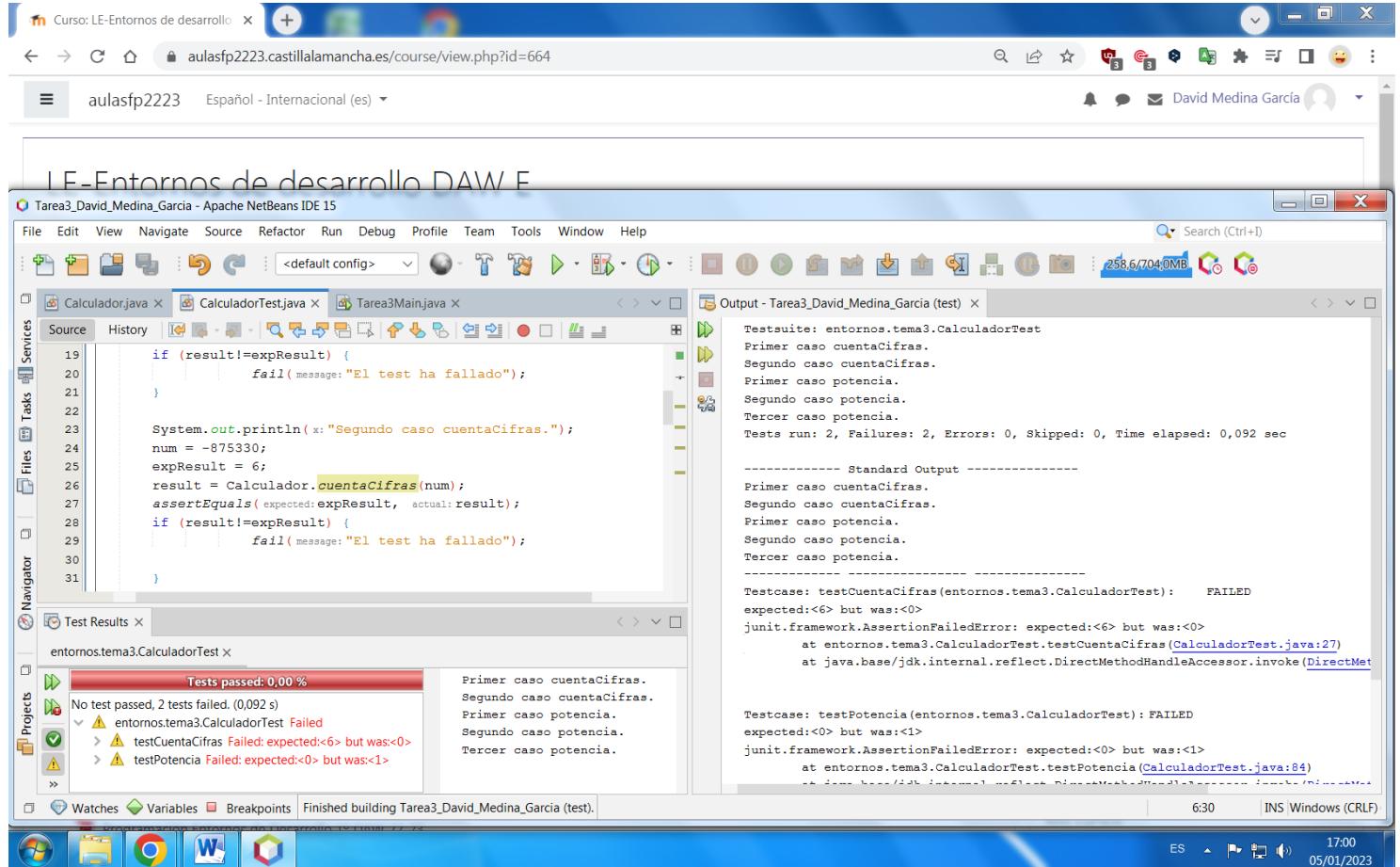
En este caso hemos forzado el error cambiando la división del **WHILE** por una multiplicación.

