

## Práctica para utilizar DEBUG

- 1) Dado el siguiente código de una clase java (con errores) que invierte un array, realizar DEBUG incluyendo puntos de ruptura para depurar dicho programa hasta que funcione correctamente. Ir generando un documento que debe entregarse en formato pdf con las capturas correspondientes para comprobar que habéis utilizado el debugger con puntos de ruptura comprobando valores de variables, etc., hasta que el programa funcione correctamente.

Nota: No tratar excepciones.

```
package arrays;
```

```
public class invertir_array {
```

```
    public invertir_array() {  
        // TODO Auto-generated constructor stub  
    }
```

```
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        //declaramos array a invertir y la variable que nos va a servir para  
intercambiar los valores del array  
        int array[] = {1,2,3,4,5,6,7,8,9};  
        int aux=0;
```

```
        // imprimimos array para comprobar  
        int longitud=array.length;  
        for (int i = 0; i<=longitud; i++)  
            System.out.print(array[i] + " ");  
        // invertimos array recorriéndolo hasta la mitad e intercambiando valores
```

```
        for(int i=0;i<=longitud/2; i++){  
            //vamos guardando la posición que vamos a intercambiar desde el final  
            aux = array[longitud-i];  
            //como ya tenemos guardada la posición longitud-i-1, ya podemos guardar en  
esa posición un valor  
            array[longitud-i-1] = array[i];  
            //guardamos en las posiciones iniciales el valor de aux que ya estaba  
guardado  
            array[i] = aux;  
        }
```

```
        // comprobamos que se ha invertido el array  
        System.out.println();  
        for (int i = 0; i < longitud; i++)  
            System.out.print(array[i] + " ");  
    }
```

## Práctica para suite de pruebas.

- 2) Realizar una clase en java denominada `menormayor_primo` con dos métodos:
  - El primero llamado `primos` calculará los tres últimos números primos de un número
  - El segundo llamado `menormayor` calculará el menor y el mayor de 3 números introducidos como parámetros

Realizar una clase de prueba con JUNIT TEST CASE para comprobar el método `menormayor` y documentarlo mediante capturas.

Realizar una suite de pruebas con dos clases de prueba parametrizadas para probar los dos métodos de la clase `menormayor_primo`.

Nota: Entregar en un solo archivo comprimido tanto las capturas documentadas como las clases que haya sido necesario crear.