OBSERVACIONES DE LA PRACTICA

Santiago Duque Cod 202021020 (Máquina 1)

Nicolás Guerrero Cod 201731839 (Máquina 2)

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Intel® Core™ i7-10510U CPU @ 1.80GHz 2.30 GHz | 1.6 GHz Dual-Core Intel Core i5 |
| Memoria RAM (GB) | 16,0 GB | 8 GB 1600 MHz DDR3 |
| Sistema Operativo | Windows 10 Home 64-bits | MacOS Big Sur versión 11.1 |

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 | 640.63 | 640.63 | 1546.88 | 1130.21 | 36.46 |
| 2000 | 2343.75 | 2382.81 | 6382.81 | 4343.75 | 78.13 |
| 4000 | 9437.50 | 9359.38 | 24906.25 | Error | 130.21 |
| 8000 | 37479.69 | 37632.81 | 99882.81 |  | 244.79 |
| 16000 | 155734.38 | 156195.31 | 412984.38 |  | 447.92 |
| 32000 | 671187.50 | 670125.00 | --- |  | 916.67 |
| 64000 | --- | --- | --- |  | 1911.46 |
| 128000 | --- | --- | --- |  | 4052.08 |
| 256000 | --- | --- | --- |  | 8463.54 |
| 512000 | --- | --- | --- | --- | --- |

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 | 48703.13 | 47578.13 | 103312.5 | 62562.67 | 140.63 |
| 2000 | 389125.00 | 388093.75 | --- | 525375.00 | 554.69 |
| 4000 | --- | --- | --- | --- | 2234.38 |
| 8000 | --- | --- | --- | --- | 9105.21 |
| 16000 | --- | --- | --- | --- | 36390.63 |
| 32000 | --- | --- | --- | --- | 145828.13 |
| 64000 | --- | --- | --- | --- | 603468.75 |
| 128000 | --- | --- | --- | --- | --- |
| 256000 | --- | --- | --- | --- | --- |
| 512000 | --- | --- | --- | --- | --- |

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 |  |  |
| 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]** |
| 1000 | 2282.90 | 2329.52 | 127.95 | 4292.84 | 85.2 |
| 2000 | 9284.63 | 10455.545 | 249.82 | 17802.56 | 170.71 |
| 4000 | 38088.55 | 38380.29 | 571.78 | 70713.462 | 346.09 |
| 8000 | 156003.92 | 155501.46 | 1381.75 | 291352.31 | 735.78 |
| 16000 | 631197.53 | 650372.26 | 3346.01 | 1131074.02 | 1563.23 |
| 32000 |  |  | 7176.43 |  | 3369.58 |
| 64000 |  |  | 17734.72 |  | 7040.93 |
| 128000 |  |  | 43265.97 |  | 14842.94 |
| 256000 |  |  | 105838.68 |  | 31503.664 |
| 512000 |  |  |  |  |  |

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 | 120362.36 | 125226.19 | 7500.41 | 238592.40 | 574.20 |
| 2000 | 1203437.58 | 1011484.406 | 30911.665 | 1973669.37 | 2174.15 |
| 4000 |  |  | 147462.29 |  | 8988.82 |
| 8000 |  |  |  |  | 34125.66 |
| 16000 |  |  |  |  | 140561.42 |
| 32000 |  |  |  |  | 592541.92 |
| 64000 |  |  |  |  |  |
| 128000 |  |  |  |  |  |
| 256000 |  |  |  |  |  |
| 512000 |  |  |  |  |  |