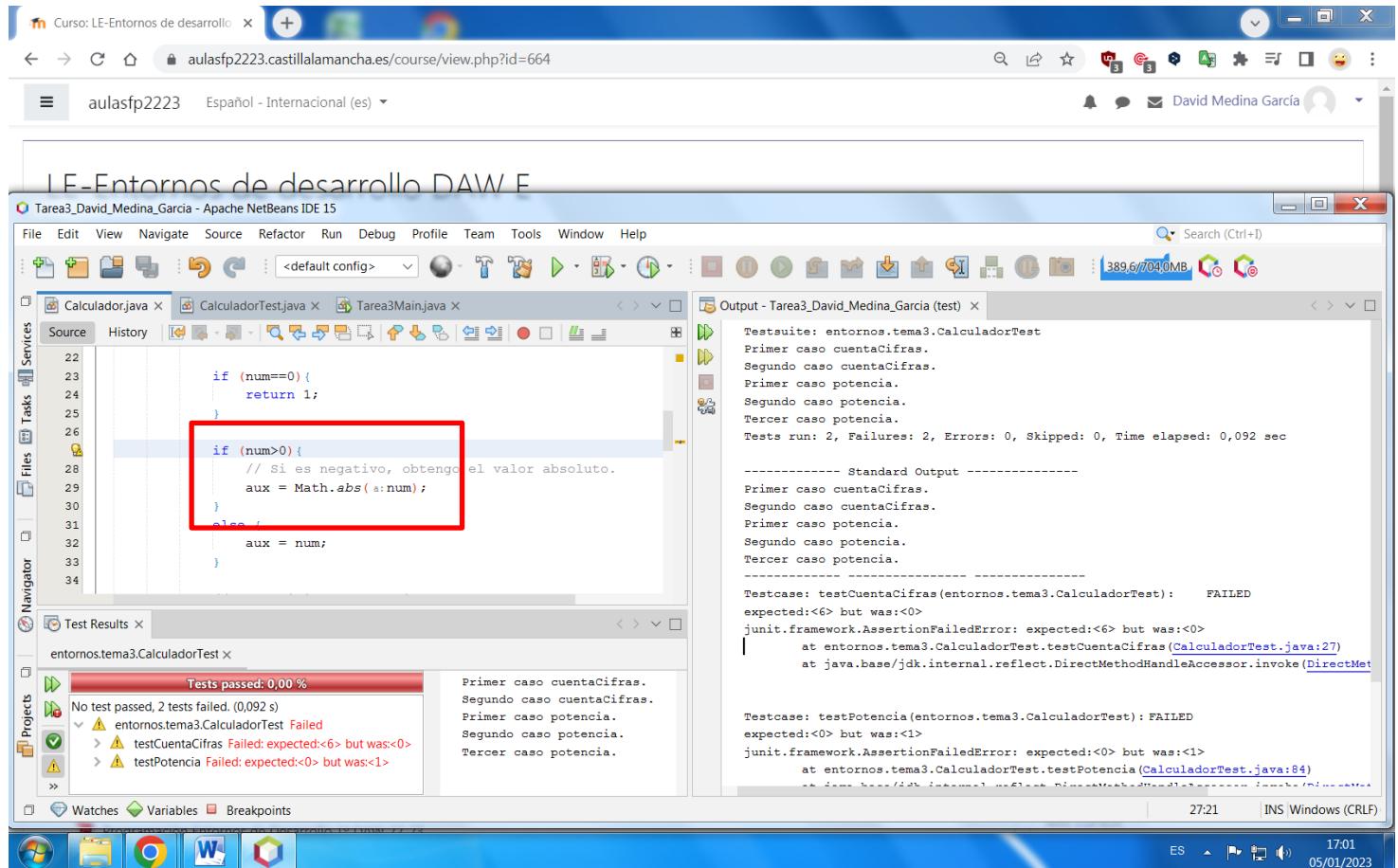




## SEGUNDO CASO CUENTACIFRAS (NUMERO NEGATIVO)

En este caso forzamos el error cambiando el sentido de comparación para obtener el valor absoluto de un numero negativo (**num<0 por num>0**).





