

## Práctica 6:

# Resolución de problemas en Java (III)

## Objetivo de la práctica

Los objetivos de esta práctica son los siguientes:

- Algoritmos para la resolución de problemas.
- Implementar algoritmos en Java.
- Ejecutar algoritmos en el entorno Eclipse.

## 1. Algoritmo de ordenación Quicksort

Crear un nuevo proyecto Java con una clase donde se implemente el algoritmo Quicksort para ordenar un vector aleatorio de números enteros de tamaño  $n$ . Ejecutar paso a paso la solución para un tamaño  $n = 20$  para verificar la corrección del algoritmo.

Modificar el programa anterior para medir su tiempo de ejecución de este modo:

```
long tiempoInicio = System.currentTimeMillis();
// Llamada al método quicksort
// ...
long tiempoTotal = System.currentTimeMillis() - tiempoInicio;
// Resto del programa
// ...
System.out.println("Segundos: " + tiempoTotal/1000.);
```

## 2. Algoritmo de ordenación por Selección

Crear una nueva clase donde se implemente el algoritmo de ordenación anterior para ordenar un vector aleatorio de números enteros de tamaño  $n$ . La idea del algoritmo es calcular en cada paso  $i$ -ésimo el mínimo del vector que falta por ordenar y situarlo en la posición  $i$ -ésima del vector ordenado.

Ejecutar paso a paso la solución para comprobar el funcionamiento correcto con  $n$  igual a 20. Modificar el programa anterior para medir su tiempo de ejecución.

### 3. Comparación de los algoritmos de ordenación

Comparar el tiempo de ejecución de los dos algoritmos de ordenación para diferentes valores de  $n$ , por ejemplo  $10^4/2$ ,  $10^4$ ,  $10^5/2$  y  $10^5$ . Representarlo gráficamente usando la hoja de cálculo Microsoft Excel. En el eje X se representará el número de iteraciones del programa ( $n$ ) y en el eje Y el tiempo de ejecución en segundos, como se ilustra en la Figura 1.

Es posible que se obtenga un error por falta de memoria en los ordenadores del laboratorio:

```
Exception in thread "main" java.lang.OutOfMemoryError: Java heap space
at Criba4.main(Criba4.java:17)
```

Para solucionarlo, en la opción “Run configurations” accedemos a la pestaña “Arguments” y aumentamos la memoria escribiendo “-Xmx1200m” en el apartado “VM arguments”.

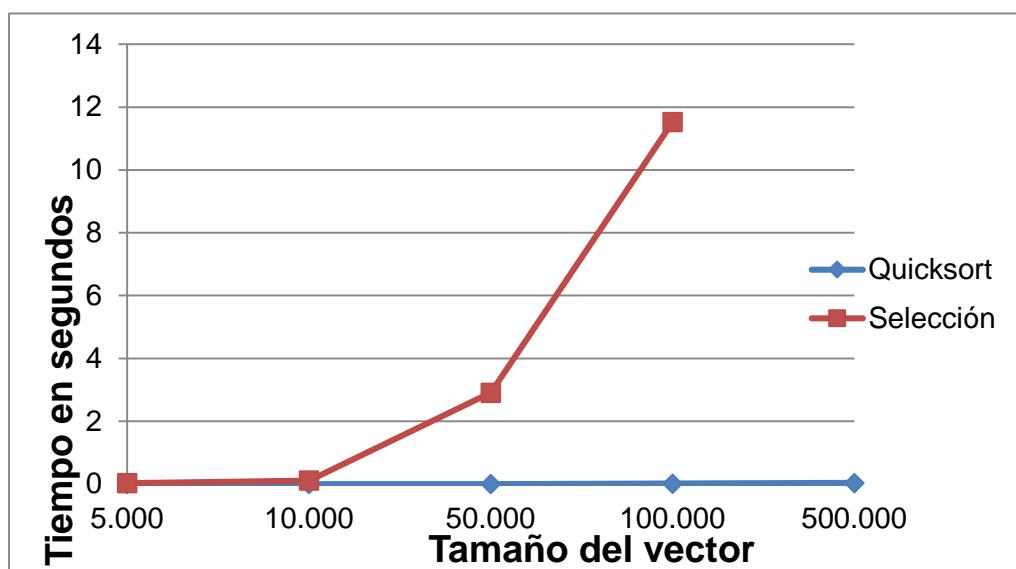


Figura 1. Tiempo de ejecución de cada algoritmo.