OBSERVACIONES DE LA PRACTICA

Sofía Vásquez 202123910

Isabella Caputi 202122075

Antonia Streubel 202123766

# **Ambientes de pruebas**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Máquina 1 | Máquina 2 | Máquina 3 |
| Procesadores | Apple M1 | Intel(R) Core(TM) i3-1005G1 CPU @ 1.20GHz 1.19 GHz | Intel(R) Core(TM) i3-7100U CPU @ 2.40GHz 2.40 GHz |
| Memoria RAM (GB) | 8,00 GB | 8,00 GB | 8,00 GB |
| Sistema Operativo | macOS Monterey | Sistema operativo de 64 bits, procesador basado en x64 | Sistema operativo de 64 bits, procesador basado en x64 |

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

# **Maquina 1**

## **Resultados**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (ARRAY\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% | 228 | 0.947 | 8.062 | 4.22 | 12,629 | 4,979 |
| 5.00% | 1148 | 4.501 | 126.192 | 6.817 | 224,495 | 13,356 |
| 10.00% | 2298 | 5.768 | 494.761 | 10.882 | 887,814 | 24,024 |
| 20.00% | 4598 | 9.268 | 2002.414 | 22.358 | 3578,23 | 49,976 |
| 30.00% | 6898 | 12.620 | 4577.040 | 36.311 | 5390,11 | 74,075 |
| 50.00% | 11498 | 14.864 | 12634.773 | 62.373 | 15121,21 | 129,212 |
| 80.00% | 18397 | 19.882 | 33056.198 | 100.522 | 38981,61 | 205,79 |
| 100.00% | 22998 | 25.439 | 51959.739 | 137.920 | 60426,90 | 791,769 |

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

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (LINKED\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% | 228 | 3.086 | 36.139 | 5.906 | 54,837 | 7,382 |
| 5.00% | 1148 | 20.496 | 3558.523 | 86.128 | 5659,91 | 54,561 |
| 10.00% | 2298 | 78.408 | 30159.435 | 362.697 | 45474,14 | 185,323 |
| 20.00% | 4598 | 283.634 | 246067.551 | 1682.157 | - | 755,437 |
| 30.00% | 6898 | 636.015 | - | 4008.499 | - | 1681,705 |
| 50.00% | 11498 | 1770.154 | - | 44971.17 | - | 4597,217 |
| 80.00% | 18397 | 4538.412 | - | 97461.88 | - | 11980,219 |
| 100.00% | 22998 | 7177.234 | - | 357546.83 | - | 22058,281 |

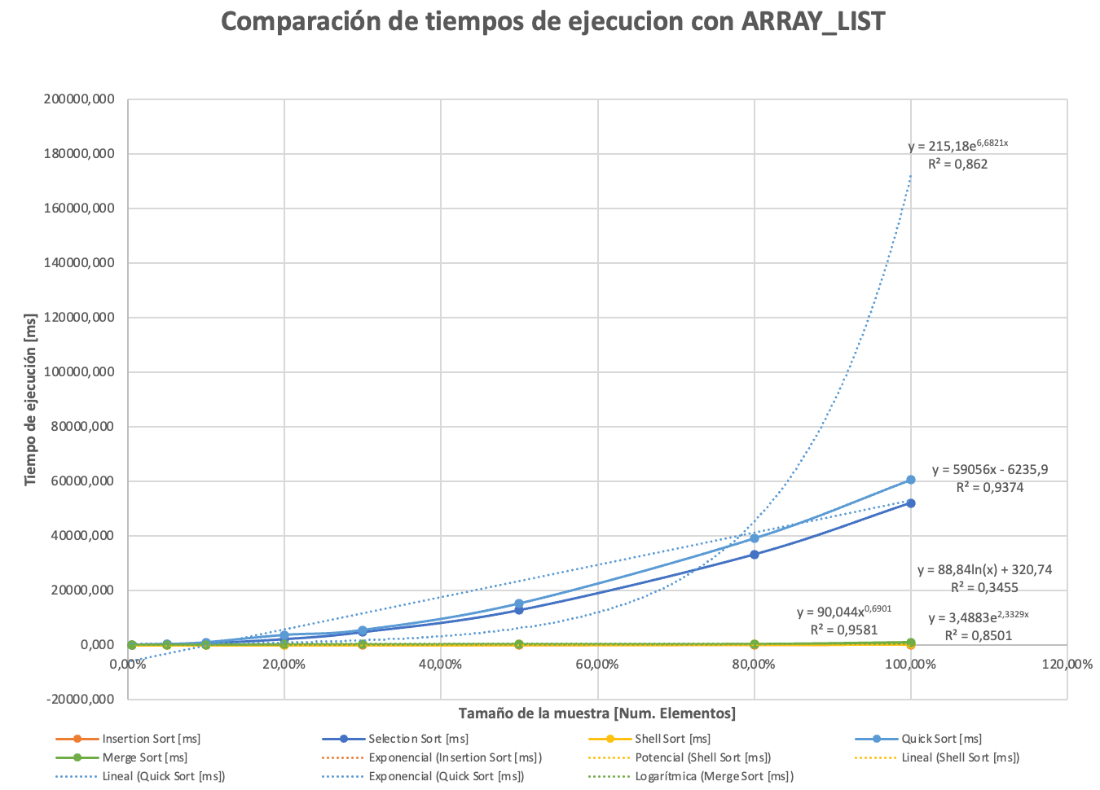
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