## TERCER CASO CUENTACIFRAS (NUMERO 0)

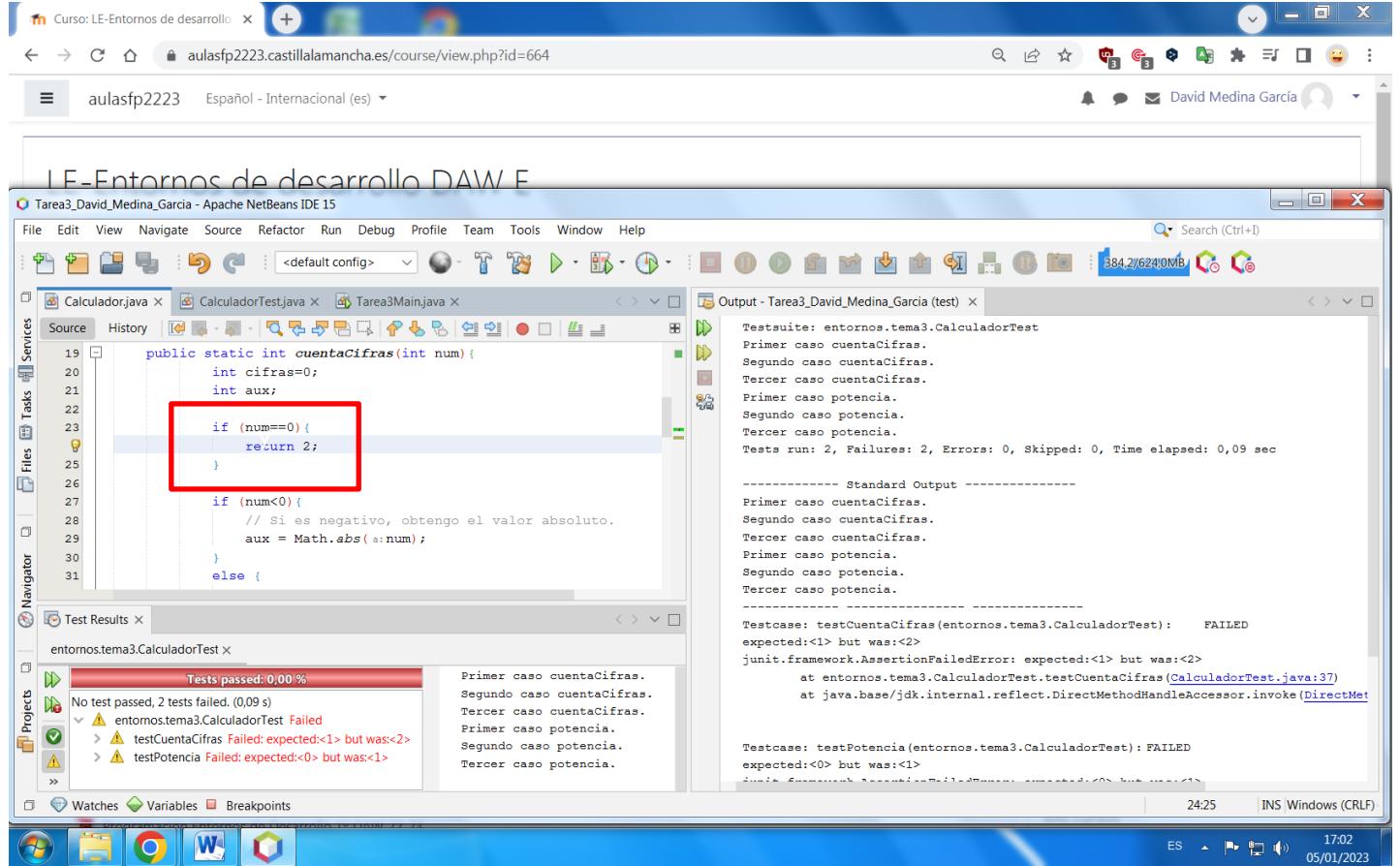
Aquí forzamos el error al cambiar el número que nos devolvería si introducimos el 0.

The screenshot shows the Apache NetBeans IDE 15 interface. At the top, a browser window displays the URL <aulasfp2223.castillalamancha.es/course/view.php?id=664>. The IDE window title is "LE-Entornos de desarrollo DAW F". The left sidebar includes "Services", "Tasks", "Files", and "Navigator". The main area contains three tabs: "Calculador.java", "CalculadorTest.java", and "Tarea3Main.java". The "CalculadorTest.java" tab is active, showing Java code for testing a calculator class. The code includes assertions for counting digits and calculating powers. The "Output - Tarea3\_David\_Medina\_Garcia (test)" tab shows the execution results of the tests. The "Test Results" tab shows that 2 tests failed. The status bar at the bottom indicates "6:30" and "17:02 05/01/2023".

```
31 }
32
33     System.out.println(x:"Tercer caso cuentaCifras.");
34     num = 0;
35     expResult = 1;
36     result = Calculador.cuentaCifras(num);
37     assertEquals(expected:expResult, actual:result);
38     if (result!=expResult) {
39         fail(message:"El test ha fallado");
40     }
41
42 }
43
44 System.out.println(x:"Cuarto caso cuentaCifras.");
```

```
Testsuite: entornos.tema3.CalculadorTest
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
Tests run: 2, Failures: 2, Errors: 0, Skipped: 0, Time elapsed: 0,09 sec
----- Standard Output -----
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
-----
Testcase: testCuentaCifras(entornos.tema3.CalculadorTest): FAILED
expected:<1> but was:<2>
junit.framework.AssertionFailedError: expected:<1> but was:<2>
|   at entornos.tema3.CalculadorTest.testCuentaCifras(CalculadorTest.java:37)
|   at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(DirectMet

Testcase: testPotencia(entornos.tema3.CalculadorTest): FAILED
expected:<0> but was:<1>
----- Standard Output -----
```

