

ACTIVIDAD 1

Pon solución a este código en Java que resolverá lo siguiente:

“Pide por teclado un número entero positivo (debemos controlarlo) y muestra el número de cifras que tiene. Por ejemplo: si introducimos 1250, nos muestre que tiene 4 cifras. Tendremos que controlar si tiene una o más cifras, al mostrar el mensaje.”

- Pon puntos de ruptura y observa lo que está pasando.
- Plantea posible solución al problema.

Código

```
package cuentacifrasapp;

import java.util.Scanner;
public class CuentaCifrasApp {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        Scanner sc = new Scanner(System.in);
        String nombre="Luisa";
        int numero=0;
        do{
            System.out.println("Introduce un numero");
            numero=sc.nextInt();
        }while(numero<0);
        int contador=0;
        for (int i=numero;i>1;i=i/10){
            //Incrementamos el contador
            contador++;
        }

        //Controlamos en el caso de que haya una cifra o mas
        if (contador==1){
            System.out.println("El numero "+numero+ " tiene
"+contador+" cifra");
        }else{
            System.out.println("El numero "+numero+ " tiene
"+contador+" cifras");
        }
    }
}
```

Solución:

Lo primero que hacemos para encontrar el error que tiene nuestro programa es colocar un punto de ruptura. En este programa el punto de ruptura lo voy a colocar en el bucle for, porque en este bucle es donde nos encontraremos los procesos por los que tiene que pasar para darnos una solución. En la siguiente imagen encontraremos la ubicación en la que he colocado el punto de ruptura.

```
CuentaCifrasApp.java x
1  import java.util.Scanner;
2  public class CuentaCifrasApp {
3
4      /**
5       * @param args the command line arguments
6       */
7      public static void main(String[] args) {
8          // TODO code application logic here
9          Scanner sc = new Scanner(System.in);
10         String nombre="Marina Laguna";
11         int numero=0;
12         do{
13             System.out.println("Introduce un numero");
14             numero=sc.nextInt();
15         }while(numero<0);
16         int contador=0;
17         for (int i=numero;i>1;i=i/10){
18             //Incrementamos el contador
19             contador++;
20         }
21
22         //Controlamos en el caso de que haya una cifra o mas
23         if (contador==1){
24             System.out.println("El numero "+numero+ " tiene "+contador+" cifra");
25         }else{
26             System.out.println("El numero "+numero+ " tiene "+contador+" cifras");
27         }
28     }
29 }
30 }
```

Una vez colocado el punto de ruptura iniciamos el debugger, como nuestro punto de ruptura se encuentra después de introducir los datos por pantalla, nos pedirá primero introducir los datos y después podremos hacer el seguimiento del debugger, con una de las flechitas avanzaremos el programa e intentaremos encontrar cuál es el fallo que tiene nuestro programa.

En este caso si ponemos un número con 9 cifras, nos debería devolver eso que el número tiene 9 cifras, pero lo que nos devuelve es que tiene 8 cifras.

Al hacer el debugger vemos que entra en el bucle y cuando cuenta la cifra 8 sale de este.

Para resolver este error lo único que he tenido que cambiar es poner en la condición del for exactamente en $i > 1$, el cambio que he realizado es añadir un igual para que sea i mayor o igual que 1.

```
12         do{
13             System.out.println("Introduce un numero");
14             numero=sc.nextInt();
15         }while(numero<0);
16         int contador=0;
17         for (int i=numero;i>=1;i=i/10){
18             //Incrementamos el contador
19             contador++;
20         }
21     }
```

Con esto nuestro error en este programa esta solucionado.

ACTIVIDAD 2

Pon solución a este código en Java que resolverá lo siguiente:

Pedir números al usuario y cuando el usuario meta un -1 se terminará el programa.

Al terminar, mostrará lo siguiente:

- mayor numero introducido
- menor número introducido
- suma de todos los números
- suma de los números positivos
- suma de los números negativos
- media de la suma (la primera que pido)

El número -1 no contara como número.

