OBSERVACIONES DE LA PRACTICA

Fernando Álvarez 202014988

johan Stivens Ardila Tapiero201911400

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Intel® Core™ i7 – 10510U CPU@ 1.80 GH 2.30 GH | Intel(R) Core(TM) i5-8300H CPU 2.30GHZ 2.30 GHZ |
| Memoria RAM (GB) | 16 GB | 8 GB |
| Sistema Operativo | Windows 10 | Windows 10 Home Single |

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]** |
| 100 | 94 | 62,5 | 16 |  |  |
| 200 | 312,5 | 312,5 | 47 |  |  |
| 400 | 2343,75 | 3375 | 234 |  |  |
| 800 | 35.734 | 43531,25 | 1218,75 |  |  |
| 1600 | 166281,25 | 232.859 | 6.078 | 656,25 | 4375 |
| 2000 |  |  | 9.516 | 843,75 | 6.891 |
| 4000 |  |  | 54.188 | 3.297 | 33468,75 |
| 8000 |  |  | 243.016 | 14.484 | 162.828 |
| 16000 |  |  |  | 57.422 | 730437,5 |

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]** |
| 100 | 62,5 | 47 | 16 |  |  |
| 200 | 312,5 | 297 | 47 |  |  |
| 400 | 24562,5 | 2.484 | 250 |  |  |
| 800 | 2937,5 | 20609,375 | 1187,5 |  |  |
| 1600 | 195.141 | 176562,5 | 6093,75 | 500 |  |
| 2000 |  |  | 10437,5 | 937,5 |  |
| 4000 |  |  | 66781,25 | 4.141 |  |
| 8000 |  |  | 271.922 | 18.609 |  |
| 16000 |  |  |  | 67250 |  |

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 | X | X |
| Quick sort |  |  |

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]** |
| 100 | 47 | 46.875 | 15.625 |  |  |
| 200 | 281.25 | 265.625 | 46.875 |  |  |
| 400 | 2765.625 | 2109.375 | 203.125 |  |  |
| 800 | 19843.75 | 17046.875 | 1046.875 |  |  |
| 1600 | 155609.375 | 142906.25 | 5281.25 | 406.25 | 3546.875 |
| 2000 |  |  | 8125 | 718.75 | 5671.875 |
| 4000 |  |  | 45421.875 | 2781.25 | 27281.25 |
| 8000 |  |  | 191093.75 | 11031.25 | 126640.625 |
| 16000 |  |  |  | 45968.75 | 541000 |

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]** |
| 100 | 31.25 | 31.25 | 15.625 |  |  |
| 200 | 250 | 234.375 | 46.875 |  |  |
| 400 | 2234.375 | 1968.75 | 187.5 |  |  |
| 800 | 19859.375 | 16015.625 | 953.125 |  |  |
| 1600 | 146562.5 | 134875 | 4578.125 | 406.25 | 3718.75 |
| 2000 |  |  | 8156.25 | 671.875 | 5531.25 |
| 4000 |  |  | 46625 | 2796.875 | 27578.125 |
| 8000 |  |  | 188984.375 | 10968.75 | 122390.625 |
| 16000 |  |  |  | 43859.375 | 533812.5 |

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 | X | X |
| Quick sort |  |  |

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?

No se obtuvo una medida lo suficiente mente fiel a lo enunciado teóricamente, la complejidad del algoritmo pudo verse afectada por la forma de implementación la lectura de datos del catálogo, los altos recursos para ordenar datos de más de 20000 datos se excede un tiempo más que significativo.

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

No una diferencia significativa, sin embargo, al compararlos en cuanto a Shellsort y Mergeshort hubo una pequeña diferencia, en la cual la máquina 1 presentó un mejor rendimiento que la 2 en un Arraylist que en un Linked List. Por otra parte, en Insertionsort hubo un comportamiento de crecimiento, decrecimiento y crecimiento en Linked List para la máquina 1, mientras que para la 2, esta siempre se mantuvo en un crecimiento exponencial.

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

Las diferencias en hardware, dadas principalmente por el tipo de procesador y por la RAM.

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

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.

* MERGE SORT
* SHELL SORT
* SELECTION SORT
* INSERTION SORT
* QUIK SORT