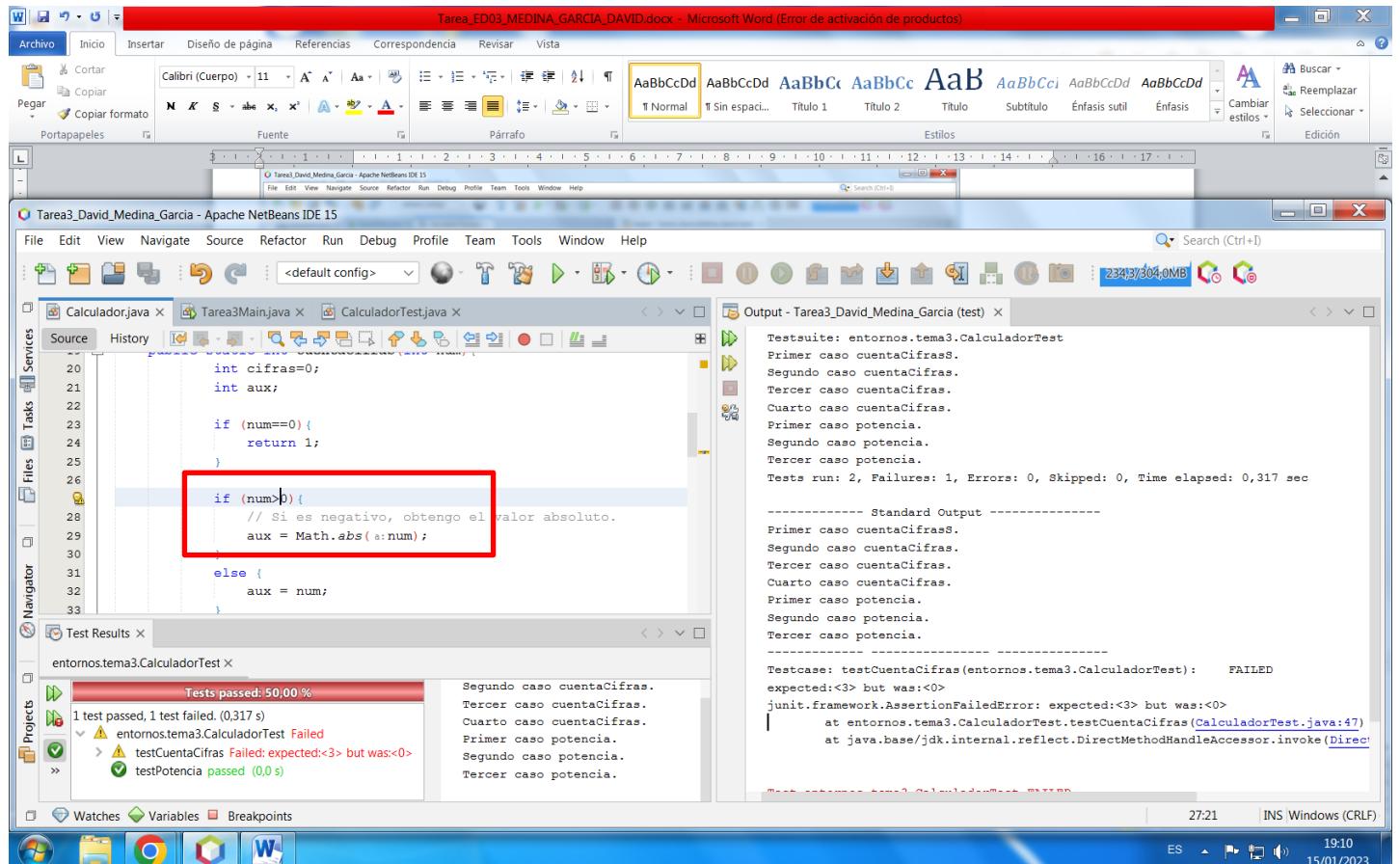


## CUARTO CASO CUENTACIFRAS (NUMERO NEGATIVO 2 CIFRAS)

En este caso forzamos el error cambiando el sentido de comparación para obtener el valor absoluto de un numero negativo (**num<0 por num>0**).