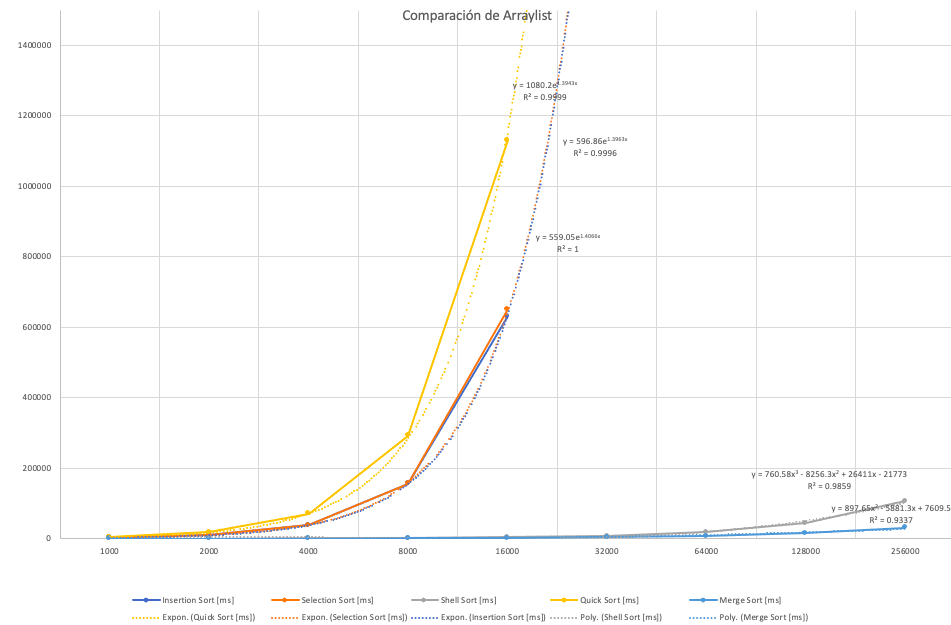
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

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

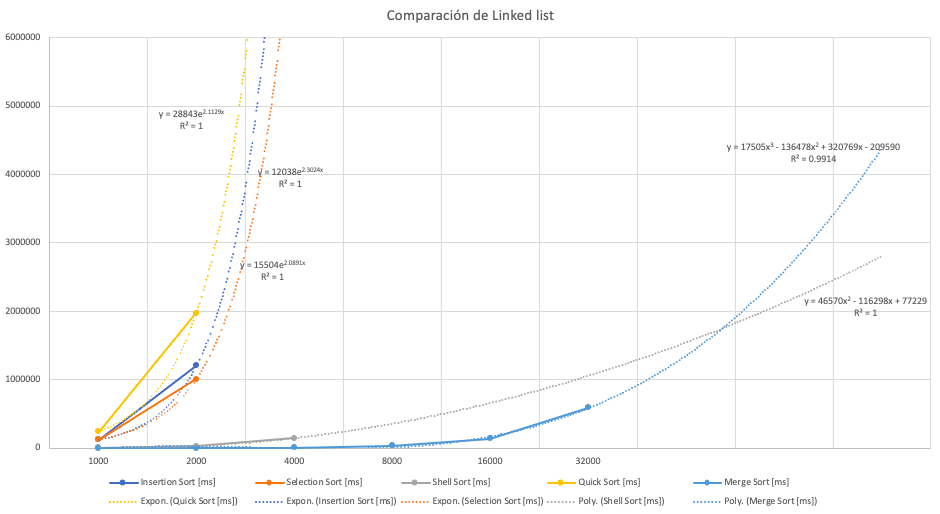
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

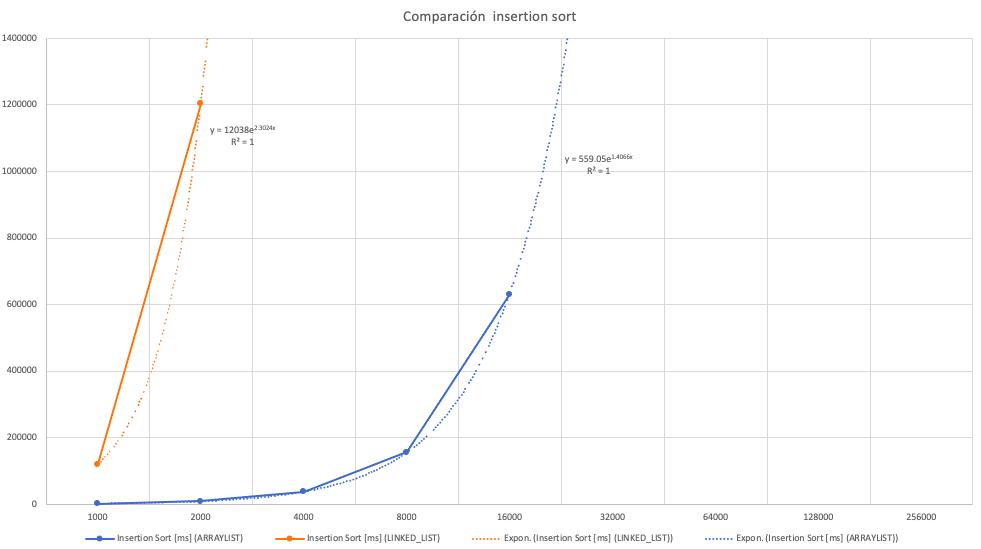
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 2.**
  + Comparación de rendimiento ARRAYLIST.



