OBSERVACIONES DE LA PRACTICA

Valentina Calderón Cod 202020771

Maria Castro Cod 202020850

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Procesador 1,4 GHz Intel Core i5 de dos núcleos | 2.3 GHz Dual-Core Intel Core i5 |
| Memoria RAM (GB) | Memoria 4 GB 1600 MHz DDR3 | 8 GB 2133 MHz LPDDR3 |
| Sistema Operativo | macOS Big Sur 11.1 | macOS Big Sur 11.2.1 |

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 | 920.02 | 946.14 | 51.15 | 51,42 | 47,66 |
| 2000 | 3927.01 | 4079.46 | 121.80 | 114,33 | 108,30 |
| 4000 | 15427.89 | 17247.38 | 275.58 | 215,87 | 210,80 |
| 8000 | 68396.08 | 66343.13 | 656.94 | 502,23 | 432,72 |
| 16000 | 258369.98 | 287726.42 | 1684.76 | 1158,17 | 1010,58 |
| 32000 |  | 1288559.08 | 4408.53 | 2304,77 | 2378,86 |
| 64000 |  | 4616267.51 | 9613.68 | 5324,19 | 4853,78 |
| 128000 |  |  | 23731.46 | 10529,61 | 9850,78 |
| 256000 |  |  | 56710.87 | 22709,20 | 21096,10 |
| 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 | 69208.09 | 62002.68 | 3617.15 | 3397,22 | 340,73 |
| 2000 | 560374.85 | 513080.73 | 17667.02 | 12952,57 | 1441,48 |
| 4000 | 4656357.47 |  | 80878.85 | 56719,10 | 6554,45 |
| 8000 |  |  | 362187.86 | 240779,18 | 22381,87 |
| 16000 |  |  | 1706173.40 |  | 92548,80 |
| 32000 |  |  |  |  | 375320,20 |
| 64000 |  |  |  |  |  |
| 128000 |  |  |  |  |  |
| 256000 |  |  |  |  |  |
| 512000 |  |  |  |  |  |

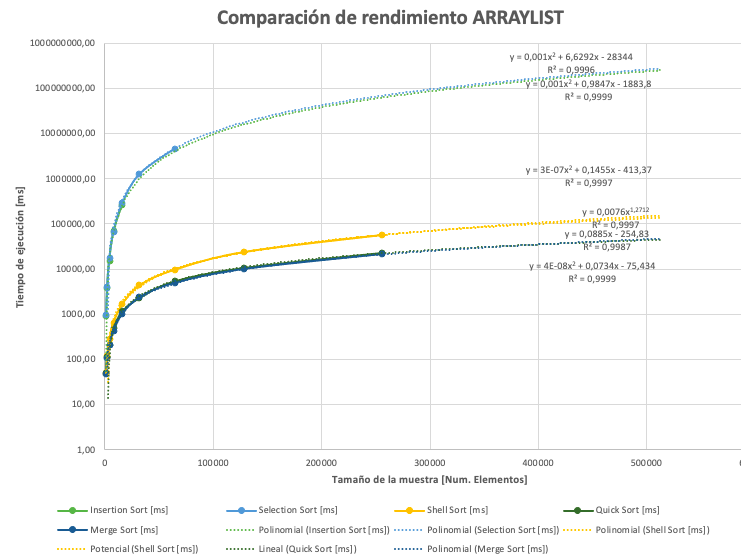
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

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

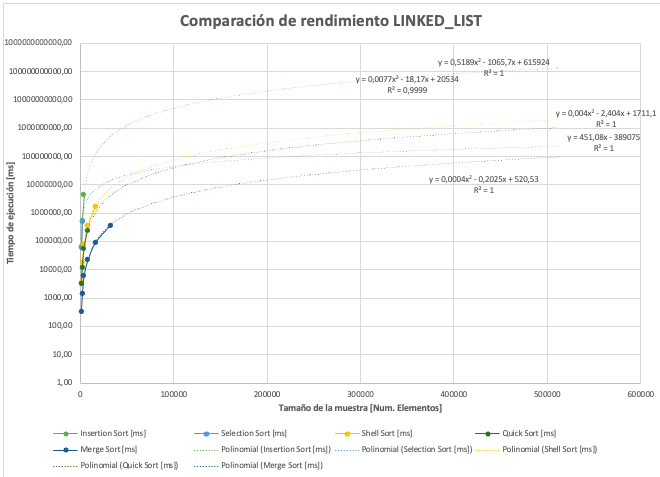
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