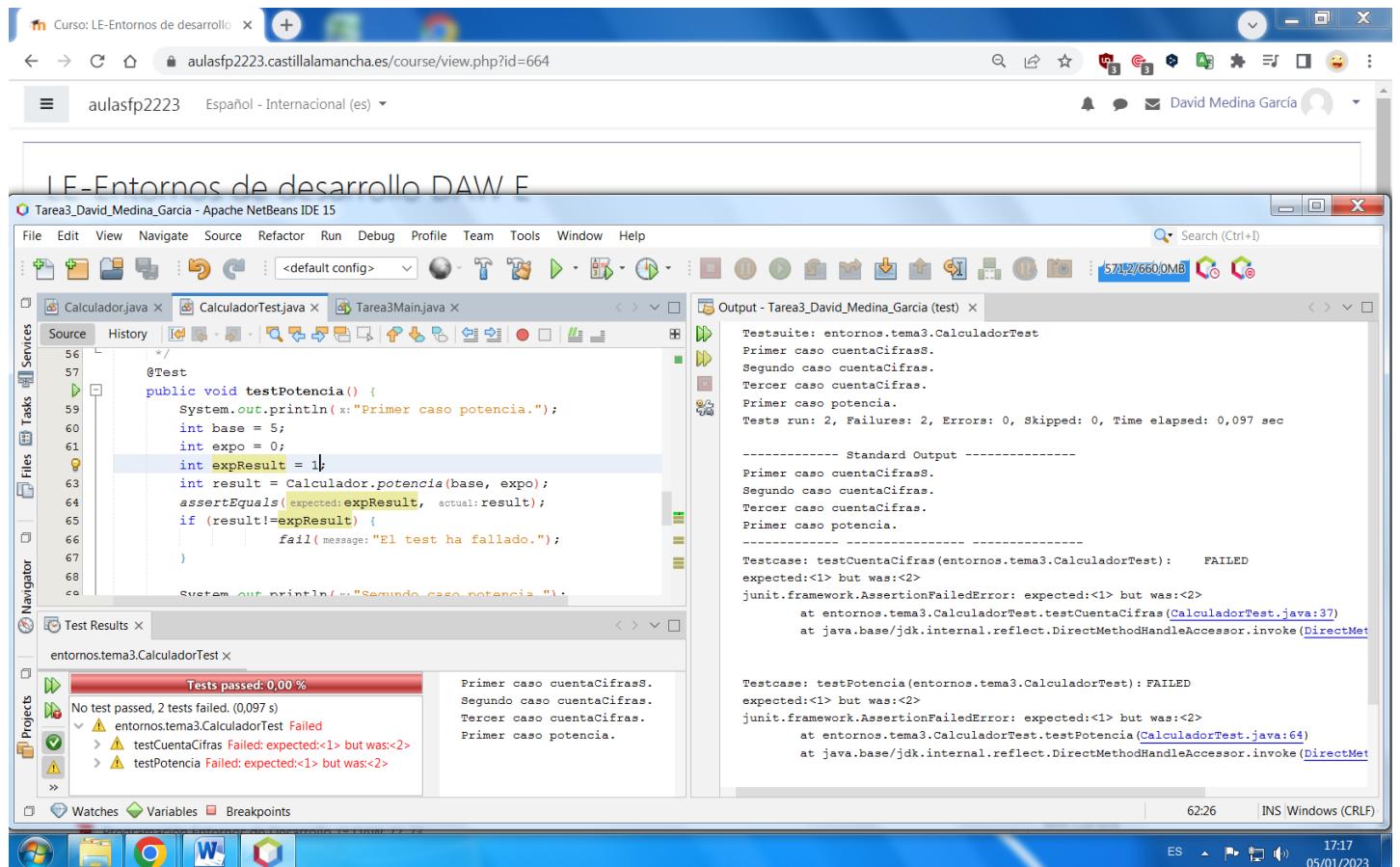
The screenshot shows the NetBeans IDE interface with the following details:

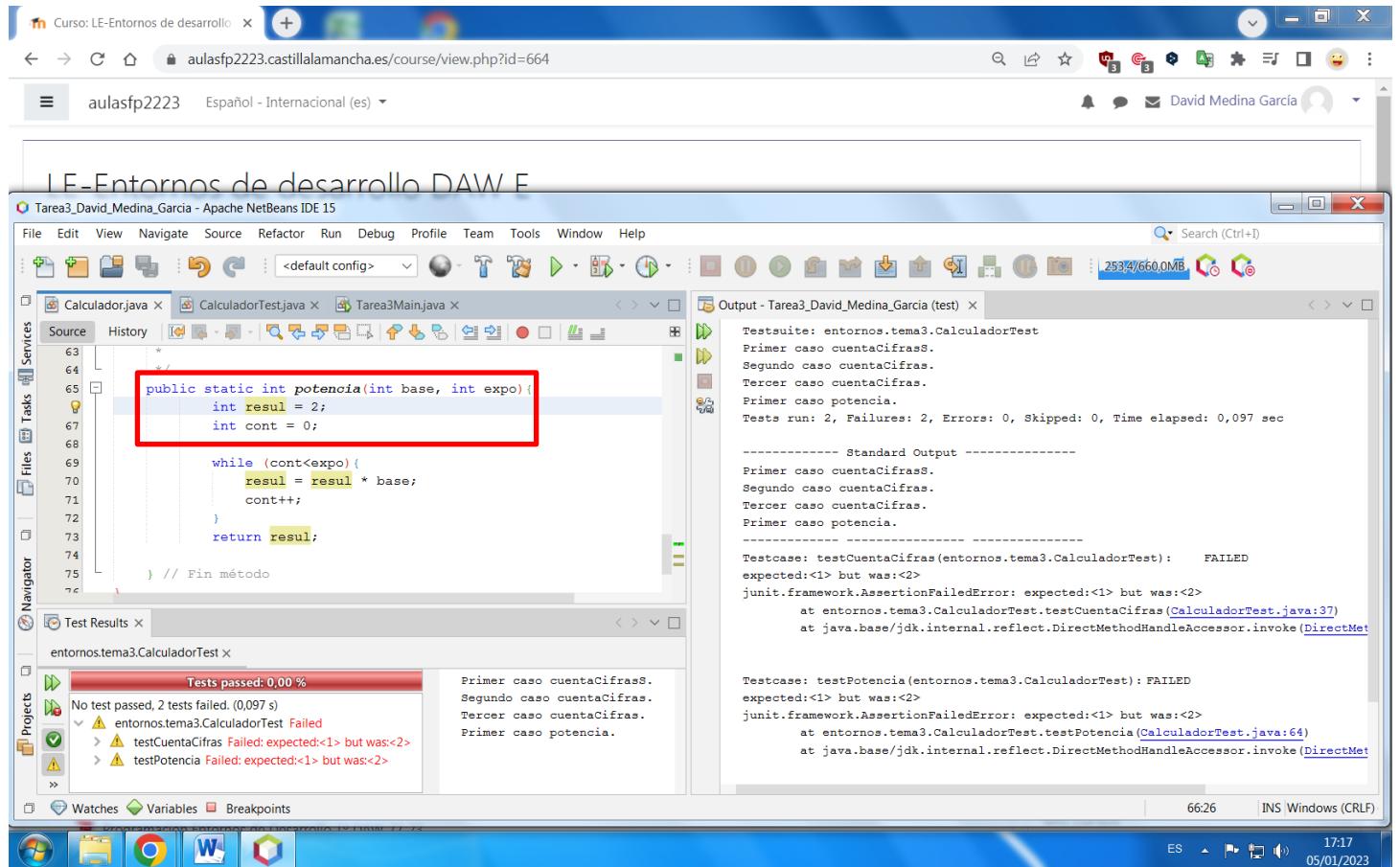
- Title Bar:** Curso: LE-Entornos de desarrollo x + aulasfp2223.castillalamancha.es/course/view.php?id=664 aulasfp2223 Español - Internacional (es) David Medina García
- Toolbar:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
- Source Editor:** Shows code for `CalculadorTest.java` with annotations for lines 38-51.
- Output Window:** Displays the execution of the test suite, showing standard output and a failed test case: `Testcase: testCuentaCifras(entornos.tema3.CalculadorTest): FAILED expected:<3> but was:<0>`.
- Test Results Window:** Shows the test results for `entornos.tema3.CalculadorTest`, indicating 1 test passed and 1 test failed.
- Bottom Status Bar:** Shows the build status as "Finished building Tarea3\_David\_Medina\_Garcia (test).", the time as 25:22, and the date as 15/01/2023.