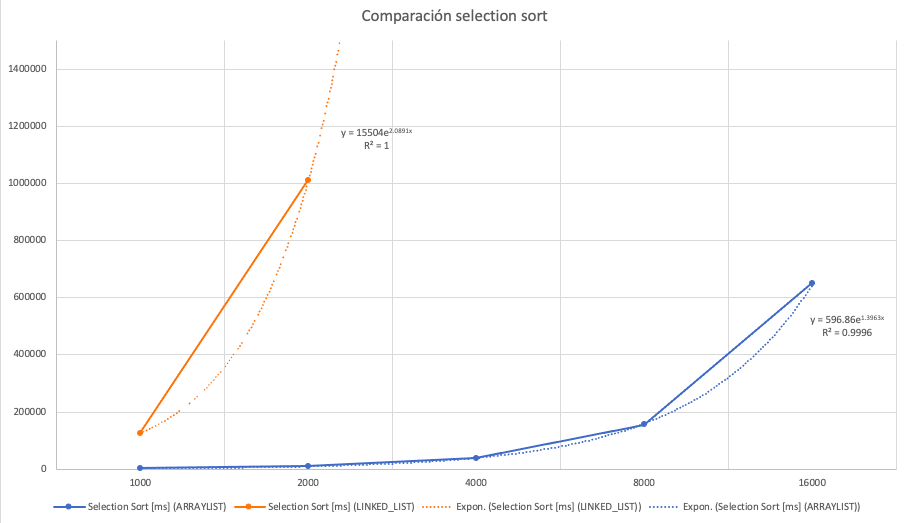
* + Comparación de rendimiento LINKED\_LIST.



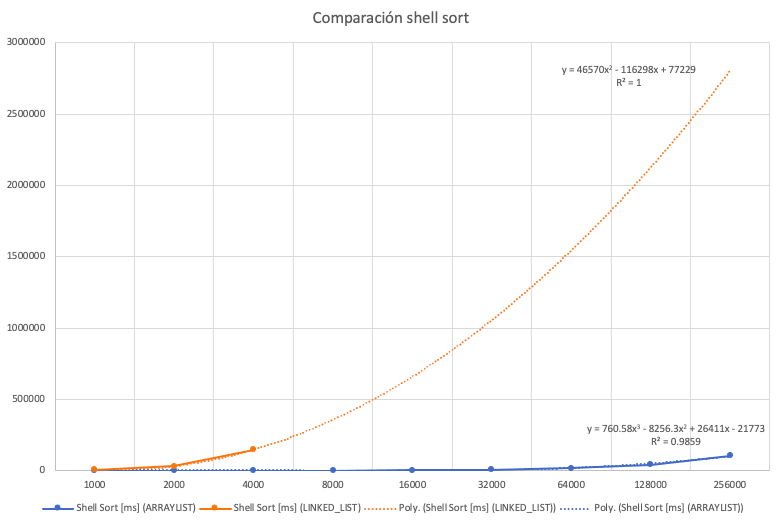
* + Comparación de rendimiento para Insertion Sort.