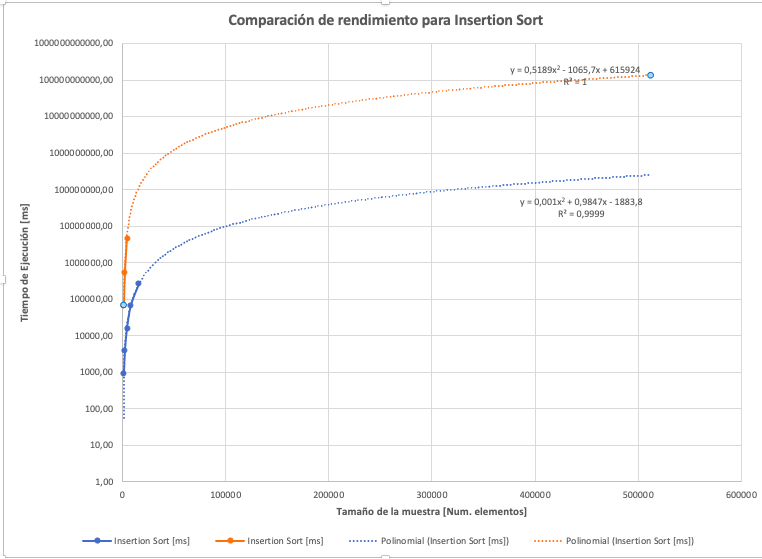
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 1.**
  + Comparación de rendimiento ARRAYLIST.



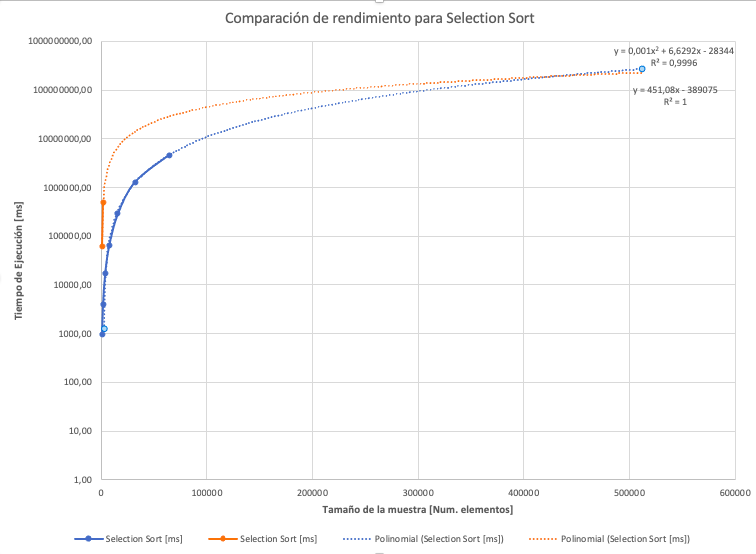
* + Comparación de rendimiento LINKED\_LIST.



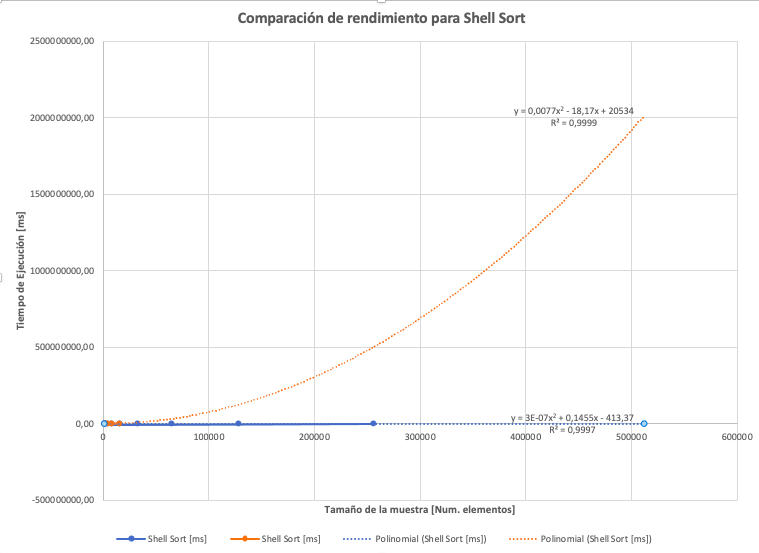
* + Comparación de rendimiento para Insertion Sort.