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

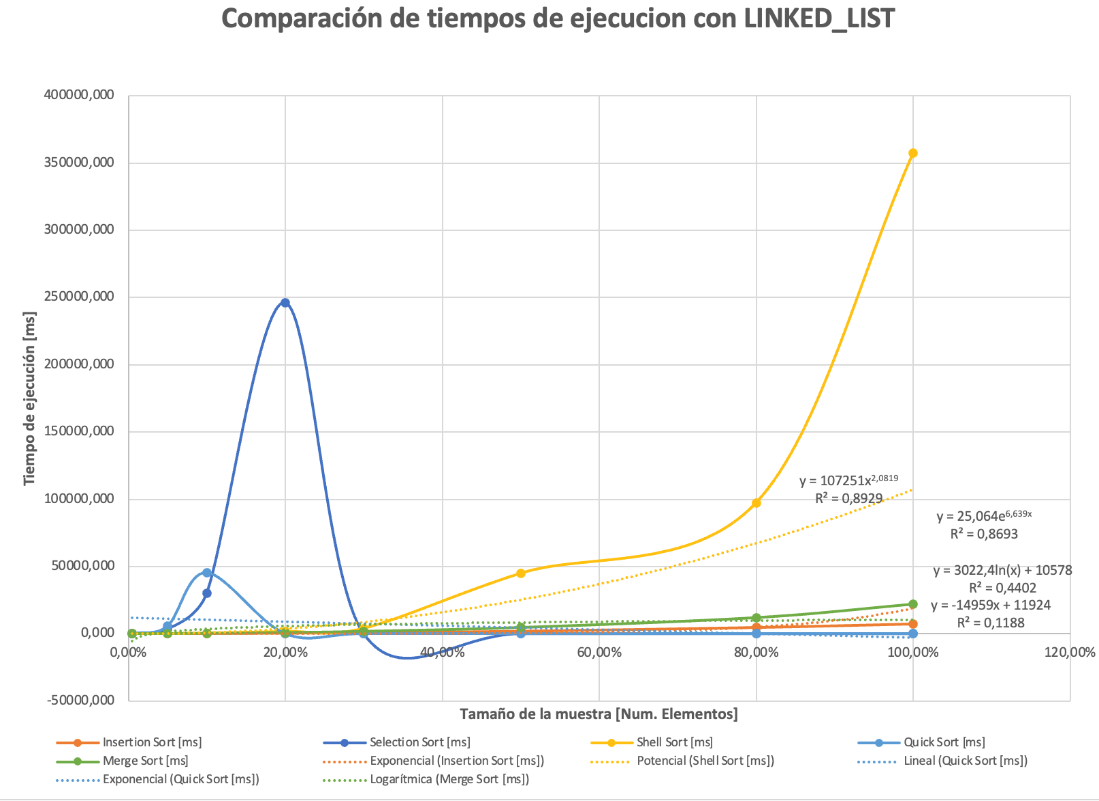
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