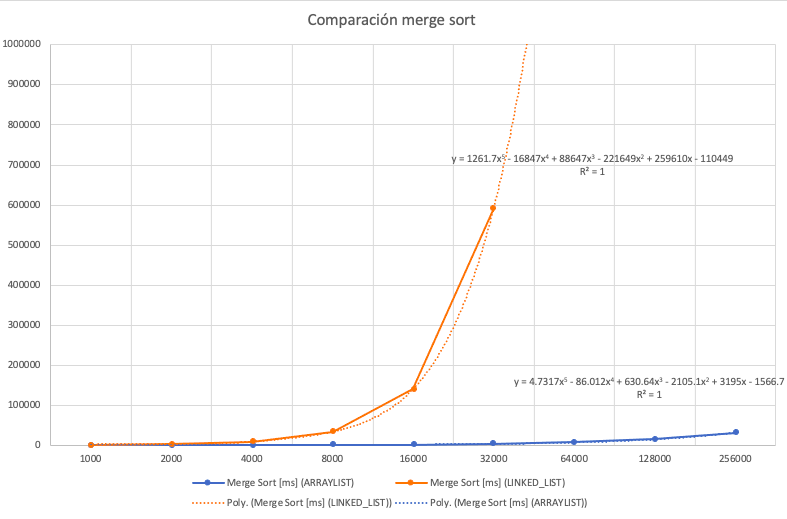
* + Comparación de rendimiento para Selection Sort.



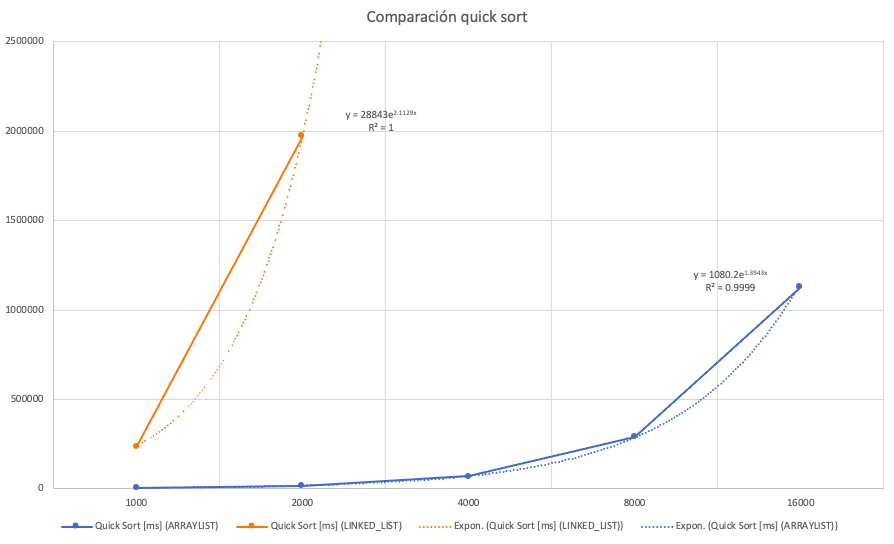
* + Comparación de rendimiento para Shell Sort.



* + 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?

De forma general, los ordenamientos recursivos tuvieron un mejor rendimiento a excepción del quick sort. El merge sort se comporto mucho mejor que todos los otros algoritmos de ordenamiento por lo que se cumple lo previsto por la teoría. Esta anomalia en el quick sort se puede deber a errores en el codigo.

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

Diferentes maquinas con diferentes procesadores y memoria ram hacen que en algunos computadores los algoritmos tarden menos tiempo en hacer las operaciones. A parte de esto, el comportamiento general es el mismo, independientemente de la maquina que se utilice.

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

Estas diferencias se deben a que una maquina puede tener un procesador más potente o mas memoria ram disponible por lo que esta se demora menos en hacer ciertas operaciones y puede correr los algoritmos de forma mas rapida y eficiente.

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

La estructura de datos que mejor se comporto en todos los algoritmos fue el arreglo por lo que si este es el único criterio, el array es el que se debe utilizar.

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 Sort
3. Selection Sort
4. Incertion Sort
5. Quick Sort
6. Shell Sort