

1. Ejecutar el programa , provocar el fallo del mismo y capturar un pantallazo del error que aparece. ¿Cuál es la hipótesis que según tu opinión ha provocado el fallo?

```
/home/ajotal998/Escritorio/programacion/Programacion_21_22AJ/src/main/java/practica_debugging_AJ/ej1.java:9: error: Illegal static dec
public static void main(String[] args) {
      ^
modifier 'static' is only allowed in constant variable declarations
```

Mi hipótesis es que las clases no están bien creadas.

2. Una vez corregido el error añade el código corregido al fichero y captura pantallazos de la consola donde se pueda ver que el programa funciona correctamente tanto para un caso en el que esté el elemento como para un caso en el que no esté el elemento.

```
public class ej1 {

    public static void main(String[] args) {

        int[] v = new int[6];
        int numero;
        Scanner sc = new Scanner(System.in);

        for (int i = 0; i < v.length; i++) {
            System.out.println("Introduce el elemento para la posición " + i + " del vector");
            v[i] = sc.nextInt();
        }

        System.out.printf("Introduce el número a buscar en el vector");
        numero = sc.nextInt();

        boolean resultado = esta(v, numero);
    }
}
```

```

        if (resultado) {
            System.out.printf("El elemento " + numero + " está en el vector " + Arrays.toString(v));
        } else {
            System.out.printf("El elemento " + numero + " no está en el vector" + Arrays.toString(v));
        }
    }

    public static boolean esta(int[] vector, int elemento) {

        for (int i = 0; i <= vector.length; i++) {
            if (vector[i] == elemento) {
                return true;
            }
        }

        return false;
    }
}

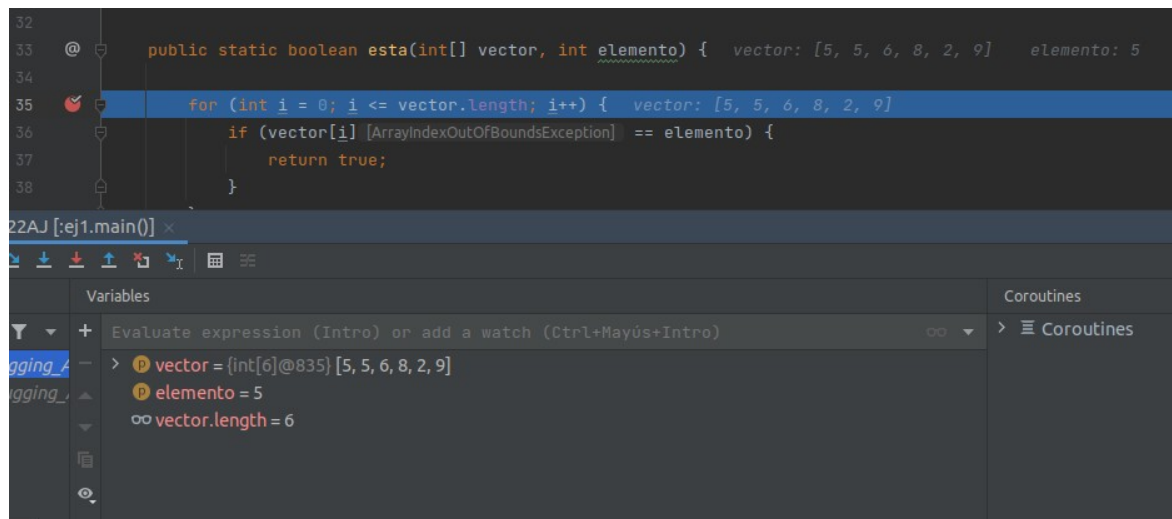
```

```

Run: Programacion_21_22AJ [:ej1.main()] x
15:45:56: Executing ':ej1.main()'...
> Task :compileJava
> Task :processResources NO-SOURCE
> Task :classes
> Task :ej1.main()
Introduce el elemento para la posición 0 del vector

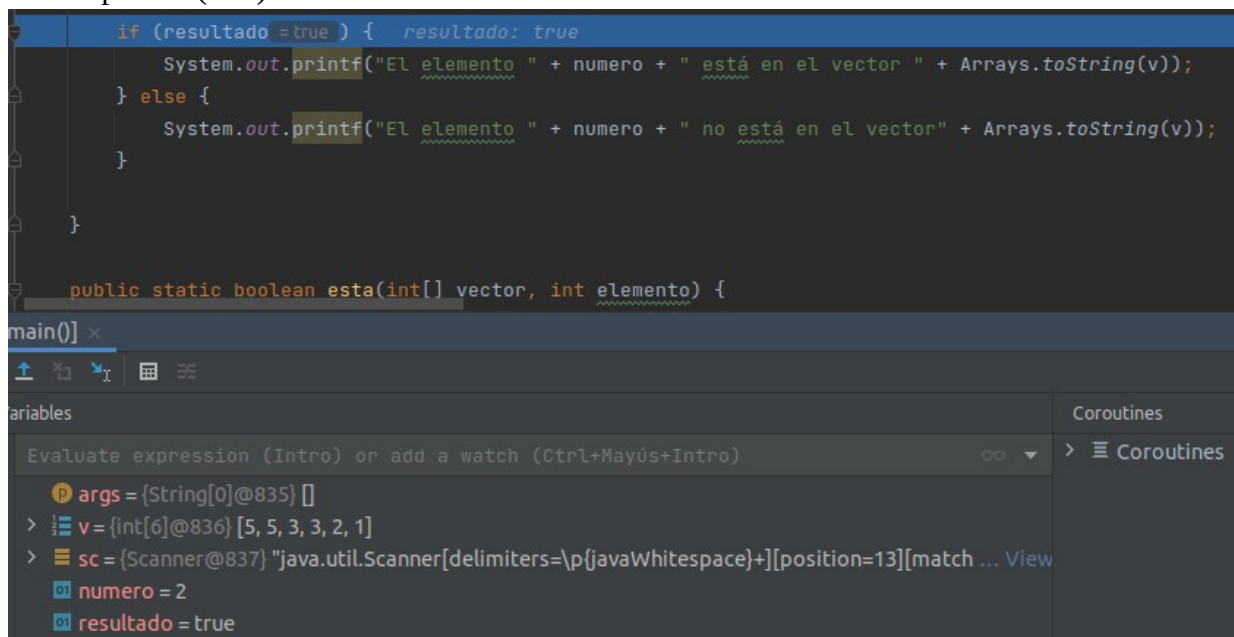
```

