

PROFILER IMPLEMENTADO

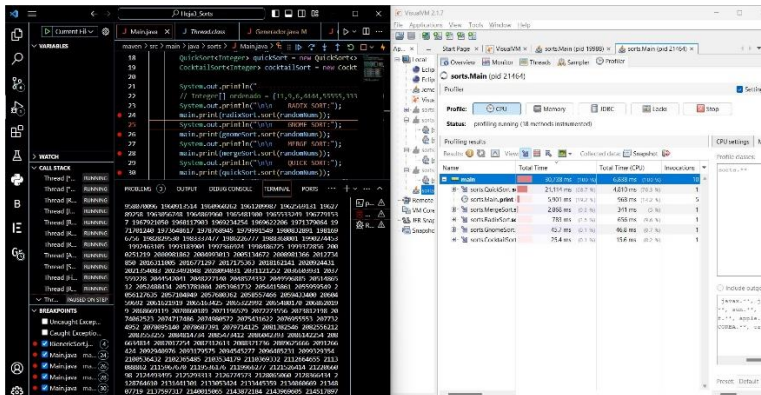
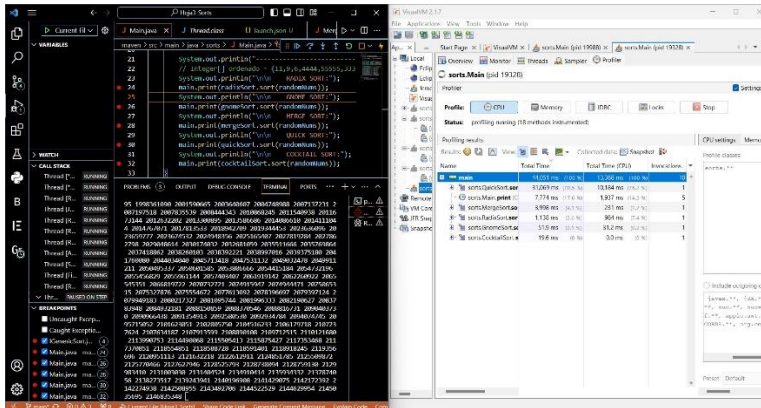
VISUALVM:

Herramienta de software utilizada para medir y analizar el rendimiento de un programa. Brinda información sobre el tiempo de ejecución de secciones del código. VisualVM es una herramienta de monitoreo, depuración y análisis de rendimiento. VisualVM puede conectarse a aplicaciones Java en ejecución.