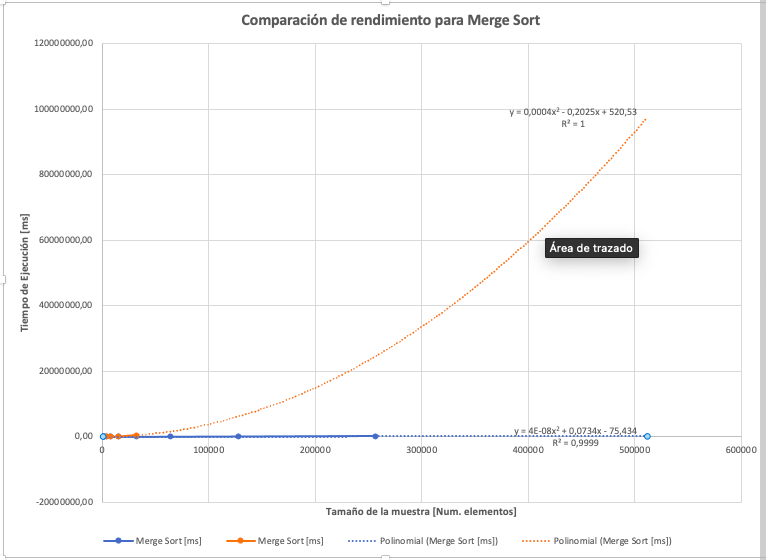
* + Comparación de rendimiento para Selection Sort.

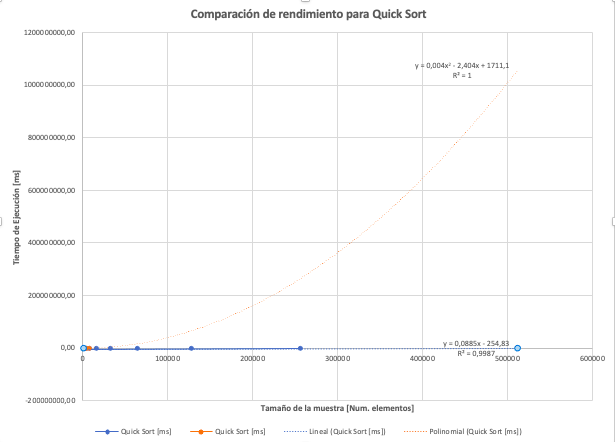


* + Comparación de rendimiento para Shell Sort.



* + 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]** |
| 1000 | 637.60 | 670.63 | 37.69 | 34,07 | 34,38 |
| 2000 | 2713.14 | 2916.61 | 85.10 | 65,27 | 66,27 |
| 4000 | 10936.16 | 10710.49 | 192.73 | 134,46 | 143,88 |
| 8000 | 43601.20 | 44929.52 | 431.34 | 302,79 | 292,93 |
| 16000 | 179375.10 | 186474.09 | 1017.22 | 648,21 | 643,30 |
| 32000 | 771539.28 | 716213.22 | 2438.77 | 1382,01 | 1357,01 |
| 64000 |  | 2890360.03 | 5651.87 | 3079,84 | 2925,99 |
| 128000 |  |  | 10518.68 | 6701,04 | 6243,53 |
| 256000 |  |  | 35082.15 | 14646,49 | 13401,32 |
| 512000 |  |  | 54937.90 | 20441,02 | 20276,78 |

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 | 45575.74 | 41042.89 | 2074.96 | 1883,39 | 225,23 |
| 2000 | 334934.56 | 335315.18 | 11181.70 | 8028,28 | 882,33 |
| 4000 | 2734041.40 | 2721915.21 | 53057.86 | 34889,46 | 3512,46 |
| 8000 |  |  | 238796.86 | 153290,22 | 14143,59 |
| 16000 |  |  | 1119501.94 | 684359,30 | 57111,74 |
| 32000 |  |  |  |  | 232006,70 |
| 64000 |  |  |  |  | 930163,59 |
| 128000 |  |  |  |  |  |
| 256000 |  |  |  |  |  |
| 512000 |  |  |  |  |  |