Código

```
package ejercicio2b;

import java.util.Scanner;

public class Ejercicio2b {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        //Variables necesarias
        String nombre = "Nombre_Apellidos";
        int numeroUsuario=0;
        int menor=0;
        int mayor=0;
        int suma=0;
        double media;
        int contador=-1;
        int sumaPositivos=0;
        int sumaNegativos=0;

        //Creamos el Scanner
        Scanner sn = new Scanner(System.in);

        do{

            //Comprobamos si el numero es menor
            //que el actual menor
            if(menor>numeroUsuario){
                mayor = numeroUsuario;
            }

            //Comprobamos si el numero es mayor
            //que el actual mayor
            if(mayor<numeroUsuario){
                menor=numeroUsuario;
            }

        }
```

```
        //Suma el numero y lo acumulamos
        suma+=numeroUsuario;

        //Si el numero es positivo, suma a la variable de los
positivos
        // y sino a la de los negativos
        if(numeroUsuario>=0){
            sumaPositivos+=numeroUsuario;
        }else{
            sumaNegativos+=numeroUsuario;
        }

        //aumento el contador
        contador++;

        //pido un numero al usuario
        System.out.println("Introduce un numero");
        numeroUsuario = sn.nextInt();

        //Cuando el usuario ponga un -1, saldremos
        //Se puede usar un while normal
    }while(numeroUsuario!=-1);

    //Calculamos la media
    media = (double) suma / contador;

    //Mostramos los valores
    System.out.println("El menor es: "+menor);
    System.out.println("El mayor es: "+mayor);
    System.out.println("La suma es: "+suma);
    System.out.println("La suma de los positivos es: "+suma);
    System.out.println("La suma de los negativos es: "+suma);
    System.out.println("La media es: "+media);

    }

}
```

Solución:

El punto de ruptura en este ejercicio lo pondremos al terminar el bucle do {...}while.

```
Ejercicio2b.java
26 // menor = numeroUsuario;
27 mayor = numeroUsuario;
28 }
29
30 //Comprobamos si el numero es mayor
31 //que el actual mayor
32 if(mayor<numeroUsuario){
33     menor=numeroUsuario;
34 }
35
36 //Suma el numero y lo acumulamos
37 suma+=numeroUsuario;
38
39 //Si el numero es positivo, suma a la variable de los positivos
40 // y sino a la de los negativos
41 if(numeroUsuario>0){
42     sumaPositivos+=numeroUsuario;
43 }else{
44     sumaNegativos+=numeroUsuario;
45 }
46
47 //aumento el contador
48 contador++;
49
50 //pido un numero al usuario
51 System.out.println("Introduce un numero");
52 numeroUsuario = sn.nextInt();
53
54 //Cuando el usuario ponga un -1, saldremos
55 //Se puede usar un while normal
56 }while(numeroUsuario!=-1);
57
```

Como hicimos en el anterior ejercicio con el debugger comprobaremos los errores que tenga y veremos donde falla. En este ejercicio el error lo encontramos en los dos if que tenemos para comprobar si el numero es mayor o menor, para ello con cambiar los signos es suficiente.

```
Scanner sn = new Scanner(System.in);

do{

    //Comprobamos si el numero es menor
    //que el actual menor
    if(mayor>numeroUsuario){
        menor = numeroUsuario;
    }

    //Comprobamos si el numero es mayor
    //que el actual mayor
    if(menor<numeroUsuario){
        mayor=numeroUsuario;
    }

}
```

ACTIVIDAD 3

Analizar el siguiente código del Ejercicio 1 del examen de programación.

OJO- Cambiar la variable de nombre.

- A) Muestra una fila, en la jornada que se realizan 5 turnos y en cada turno hay 3 personas.- El programa tiene que mostrar:

```
Turno 1
  Personas atendidas en este turno: 3
Turno 2
  Personas atendidas en este turno: 3
Turno 3
  Personas atendidas en este turno: 3
Turno 4
  Personas atendidas en este turno: 3
Turno 5
  Personas atendidas en este turno: 3
```

Utilizando depuración, soluciona el problema. -Captura debug utilizado.

- B) Utilizando las funciones de depuración, muestra el valor de la cola para el primer turno y el valor de la posición 9 para el turno 3. Capturas de debug.

