Alumno—
Jesus Octavio Amarillas Amaya

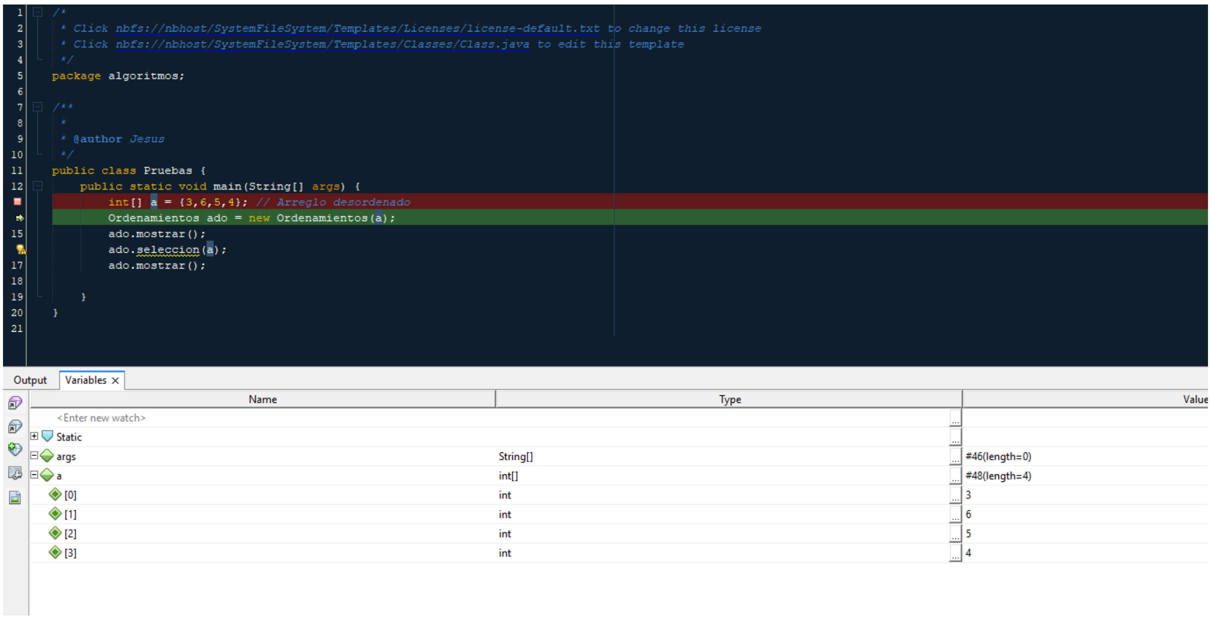
ID—
207653

Asignación—
Algoritmo Selección

Materia—
Análisis de Algoritmos

Profesor—
Sergio Castellanos Bustamante

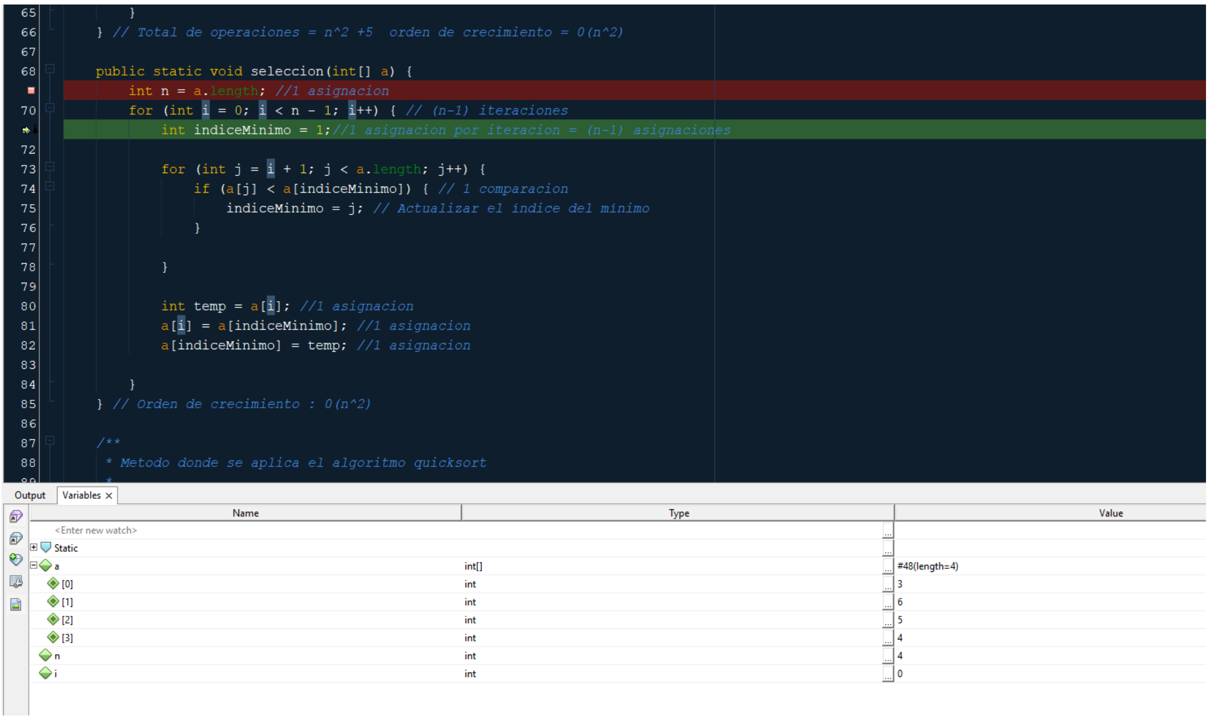
1. **Se declara el arreglo y se accede al método de selección.**



```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package algoritmos;

/**
 *
 * @author Jesus
 */
public class Pruebas {
    public static void main(String[] args) {
        int[] a = {3,6,5,4}; // Arreglo desordenado
        Ordenamientos ado = new Ordenamientos(a);
        ado.mostrar();
        ado.seleccion(a);
        ado.mostrar();

    }
}
```

Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#46(length=0)
a	int[]	#48(length=4)
[0]	int	3
[1]	int	6
[2]	int	5
[3]	int	4