## PRIMER CASO POTENCIA (EXPONENTE 0)

Forzamos el error cambiando el valor de la variable **result** (1 por 2).





## SEGUNDO CASO POTENCIA (EXPONENTE NEGATIVO)

El error anterior nos sirve también en este caso, cambiando el valor de la variable **result** (1 por 2).

The screenshot shows the Apache NetBeans IDE 15 interface. The top bar displays the URL <aulasfp2223.castillalamancha.es/course/view.php?id=664>. The main window has tabs for `Calculadora.java`, `CalculadoraTest.java`, and `Tarea3Main.java`. The `CalculadoraTest.java` tab is active, showing the following code:

```
public void testPotencia() {
    System.out.println("Segundo caso potencia.");
    int base = 5;
    int expo = -10;
    int expResult = 1;
    int result = Calculadora.potencia(base, expo);
    assertEquals(expected:expResult, actual:result);
    if (result!=expResult) {
        fail(message:"El test ha fallado");
    }
    System.out.println("Primer caso potencia.");
    base = 5;
    expo = 1;
```

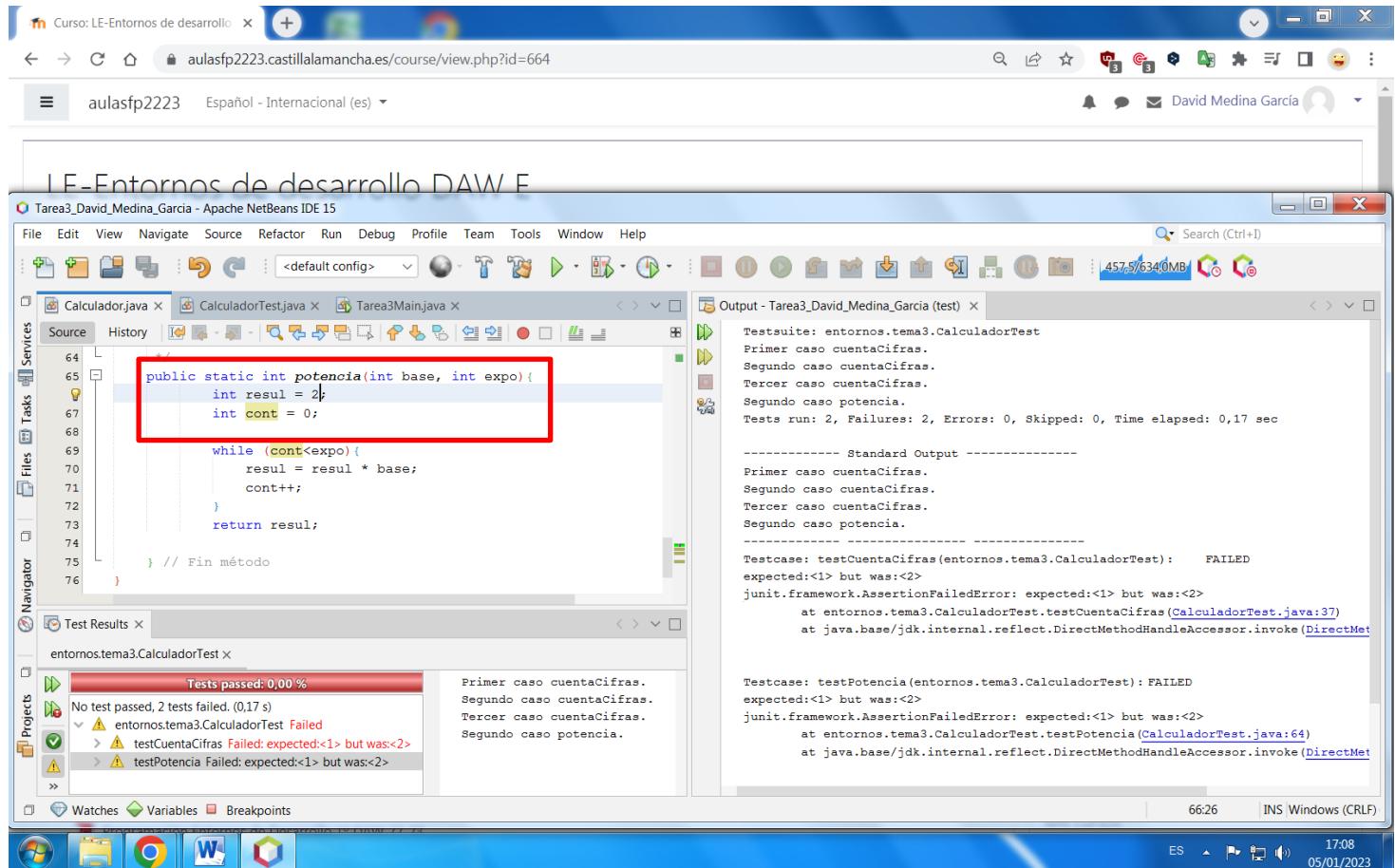
The `Output - Tarea3_David_Medina_Garcia (test)` window shows the test results:

```
Testsuite: entornos.tema3.CalculadoraTest
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Segundo caso potencia.
Tests run: 2, Failures: 2, Errors: 0, Skipped: 0, Time elapsed: 0,17 sec
-----
Standard Output -----
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Segundo caso potencia.
-----
Testcase: testCuentaCifras(entornos.tema3.CalculadoraTest): FAILED
expected:<1> but was:<2>
junit.framework.AssertionFailedError: expected:<1> but was:<2>
|   at entornos.tema3.CalculadoraTest.testCuentaCifras(CalculadoraTest.java:37)
|   at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(DirectMet...
```

The `Test Results` window shows the overall status:

```
Tests passed: 0,00 %
No test passed, 2 tests failed. (0,17 s)
entornos.tema3.CalculadoraTest Failed
  > testCuentaCifras Failed:expected:<1> but was:<2>
  > testPotencia Failed:expected:<1> but was:<2>
```

The `Output` window also displays the standard output from the test cases.



## TERCER CASO POTENCIA (BASE 0)

En este caso hemos forzado el error cambiando la multiplicación del **WHILE** por una resta.

