

Reporte Laboratorio 4 - CI2692

Miguel Salomon – Carnet: 1910274

Esta prueba se realizó en un computador con las siguientes especificaciones:

-Procesador: AMD Ryzen 5 5600H 3.30GHz

-16 gb de RAM

- Versión del compilador de Kotlin: 1.8.21

- Versión de la JVM: 11.0.19

Para un arreglo de tamaño 500000:

La primera prueba la realizamos la prueba con el siguiente comando: `“./Sortlib.sh -n 500000 -t 10 -s random”`.

Donde el parámetro -n es el tamaño del arreglo, -t es la cantidad de veces a ordenar el mismo arreglo y -s es el tipo de secuencia del arreglo, en este caso “random”.

Los resultados son los siguientes:

Algoritmo utilizado	Tiempo promedio (segundos)	Desviación estándar (segundos)
QuickSort Clásico	0.0889805459	0.0176992118834
QuickSort – 3 particiones	0.07892782479	0.0149307578020
QuickSort – Doble pivote	0.0859254163	0.01884809882390

Para un arreglo de tamaño 1000000:

La primera prueba la realizamos la prueba con el siguiente comando: `“./Sortlib.sh -n 1000000 -t 10 -s random”`.

Donde el parámetro -n es el tamaño del arreglo, -t es la cantidad de veces a ordenar el mismo arreglo y -s es el tipo de secuencia del arreglo, en este caso “random”.

Los resultados son los siguientes:

Algoritmo utilizado	Tiempo promedio (segundos)	Desviación estándar (segundos)
QuickSort Clásico	0.1834926972	0.0253845129419
QuickSort – 3 particiones	0.1792387474	0.0267355723687
QuickSort – Doble pivote	0.160850076	0.02347747327759

Para un arreglo de tamaño 1500000:

La primera prueba la realizamos la prueba con el siguiente comando: “./Sortlib.sh -n 1500000 -t 10 -s random”.

Donde el parámetro -n es el tamaño del arreglo, -t es la cantidad de veces a ordenar el mismo arreglo y -s es el tipo de secuencia del arreglo, en este caso “random”.

Los resultados son los siguientes:

Algoritmo utilizado	Tiempo promedio (segundos)	Desviación estándar (segundos)
QuickSort Clásico	0.28295998089	0.02851540219851173
QuickSort – 3 particiones	0.2704014845	0.03630077502494751
QuickSort – Doble pivote	0.2530488409	0.0378836977129145574

Para un arreglo de tamaño 2000000:

La primera prueba la realizamos la prueba con el siguiente comando: “./Sortlib.sh -n 2000000 -t 10 -s random”.

Donde el parámetro -n es el tamaño del arreglo, -t es la cantidad de veces a ordenar el mismo arreglo y -s es el tipo de secuencia del arreglo, en este caso “random”.

Los resultados son los siguientes:

Algoritmo utilizado	Tiempo promedio (segundos)	Desviación estándar (segundos)
QuickSort Clásico	0.3900014178	0.04675311712235166
QuickSort – 3 particiones	0.4287096954	0.034431936635526635
QuickSort – Doble pivote	0.3479402251	0.03641074523471286

En la siguiente tabla podemos ver el comportamiento de las variantes de Quicksort en un gráfico de tiempo versus tamaño.