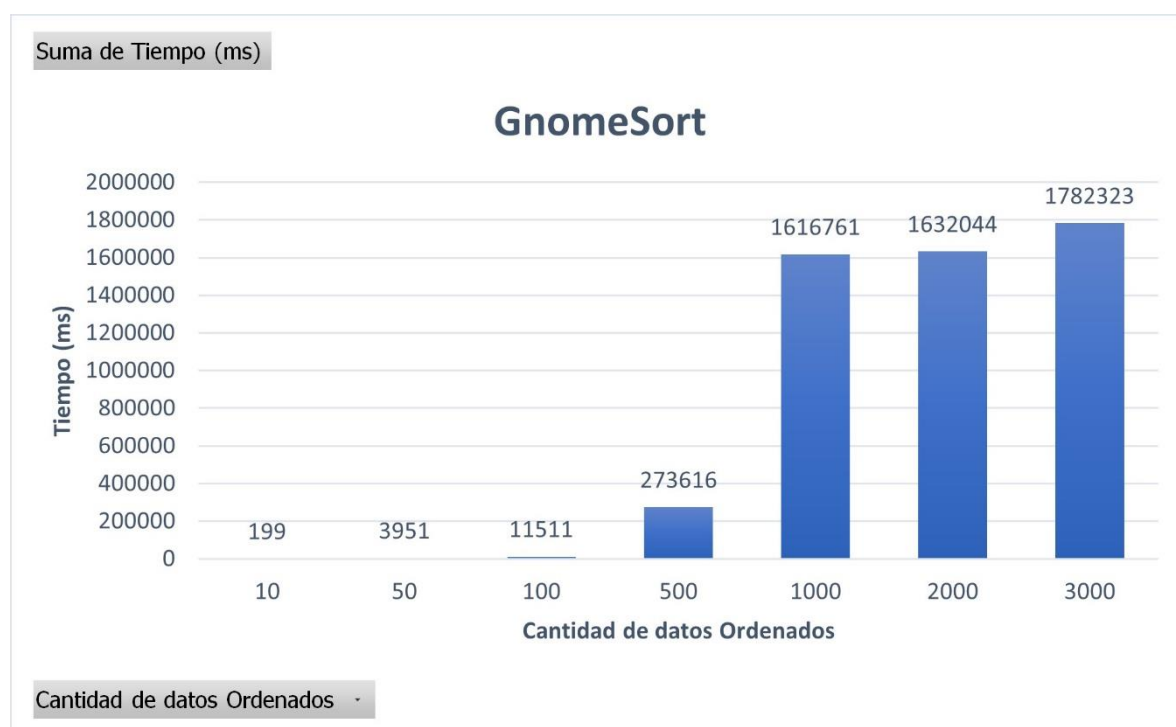
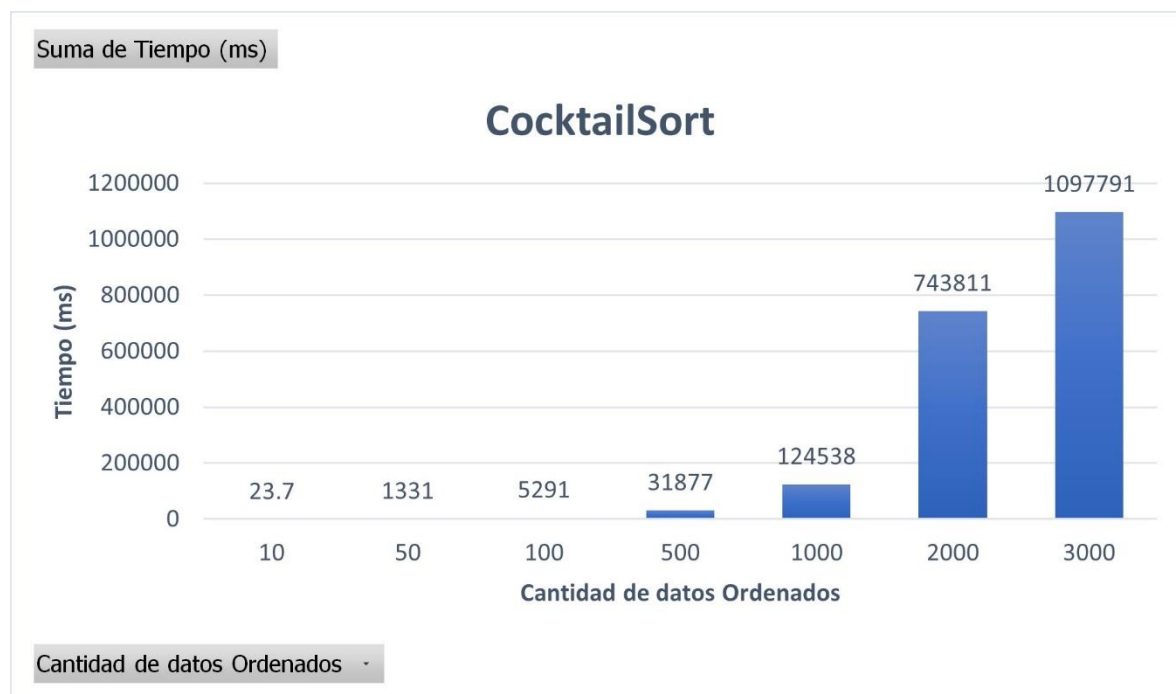
Para implementar VisualVM se instaló el zip del sitio web oficial, se extrajo el programa y seguidamente se conectó la aplicación con Visual Studio Code, para ello se necesitó instalar la extensión de GraalVM.