The screenshot shows the Apache NetBeans IDE 15 interface. The top bar displays the URL <aulasfp2223.castillalamancha.es/course/view.php?id=664>. The main window title is "LE-Entornos de desarrollo DAW F". The menu bar includes File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help. The toolbar has various icons for file operations. The left sidebar shows Navigator, Projects, Services, Tasks, Files, and History. The central workspace contains three tabs: "Calculador.java", "CalculadorTest.java", and "Tarea3Main.java". The "CalculadorTest.java" tab shows Java code for testing a power function. The "Output - Tarea3\_David\_Medina\_Garcia (test)" tab displays the test results. It shows a testsuite named "entornos.tema3.CalculadorTest" with three test cases: "Primer caso cuentaCifras.", "Segundo caso cuentaCifras.", and "Tercer caso cuentaCifras.". The output also includes standard output and detailed failure messages for the failed test cases. The bottom status bar shows the time as 17:04 and the date as 05/01/2023.

```
76     fail(message: "El test ha fallado");
77 }
78
79 System.out.println(x:"Tercer caso potencia.");
80 base = 0;
81 expo = 5;
82 expResult = 0;
83 result = Calculador.potencia(base, expo);
84 assertEquals(expected:expResult, actual:result);
85 if (result!=expResult) {
86     fail(message: "El tercer test ha fallado");
87 }
88 }
```

```
Testsuite: entornos.tema3.CalculadorTest
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
Tests run: 2, Failures: 2, Errors: 0, Skipped: 0, Time elapsed: 0,09 sec
----- Standard Output -----
Primer caso cuentaCifras.
Segundo caso cuentaCifras.
Tercer caso cuentaCifras.
Primer caso potencia.
Segundo caso potencia.
Tercer caso potencia.
-----
Testcase: testCuentaCifras(entornos.tema3.CalculadorTest): FAILED
expected:<1> but was:<2>
junit.framework.AssertionFailedError: expected:<1> but was:<2>
    at entornos.tema3.CalculadorTest.testCuentaCifras(CalculadorTest.java:37)
    at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(DirectMethodHandleAccessor.java:110)
    at java.base/java.lang.reflect.Method.invoke(Method.java:527)
    at org.junit.runners.model.FrameworkMethod$1.runReflectiveCall(FrameworkMethod.java:50)
    at org.junit.internal.runners.model.ReflectiveCallable.run(ReflectiveCallable.java:12)
    at org.junit.runners.model.FrameworkMethod.invokeExplosively(FrameworkMethod.java:47)
    at org.junit.internal.runners.statements.InvokeMethod.evaluate(InvokeMethod.java:17)
    at org.junit.runners.ParentRunner$3.evaluate(ParentRunner.java:290)
    at org.junit.runners.ParentRunner$12.evaluate(ParentRunner.java:501)
    at org.junit.runners.ParentRunner$11.evaluate(ParentRunner.java:484)
    at org.junit.runners.ParentRunner$10.evaluate(ParentRunner.java:463)
    at org.junit.runners.ParentRunner$9.evaluate(ParentRunner.java:437)
    at org.junit.runners.ParentRunner$8.evaluate(ParentRunner.java:408)
    at org.junit.runners.ParentRunner$7.evaluate(ParentRunner.java:380)
    at org.junit.runners.ParentRunner$6.evaluate(ParentRunner.java:358)
    at org.junit.runners.ParentRunner$5.evaluate(ParentRunner.java:328)
    at org.junit.runners.ParentRunner$4.evaluate(ParentRunner.java:307)
    at org.junit.runners.ParentRunner$2.evaluate(ParentRunner.java:285)
    at org.junit.runners.ParentRunner$1.evaluate(ParentRunner.java:264)
    at org.junit.runners.ParentRunner.run(ParentRunner.java:364)
    at org.junit.runner.JUnitCore.run(JUnitCore.java:137)
    at org.junit.runner.JUnitCore.(JUnitCore.java:83)
    at org.junit.runner.JUnitCore.(JUnitCore.java:83)
    at entornos.tema3.CalculadorTest.main(CalculadorTest.java:14)
Testcase: testPotencia(entornos.tema3.CalculadorTest): FAILED
expected:<0> but was:<1>
junit.framework.AssertionFailedError: expected:<0> but was:<1>
    at entornos.tema3.CalculadorTest.testPotencia(CalculadorTest.java:47)
    at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(DirectMethodHandleAccessor.java:110)
    at java.base/java.lang.reflect.Method.invoke(Method.java:527)
    at org.junit.runners.model.FrameworkMethod$1.runReflectiveCall(FrameworkMethod.java:50)
    at org.junit.internal.runners.model.ReflectiveCallable.run(ReflectiveCallable.java:12)
    at org.junit.runners.model.FrameworkMethod.invokeExplosively(FrameworkMethod.java:47)
    at org.junit.internal.runners.statements.InvokeMethod.evaluate(InvokeMethod.java:17)
    at org.junit.runners.ParentRunner$3.evaluate(ParentRunner.java:290)
    at org.junit.runners.ParentRunner$12.evaluate(ParentRunner.java:501)
    at org.junit.runners.ParentRunner$11.evaluate(ParentRunner.java:484)
    at org.junit.runners.ParentRunner$10.evaluate(ParentRunner.java:463)
    at org.junit.runners.ParentRunner$9.evaluate(ParentRunner.java:437)
    at org.junit.runners.ParentRunner$8.evaluate(ParentRunner.java:408)
    at org.junit.runners.ParentRunner$7.evaluate(ParentRunner.java:380)
    at org.junit.runners.ParentRunner$6.evaluate(ParentRunner.java:358)
    at org.junit.runners.ParentRunner$5.evaluate(ParentRunner.java:328)
    at org.junit.runners.ParentRunner$4.evaluate(ParentRunner.java:307)
    at org.junit.runners.ParentRunner$2.evaluate(ParentRunner.java:285)
    at org.junit.runners.ParentRunner$1.evaluate(ParentRunner.java:264)
    at org.junit.runners.ParentRunner.run(ParentRunner.java:364)
    at org.junit.runner.JUnitCore.run(JUnitCore.java:137)
    at org.junit.runner.JUnitCore.(JUnitCore.java:83)
    at org.junit.runner.JUnitCore.(JUnitCore.java:83)
    at entornos.tema3.CalculadorTest.main(CalculadorTest.java:14)
```