PRUEBA_PRACTICA2

```
fun main(args: Array<String>) {
    val MEDIA:Int = 5
    var cola= Array<Int>(15){0}
    var personasPorTurno: Int = 15
    var turno: Int = 1
    var nombre: String = "Luisa"
    while(turno<MEDIA){
        println(texto(4) + turno)
        rellenarCola(personasPorTurno,cola)
        personasPorTurno=3
        println( texto(2)+ personasPorTurno)
        avanzarCola(personasPorTurno, cola)
        turno ++
    }
}

fun limites(i:Int):Int{
    var n:Int=0
    when (i){
        1 -> n = 1
        3 -> n = 5
        9 -> n = 19
    }
    return n
}

fun texto(i:Int):String{
```

```

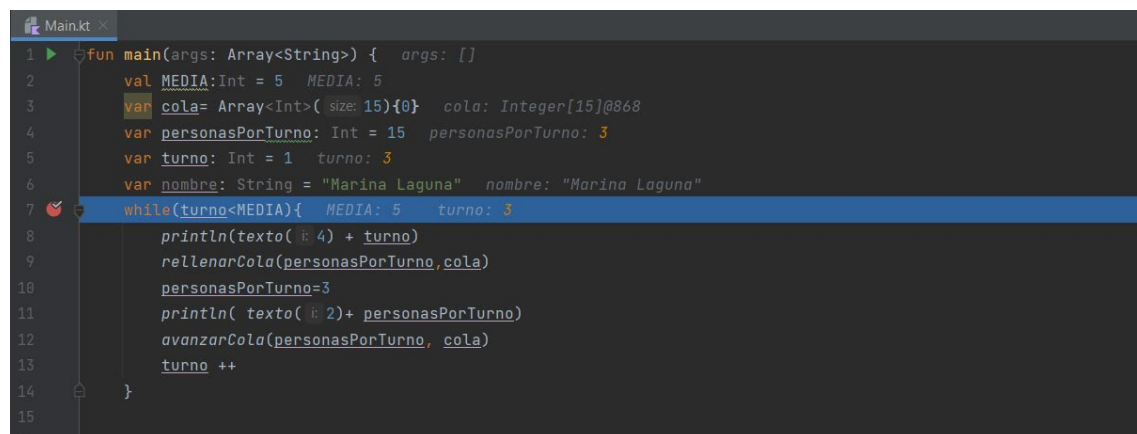
var s:String=""
when (i){
    1 -> s = "Cola esperando turno"
    2 -> s="    Personas atendidas en este turno: "
    3 -> s="Cola con los huecos generados"
    4 -> s="Turno "
    5 -> s="["
    6 -> s="]"
    7 -> s = "    "
}
return s
}
}
fun avanzarCola(cantidad:Int, v: Array<Int>){
    var i:Int = (v.size-1) - cantidad
    while (i>=0){
        v[i+cantidad]= v[i]
        i--;
    }
    i=0
    while (i< cantidad){
        v[i]= 0
        i++;
    }
}
fun rellenarCola(cantidad: Int,v:Array<Int>){
    // cantidad es el numero de elementos a rellenar, se restaran a la
    longitud total
    //y desde esa posición se rellenara
    var i:Int = 0

    while (i<cantidad){
        v[i]= (limites(1)..limites(9)).random()
        i++;
    }
}
}

```

Solución:

En este ejercicio el punto de debugger lo ponemos al empezar el bucle While.



```

Main.kt
1  fun main(args: Array<String>) { args: []
2      val MEDIA:Int = 5 MEDIA: 5
3      var cola= Array<Int>( size: 15){0} cola: Integer[15]@868
4      var personasPorTurno: Int = 15 personasPorTurno: 3
5      var turno: Int = 1 turno: 3
6      var nombre: String = "Marina Laguna" nombre: "Marina Laguna"
7      while(turno<MEDIA){ MEDIA: 5 turno: 3
8          println(texto( 4) + turno)
9          rellenarCola(personasPorTurno,cola)
10         personasPorTurno=3
11         println( texto( 2)+ personasPorTurno)
12         avanzarCola(personasPorTurno, cola)
13         turno ++
14     }
15

```