Al estar conectada la aplicación se elige el tipo de profiling que se vaya a realizar. Se sabe que la aplicación está conectada ya que en la terminal se muestra que el profiler está analizando datos. Se corre el programa en forma de Debug y se evidencia el tiempo que se tarda en ejecutar en VisualVM. El programa comenzará a recopilar datos de rendimiento en los que se verán reflejados los datos del Profiler.

Vista del profiler de VisualVM en ejecución:

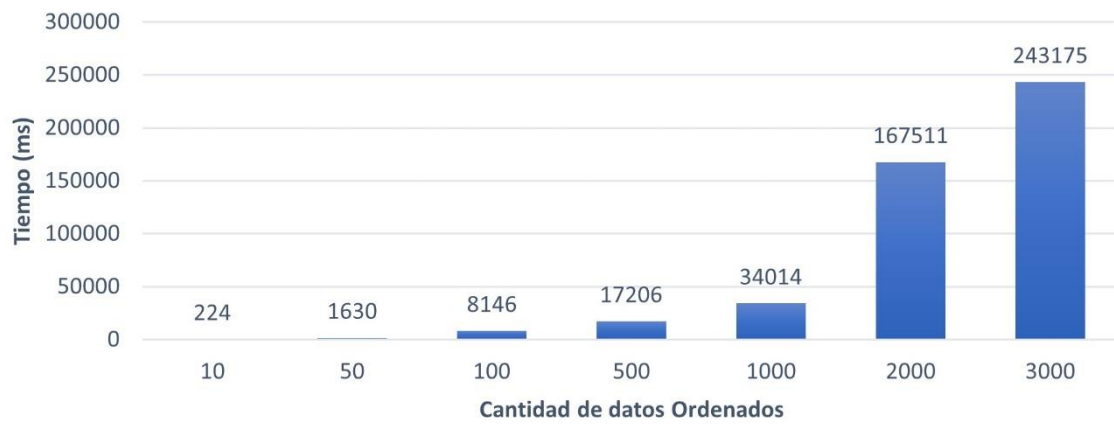


Gráficos Colección de Datos No Ordenados:



Suma de Tiempo (ms)

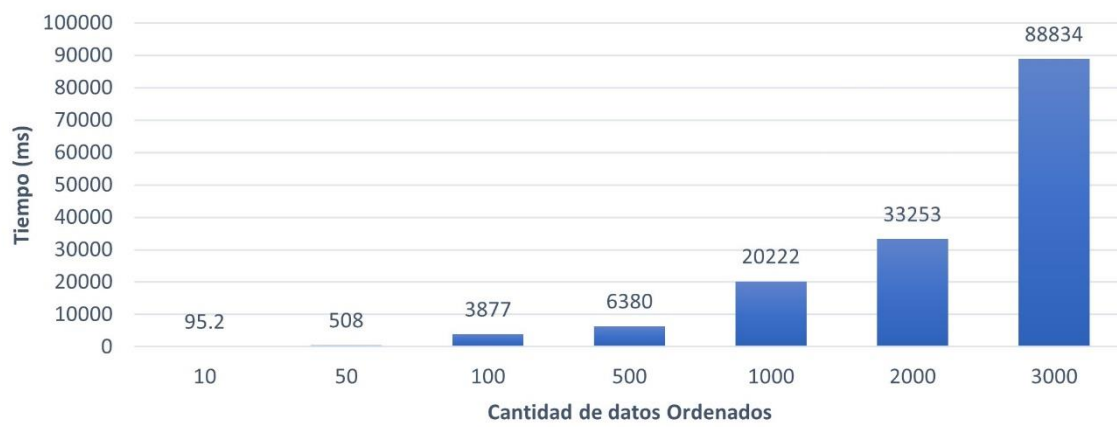
MergeSort



Cantidad de datos Ordenados

Suma de Tiempo (ms)

