OBSERVACIONES DE LA PRACTICA

Estudiante 1:Juan Felipe García 202014961 jf.garciam1

Estudiante 2: Santiago Rodríguez 202020476 s.rodriguez64

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Intel(R) Core (TM) i5-10300h CPU @2.50GHZ 2.50GHZ | Intel(R) Core(TM) i7-1065G7 CPU @ 1.30GHz 1.50GHz |
| Memoria RAM (GB) | 8Gb | 12 GB |
| Sistema Operativo | Windows 10 64-bits | Windows 10 Home Single Language 64-Bits |

Tabla 1. Especificaciones de las máquinas para ejecutar las pruebas de rendimiento.

# **Maquina 1**

## **Resultados**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tamaño de la muestra (ARRAYLIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 1125,00 | 1343,75 | 62,50 | 46,88 | 31,25 |
| 2000 | 4671,88 | 5375,00 | 140,63 | 93,75 | 93,75 |
| 4000 | 18265,63 | 21531,25 | 312,50 | 243,38 | 203,13 |
| 8000 | 76390,63 | 87875,00 | 750,00 | 453,13 | 453,13 |
| 16000 | 292234,38 | 359734,38 | 1796,88 | 937,50 | 968,75 |
| 32000 | 1219656,25 | 1445171,88 | 4000,00 | 2125,00 | 2078,13 |
| 64000 | 4945706,10 | 5925204,70 | 10000,00 | 4531,25 | 4531,25 |
| 128000 | tiempo excesivo | tiempo excesivo | 24343,75 | 9859,38 | 9531,25 |
| 256000 | tiempo excesivo | tiempo excesivo | 59453,13 | 22734,38 | 20171,88 |
| 512000 | tiempo excesivo | tiempo excesivo |  | 34671,88 | 30531,25 |

Tabla 2. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación arreglo.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tamaño de la muestra (LINKED\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 70109,38 | 62312,50 | 3734,38 | 2421,88 | 359,38 |
| 2000 | 581312,50 | 510843,75 | 15890,63 | 13312,50 | 1328,13 |
| 4000 | 2324085,38 | 21097846,88 | 75359,38 | 57703,13 | 5312,50 |
| 8000 | 9528750,04 | 8312551,66 | 3763795,04 | 227156,25 | 21156,25 |
| 16000 | tiempo excesivo | tiempo excesivo | tiempo excesivo | 958281,25 | 84750,00 |
| 32000 | tiempo excesivo | tiempo excesivo | tiempo excesivo | Tiempo excesivo | 312684,78 |
| 64000 | tiempo excesivo | tiempo excesivo | tiempo excesivo | Tiempo excesivo | 1602383,00 |
| 128000 | tiempo excesivo | tiempo excesivo | tiempo excesivo | Tiempo excesivo | Tiempo excesivo |
| 256000 | tiempo excesivo | tiempo excesivo | tiempo excesivo | Tiempo excesivo | Tiempo excesivo |
| 512000 | tiempo excesivo | tiempo excesivo | tiempo excesivo | Tiempo excesivo | Tiempo excesivo |

Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| Merge sort | mejor | Mejor |
| Quick sort | peor | peor |

Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 1.**
  + Comparación de rendimiento ARRAYLIST.
  + Comparación de rendimiento LINKED\_LIST.
  + Comparación de rendimiento para Insertion Sort.
  + Comparación de rendimiento para Selection Sort.
  + Comparación de rendimiento para Shell Sort.
  + Comparación de rendimiento para MergeSort.
  + Comparación de rendimiento para QuickSort.

# **Maquina 2**

## **Resultados**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tamaño de la muestra (ARRAYLIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 625 | 859,38 | 46,88 | 26,04 | 57,29 |
| 2000 | 2588,54 | 3291,64 | 78,13 | 57,29 | 93,75 |
| 4000 | 9822,92 | 13031,25 | 187,5 | 125 | 140,63 |
| 8000 | 45604,17 | 57104,17 | 458,33 | 223,96 | 270,83 |
| 16000 | 188494,79 | 230677,08 | 1197,92 | 468,75 | 552,08 |
| 32000 | 812739,58 | 938901,04 | 2494,79 | 1036,46 | 1109,38 |
| 64000 | Exceso de tiempo | Exceso de tiempo | 6078,13 | 2250 | 2343,75 |
| 128000 | Exceso de tiempo | Exceso de tiempo | 14317,71 | 5166,67 | 5026,04 |
| 256000 | Exceso de tiempo | Exceso de tiempo | 35562,5 | 11890,63 | 10177,08 |
| 512000 | Exceso de tiempo | Exceso de tiempo |  |  |  |

Tabla 2. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación arreglo.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tamaño de la muestra (LINKED\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 48828,13 | 43255,21 | 2609,38 | 1177,08 | 177,08 |
| 2000 | 398729,17 | 371757,81 | 10687,5 | 6432,29 | 656,25 |
| 4000 | 3223656.25 | 2846437,5 | 53473,96 | 29973,96 | 2692,71 |
| 8000 | Exceso de tiempo | Exceso de tiempo | 261890,63 | 158635,42 | 11828,125 |
| 16000 | Exceso de tiempo | Exceso de tiempo | 1241281.25 | 477390.63 | 43348,96 |
| 32000 | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo | 2150718.75 | 168906,25 |
| 64000 | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo | 728328,125 |
| 128000 | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo |
| 256000 | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo |
| 512000 | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo | Exceso de tiempo |

Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| Merge sort |  | Mejor |
| Quick sort | Mejor |  |

Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 2.**
  + Comparación de rendimiento ARRAYLIST.
  + Comparación de rendimiento LINKED\_LIST.
  + Comparación de rendimiento para Insertion Sort.
  + Comparación de rendimiento para Selection Sort.
  + Comparación de rendimiento para Shell Sort.
  + Comparación de rendimiento para MergeSort.
  + Comparación de rendimiento para QuickSort.

# **Preguntas de análisis**

1. ¿El comportamiento de los algoritmos es acorde a lo enunciado teóricamente?

Los Algoritmos actúan acordes a lo planteado como se enuncia teóricamente, pues en general los más eficientes son quick y merge con complejidad O(nlogn), de hecho, en Array son casi idénticos. En linked quick se demora más dada la dificultad de recorrer la lista. Además, como se dijo en el laboratorio anterior, los algoritmos iterativos también siguen el comportamiento teórico.

1. ¿Existe alguna diferencia entre los resultados obtenidos al ejecutar las pruebas en diferentes máquinas?

Si existen diferencias, la maquina 2 registra mejores y distintos tiempos comparados con la maquina 1.

1. De existir diferencias, ¿A qué creen ustedes que se deben dichas diferencias?

Las diferencias probablemente se dan debido a la diferencia en procesadores y memoria ram (8GB y 12GB). En ambos casos la maquina 2 es mejor.

1. ¿Cuál Estructura de Datos es mejor utilizar si solo se tiene en cuenta los tiempos de ejecución de los algoritmos?

ARRAYLIST es mejor que LINKED\_LIST en cada algoritmo, tanto en la teoría como en la práctica. Si se quisiera ser lo más eficiente se debe usar el algoritmo Merge.

1. Para el caso analizado de ordenamiento de los videos, teniendo en cuenta los resultados de tiempo reportados por todos los algoritmos de ordenamiento estudiados (iterativos y recursivos), proponga un ranking de los algoritmos de ordenamiento (de mayor eficiencia a menor eficiencia en tiempo) para ordenar la mayor cantidad de videos.
2. Merge
3. Quick
4. Shell
5. Selection
6. Insertion