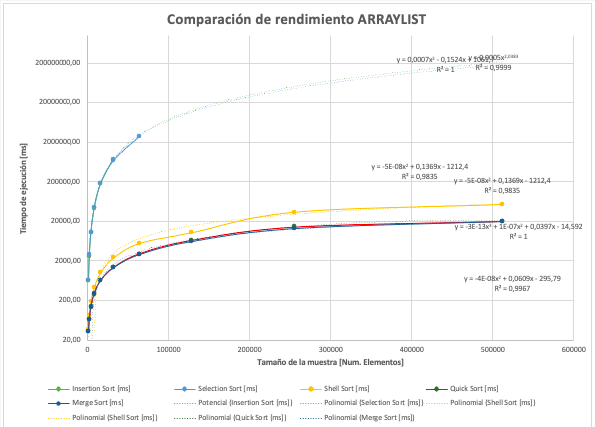
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

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

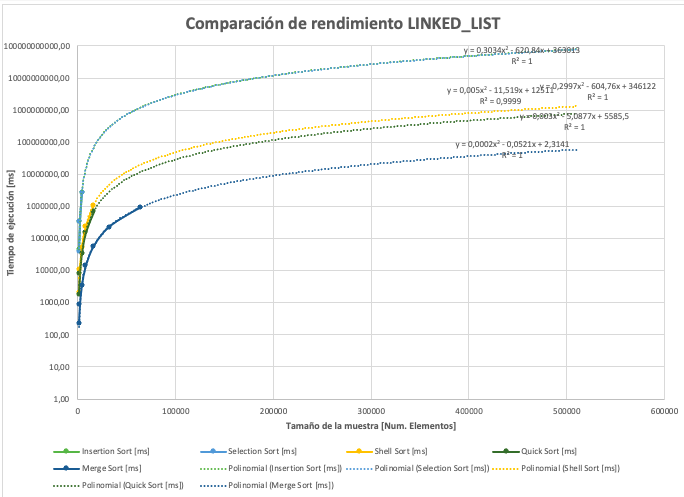
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

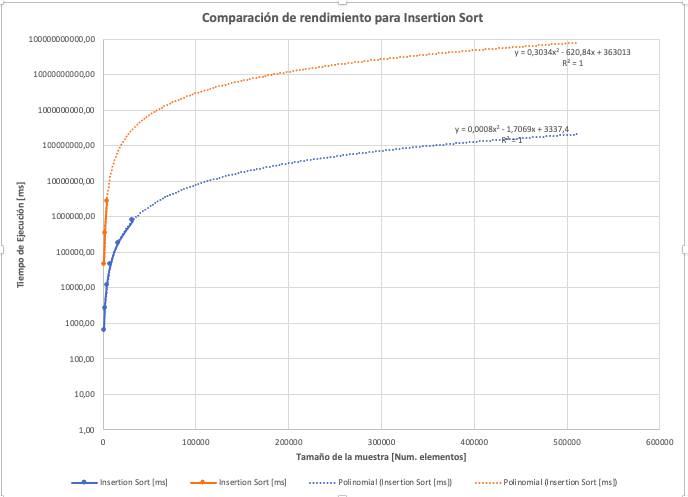
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 2.**
  + Comparación de rendimiento ARRAYLIST.



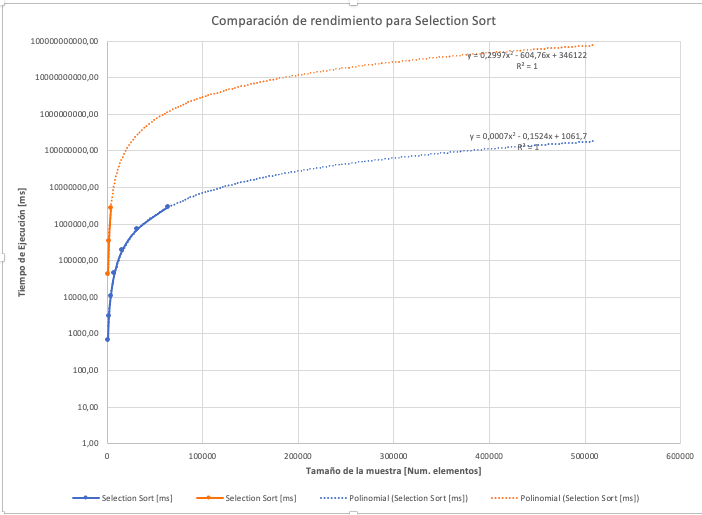
* + Comparación de rendimiento LINKED\_LIST.