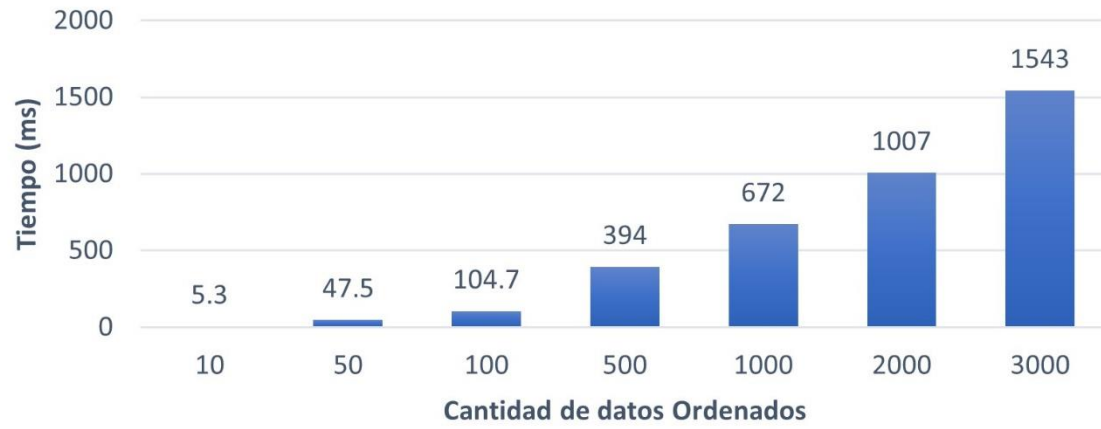
QuickSort



Cantidad de datos Ordenados

