

# Tiempo de cada Sorting

El objetivo de este laboratorio era determinar el tiempo de ordenamiento por cada tipo de sorting existente, en este caso se usaron los siguientes:

- Merge Sort
- Heap Sort
- Quick Sort
- Radix Sort
- Selection Sort

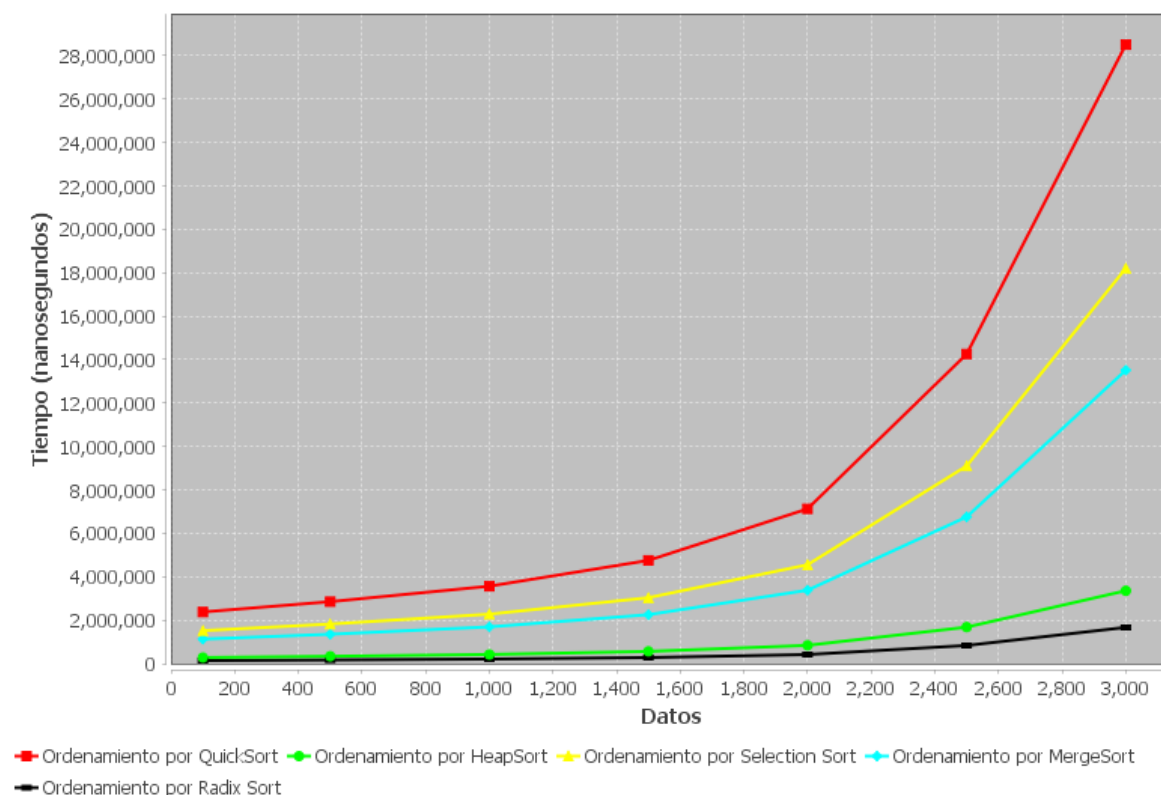
Para poder determinar el tiempo de ordenamiento de cada uno se uso las clases que java nos da, la cual es **javax.swing.Timer** y **java.util.Timer**. Estas nos permiten tener un control del tiempo de ejecución de cada método o del programa, logrando establecer alertas cada cierto tiempo mientras el programa este abierto.

Un ejemplo de como se implemento esto es el siguiente:

```
long startTimeHeapSort = System.nanoTime();
ob6.sort(arri);
long endTimeHeapSort = System.nanoTime() - startTimeHeapSort;
```

Donde nos permite determinar el tiempo en el cual el método se tardo en realizar. Obteniendo el siguiente resultado (los datos varían dependiendo de la computadora).

**Gráfica Tiempo/Elemento**



Donde X representa la cantidad de datos ordenados y Y el tiempo que se tardó. Siendo así el más rápido en este ejemplo Radix Sort.