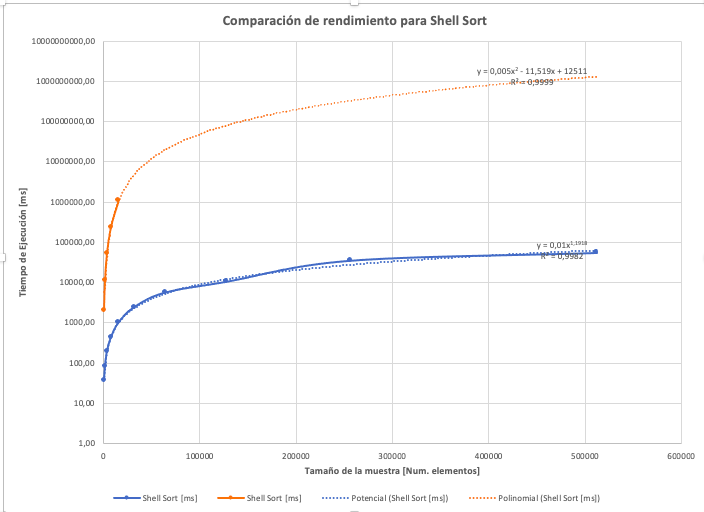
* + Comparación de rendimiento para Insertion Sort.



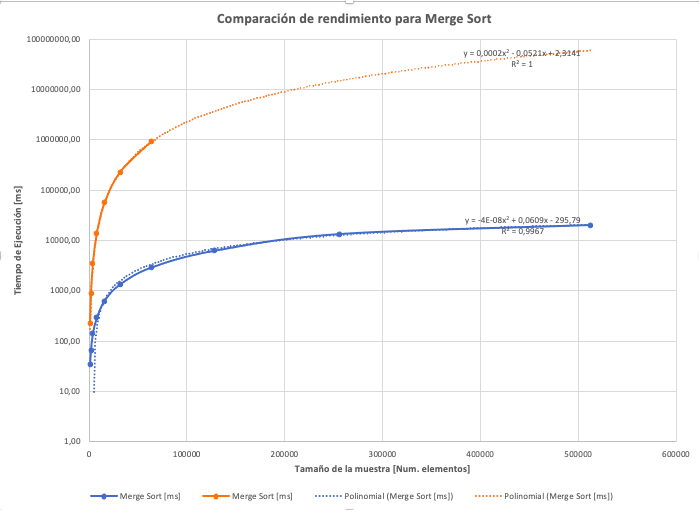
* + Comparación de rendimiento para Selection Sort.



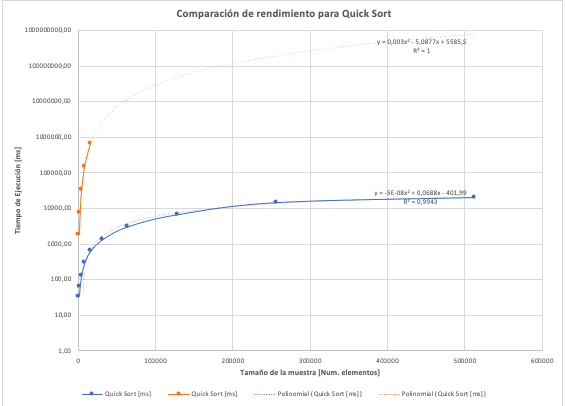
* + Comparación de rendimiento para Shell Sort.



* + 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?

El comportamiento de los algoritmos sí fue acorde a lo enunciado teóricamente, por ejemplo la linea de tendencia del QuickSort evidencia una complejidad de n^2 el cual va acorde a la complejidad en el peor caso.

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

SI

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

La cantidad de RAM, el procesador y antigüedad del computador.

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

En el caso del ArrayList, los dos algoritmos estuvieron muy parejos en cuanto a tiempo de ejecución, sin embargo el Merge fue levemente más rápido. En el caso del LinkedList el Merge fue considerablemente más rápido que el QuickSort.

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.

Ranking de los algoritmos de ordenamiento (Mejor a Peor):

1. MergeSort
2. QuickSort
3. ShellSort
4. InsertionSort
5. SelectionSort