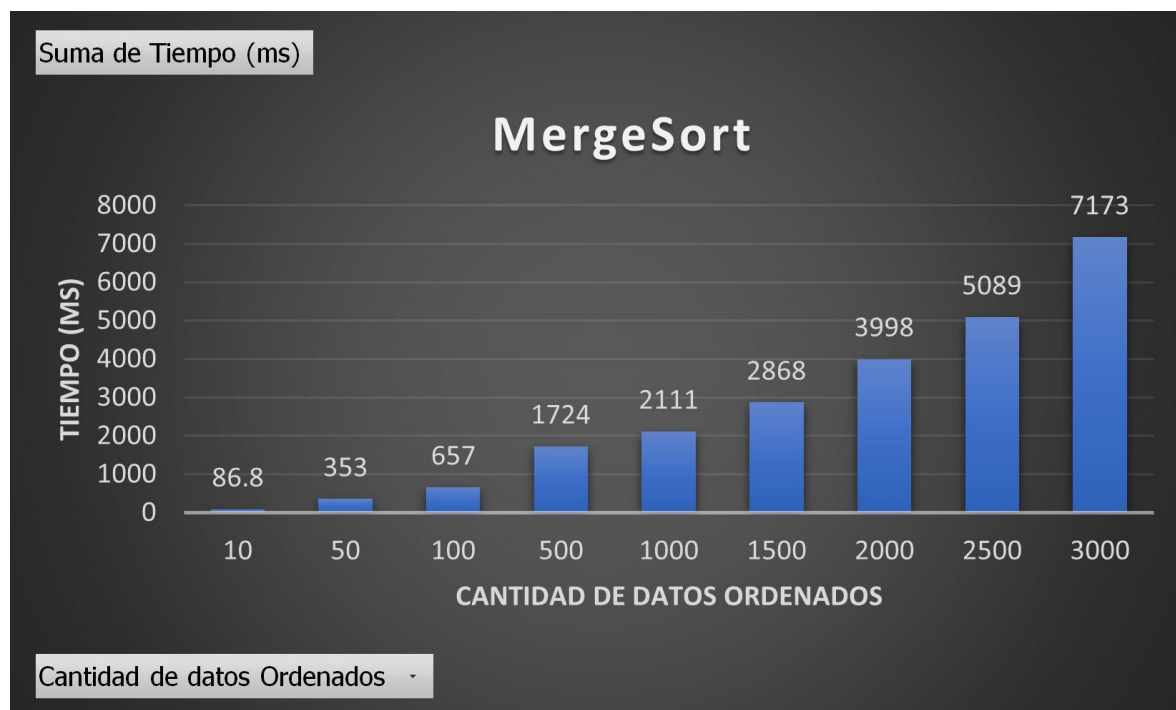
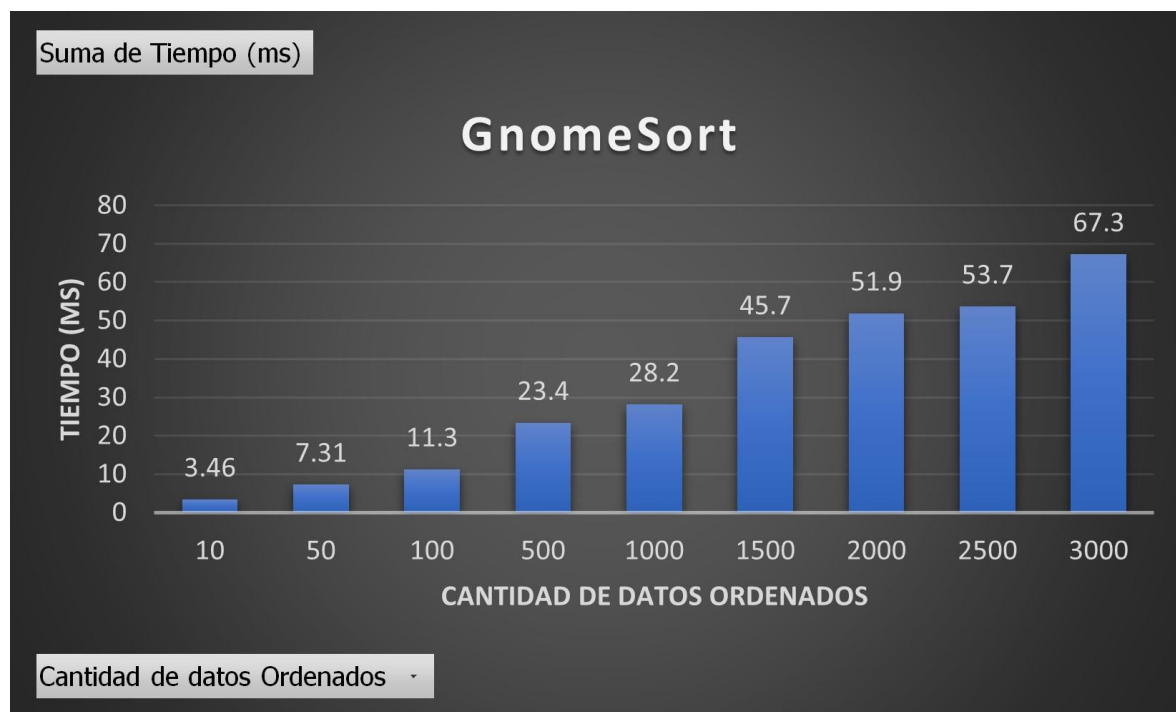
Suma de Tiempo (ms)