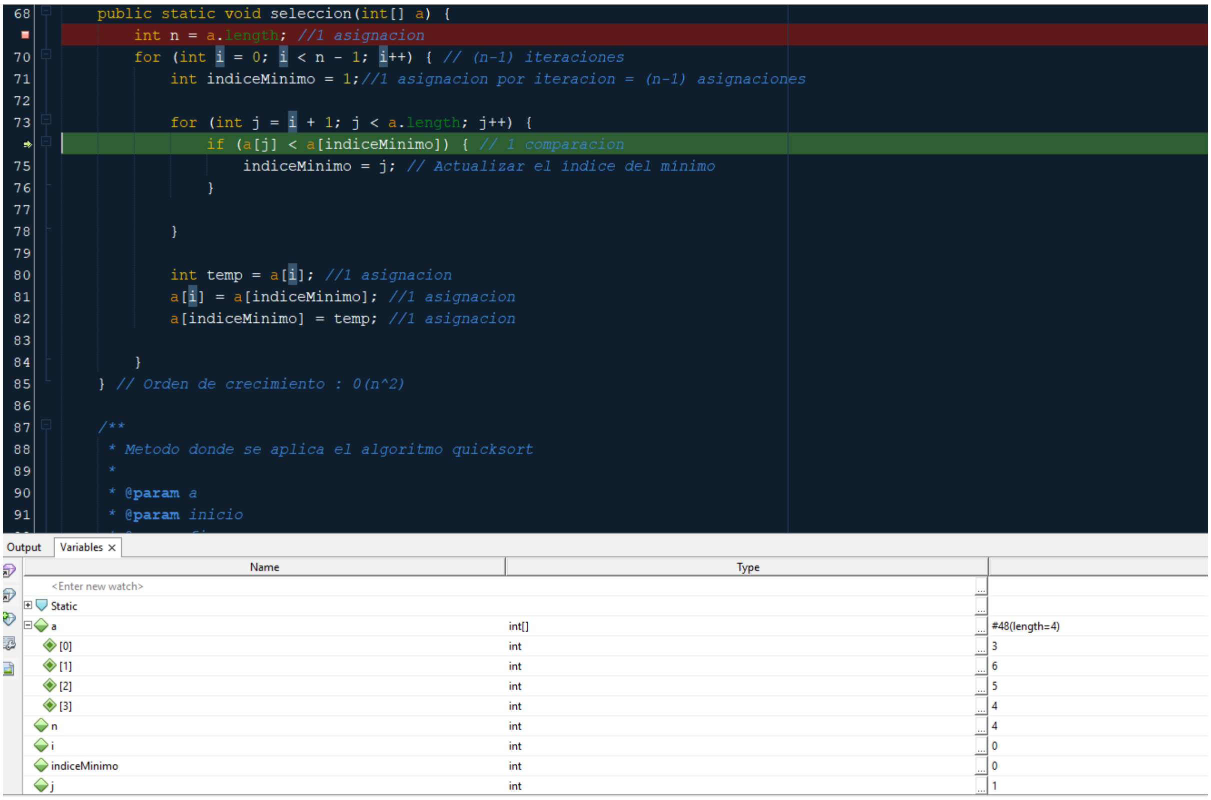
2. **Se declara n como el valor del tamaño del arreglo a ordenar.**



```
            }
    } // Total de operaciones = n^2 +5  orden de crecimiento = 0(n^2)

    public static void seleccion(int[] a) {
        int n = a.length; //1 asignacion
        for (int i = 0; i < n - 1; i++) { // (n-1) iteraciones
            int indiceMinimo = 1;//1 asignacion por iteracion = (n-1) asignaciones

            for (int j = i + 1; j < a.length; j++) {
                if (a[j] < a[indiceMinimo]) { // 1 comparacion
                    indiceMinimo = j; // Actualizar el índice del mínimo
                }

            }

            int temp = a[i]; //1 asignacion
            a[i] = a[indiceMinimo]; //1 asignacion
            a[indiceMinimo] = temp; //1 asignacion

        }
    } // Orden de crecimiento : 0(n^2)

    /**
     * Metodo donde se aplica el algoritmo quicksort
     *
```

Name	Type	Value
<Enter new watch>		
Static		
a	int[]	#48(length=4)
[0]	int	3
[1]	int	6
[2]	int	5
[3]	int	4
n	int	4
i	int	0