3. Establecer un breakpoint para comprobar que v se rellena correctamente. Justifica por qué has elegido establecer el breakpoint en esa línea de código e incluye una captura de pantalla donde se pueda comprobar que se ha parado la ejecución y que muestre en el área de depuración "Variables" todos los elementos del vector.



He establecido el breakpoint en esta posición para comprobar que el vector se rellena con los dígitos que introducimos de una manera correcta. En el área de depuración aparece el vector, el elemento que hemos solicitado y la longitud del vector.

4. Establecer un breakpoint en el programa principal que nos permita comprobar que la función está devolviendo un valor true (deberás introducir los valores necesarios para que así sea). Justificar por qué has elegido establecer el breakpoint en esa línea de código. Incluye un pantallazo donde se pueda comprobar que se efectivamente se ha devuelto el resultado esperado (true).



He elegido establecer el breakpoint en el primer condicional if porque es el lugar donde se va a mostrar si el valor de la variable booleana es true.

5. Modificar el valor del resultado obtenido en el apartado anterior a false desde la zona de Variables. Describir el proceso realizado y capturar un pantallazo donde se pueda comprobar que el mensaje ya no es el adecuado "No está" y donde se pueda ver en la zona de variables que efectivamente se ha modificado el resultado de la función.

```

25  if (resultado == false) { resultado: false
26      System.out.printf("El elemento " + numero + " está en el vector " + Arrays.toString(v));
27  } else {
28      System.out.printf("El elemento " + numero + " no está en el vector" + Arrays.toString(v));
29  }
30
31  }
32
33  @
    public static boolean esta(int[] vector, int elemento) {

```

22AJ [ej1.main()] x

Variables

Evaluate expression (Intro) or add a watch (Ctrl+Mayús+Intro)

args = (String[0]@835) []

v = (int[6]@836) [2, 4, 6, 3, 4, 2]

sc = (Scanner@837) "java.util.Scanner[delimiters=\p(javaWhitespace)+][position=13][match ... View

numero = 2

resultado = false

Coroutines

> Coroutines

Volvemos a posicionar el breakpoint en el primer condicional if que este caso nos va a devolver un valor false porque hemos modificado la variable booleana de la función invirtiendo de true a false los valores que debe devolver.

6. Establecer un breakpoint dentro de la función y realizar un pantallazo donde se pueda comprobar que se ha parado la ejecución y el contenido del vector recibido.

```

    for (int i = 0; i <= vector.length; i++) { i: 0
        if (vector[i] == elemento) { vector: [5, 3, 6, 4, 7, 4] elemento: 3 i: 0
            return false;
        }
    }

    return true;
}

```

main() x

Variables

Evaluate expression (Intro) or add a watch (Ctrl+Mayús+Intro)

vector = (int[6]@835) [5, 3, 6, 4, 7, 4]

elemento = 3

i = 0

vector[i] = 5

vector.length = 6

Coroutine

> Cor

7. Utilizando la zona de variables y una vez se ha parado en el breakpoint anterior modificar el contenido del vector para modificar el resultado previsto en la función. Si debía de ser true pasará a ser false y al revés. Incluid los pantallazos necesarios para justificar dicho cambio.

```
boolean resultado = esta(v, numero); v: [3, 1, 6, 7, 4, 2] numero: 1 resultado: true

if (resultado == true) { resultado: true
    System.out.printf("El elemento " + numero + " está en el vector " + Arrays.toString(v));
} else {
    System.out.printf("El elemento " + numero + " no está en el vector" + Arrays.toString(v));
}
}
```

main() x

Variables

Evaluate expression (Intro) or add a watch (Ctrl+Mayús+Intro)

- args = {String[0]@835} []
- v = {int[6]@836} [3, 1, 6, 7, 4, 2]
- sc = {Scanner@837} "java.util.Scanner[delimiters=\p{javaWhitespace}+][position=13][match ... View
- numero = 1
- resultado = true

Coroutines

> Coroutines

En este caso hemos comprobado que devuelve true porque el numero se encuentra en el vector.

```
boolean resultado = esta(v, numero); v: [5, 5, 3, 2, 7, 8] numero: 9 resultado: false

if (resultado == false) { resultado: false
    System.out.printf("El elemento " + numero + " está en el vector " + Arrays.toString(v));
} else {
    System.out.printf("El elemento " + numero + " no está en el vector" + Arrays.toString(v));
}
}
```

main() x

Variables

Evaluate expression (Intro) or add a watch (Ctrl+Mayús+Intro)

- args = {String[0]@835} []
- v = {int[6]@836} [5, 5, 3, 2, 7, 8]
- sc = {Scanner@837} "java.util.Scanner[delimiters=\p{javaWhitespace}+][position=13][match ... View
- numero = 9
- resultado = false

Coroutines

> Coroutines

Modificamos clicando en numero e indicándole el nuevo valor con set value o F2, la variable booleana en este caso pasará a ser false porque el número no se encuentra en el vector.