RadixSort



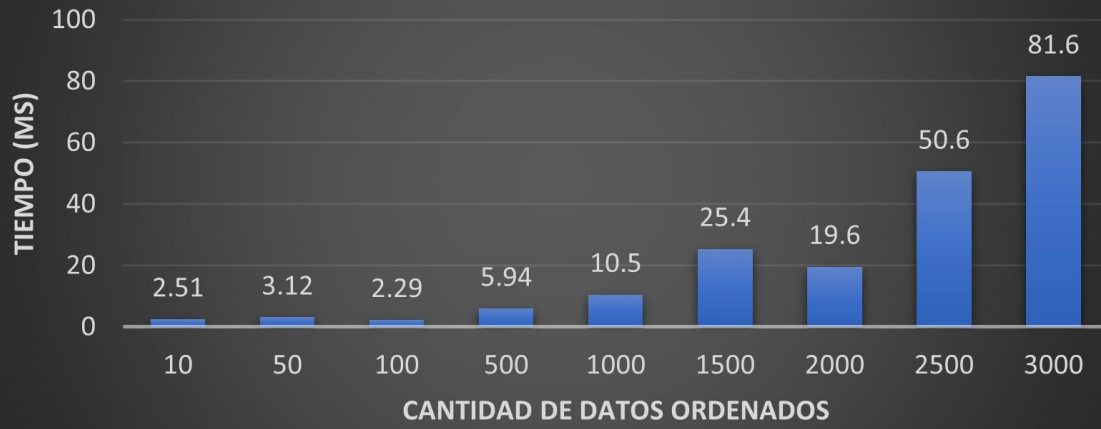
Cantidad de datos Ordenados ▾

Gráficos Colección de Datos Ordenados:



Suma de Tiempo (ms)

CocktailSort



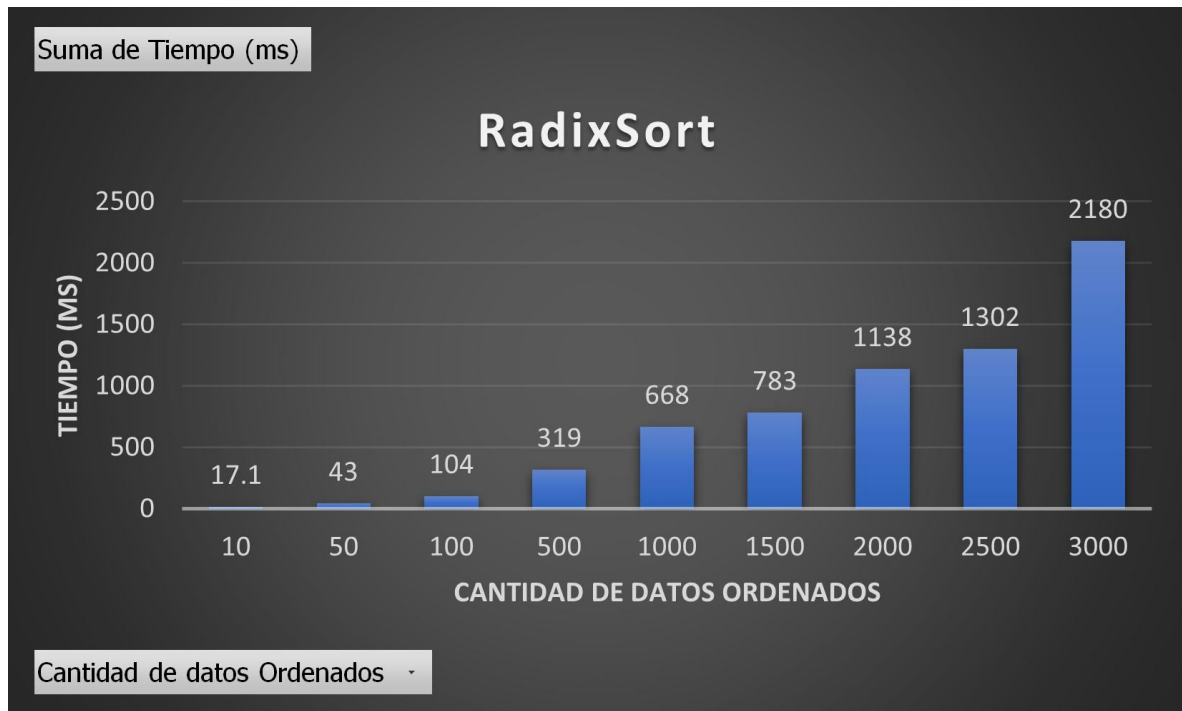
Cantidad de datos Ordenados ▾

Suma de Tiempo (ms)

QuickSort



Cantidad de datos Ordenados ▾



Complejidad de los Algoritmos Implementados:

- → Cocktail Sort: $O(n^2)$
- → Merge Sort: $O(n \log n)$
- → Quicksort: $O(n^2)$
- → Radix Sort: $O(nk)$
- → Gnome Sort: $O(n^2)$