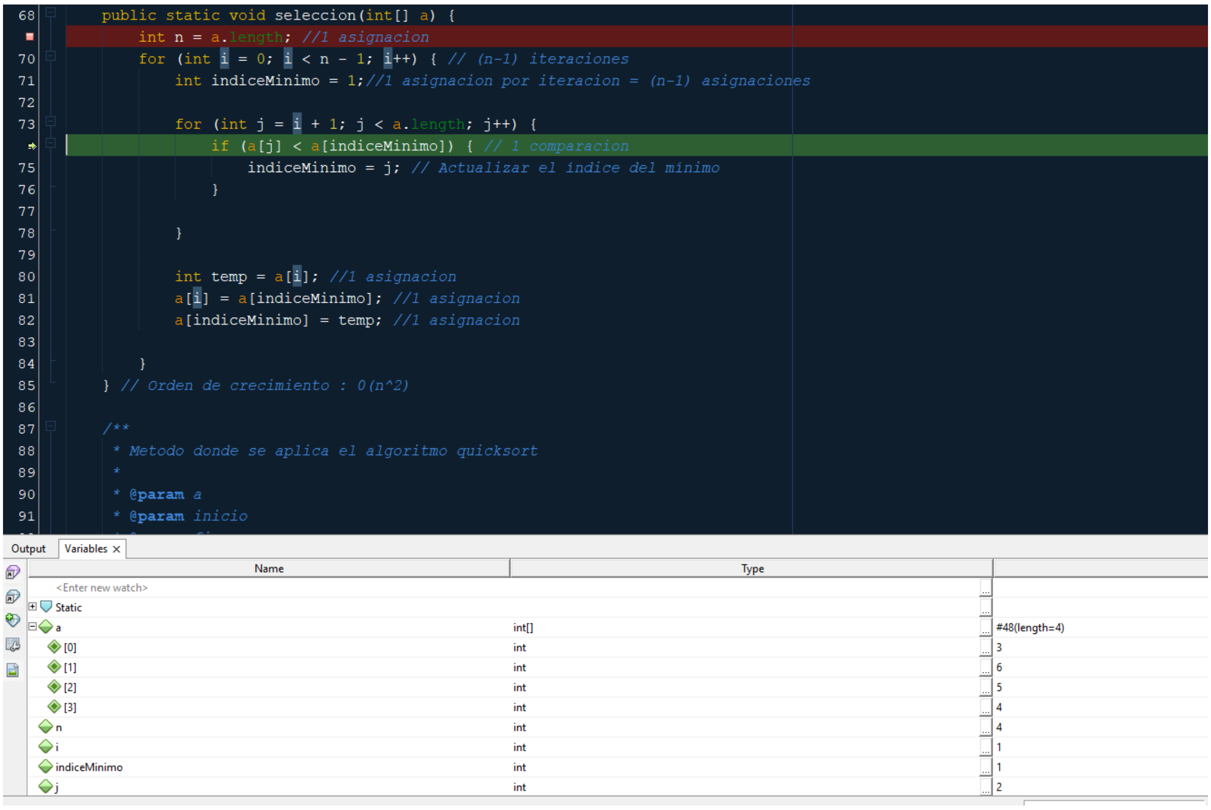
3. **Se inicia el ciclo for y se crea una variable dentro del mismo con el valor**



```java
    public static void seleccion(int[] a) {
        int n = a.length; //1 asignacion
        for (int i = 0; i < n - 1; i++) { // (n-1) iteraciones
            int indiceMinimo = 1;//1 asignacion por iteracion = (n-1) asignaciones

            for (int j = i + 1; j < a.length; j++) {
                if (a[j] < a[indiceMinimo]) { // 1 comparacion
                    indiceMinimo = j; // Actualizar el indice del minimo
                }

            }

            int temp = a[i]; //1 asignacion
            a[i] = a[indiceMinimo]; //1 asignacion
            a[indiceMinimo] = temp; //1 asignacion

        }
    } // Orden de crecimiento : O(n^2)

    /**
     * Metodo donde se aplica el algoritmo quicksort
     *
     * @param a
     * @param inicio
```

Name	Type	
<Enter new watch>		
Static		
a	int[]	#48(length=4)
[0]	int	3
[1]	int	6
[2]	int	5
[3]	int	4
n	int	4
i	int	0
indiceMinimo	int	0
j	int	1

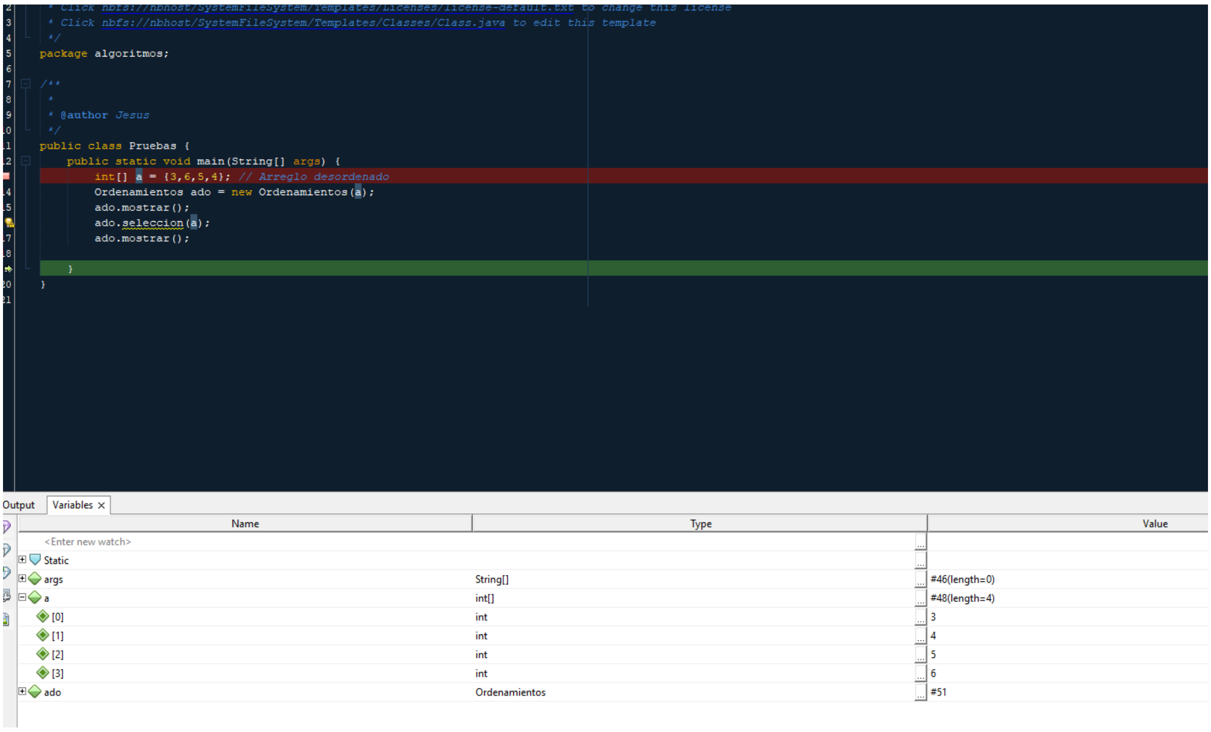
4. **Se crea un segundo for, con la diferencia de que la variable j tiene valor de i + 1.**



```java
    public static void seleccion(int[] a) {
        int n = a.length; //1 asignacion
        for (int i = 0; i < n - 1; i++) { // (n-1) iteraciones
            int indiceMinimo = 1;//1 asignacion por iteracion = (n-1) asignaciones

            for (int j = i + 1; j < a.length; j++) {
                if (a[j] < a[indiceMinimo]) { // 1 comparacion
                    indiceMinimo = j; // Actualizar el indice del minimo
                }

            }

            int temp = a[i]; //1 asignacion
            a[i] = a[indiceMinimo]; //1 asignacion
            a[indiceMinimo] = temp; //1 asignacion

        }
    } // Orden de crecimiento : O(n^2)

    /**
     * Metodo donde se aplica el algoritmo quicksort
     *
     * @param a
     * @param inicio
```

Name	Type	
<Enter new watch>		
Static		
a	int[]	#48(length=4)
[0]	int	3
[1]	int	6
[2]	int	5
[3]	int	4
n	int	4
i	int	1
indiceMinimo	int	1
j	int	2

5. **Al finalizar se termina el for principal y se sale del ciclo ya con el arreglo ordenado correctamente.**



```
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package algoritmos;

/**
 *
 * @author Jesus
 */
public class Pruebas {
    public static void main(String[] args) {
        int[] a = {3,6,5,4}; // Arreglo desordenado
        Ordenamientos ado = new Ordenamientos(a);
        ado.mostrar();
        ado.seleccion(a);
        ado.mostrar();

    }
}
```

Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#46(length=0)
a	int[]	#48(length=4)
[0]	int	3
[1]	int	4
[2]	int	5
[3]	int	6
ado	Ordenamientos	#51