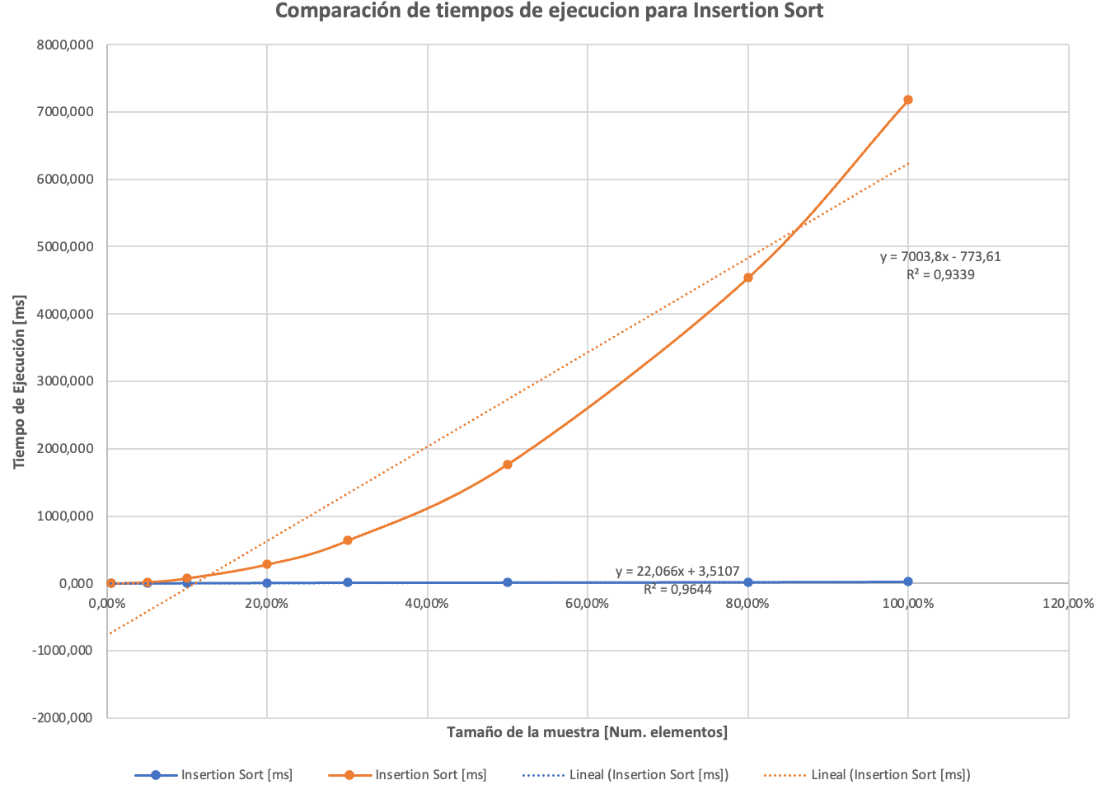
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 1.**
  + Comparación de rendimiento ARRAYLIST.



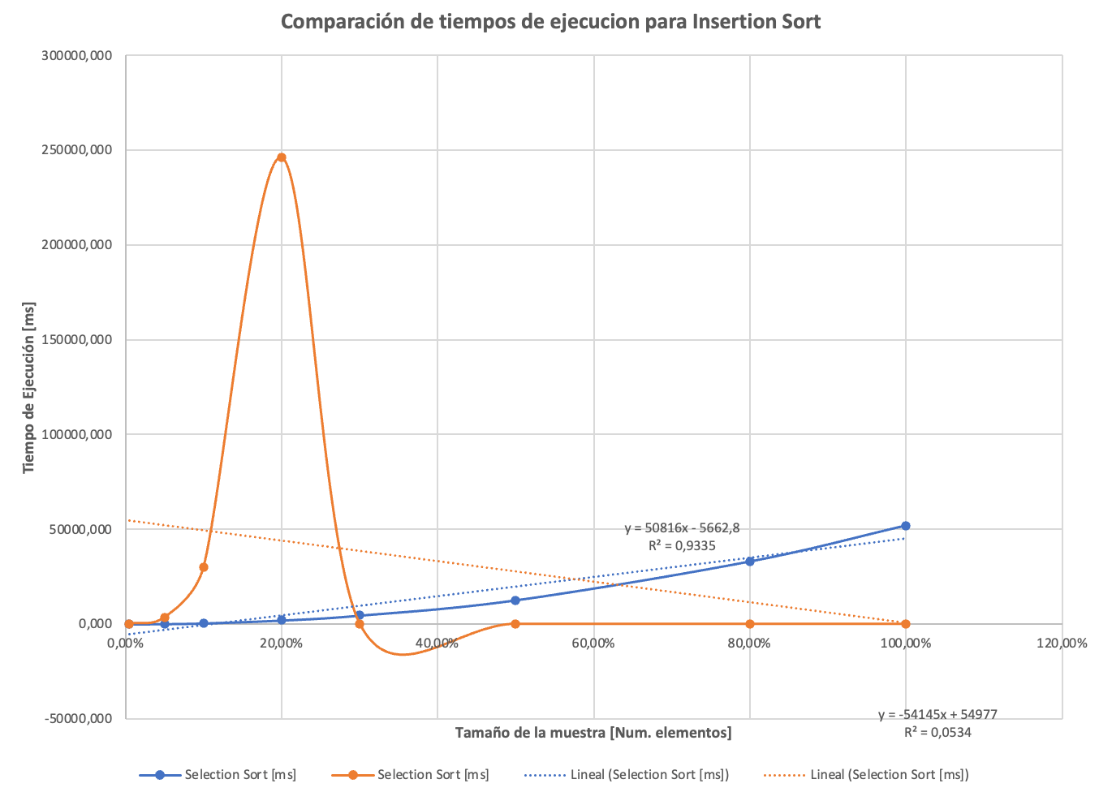
* + Comparación de rendimiento LINKED\_LIST.



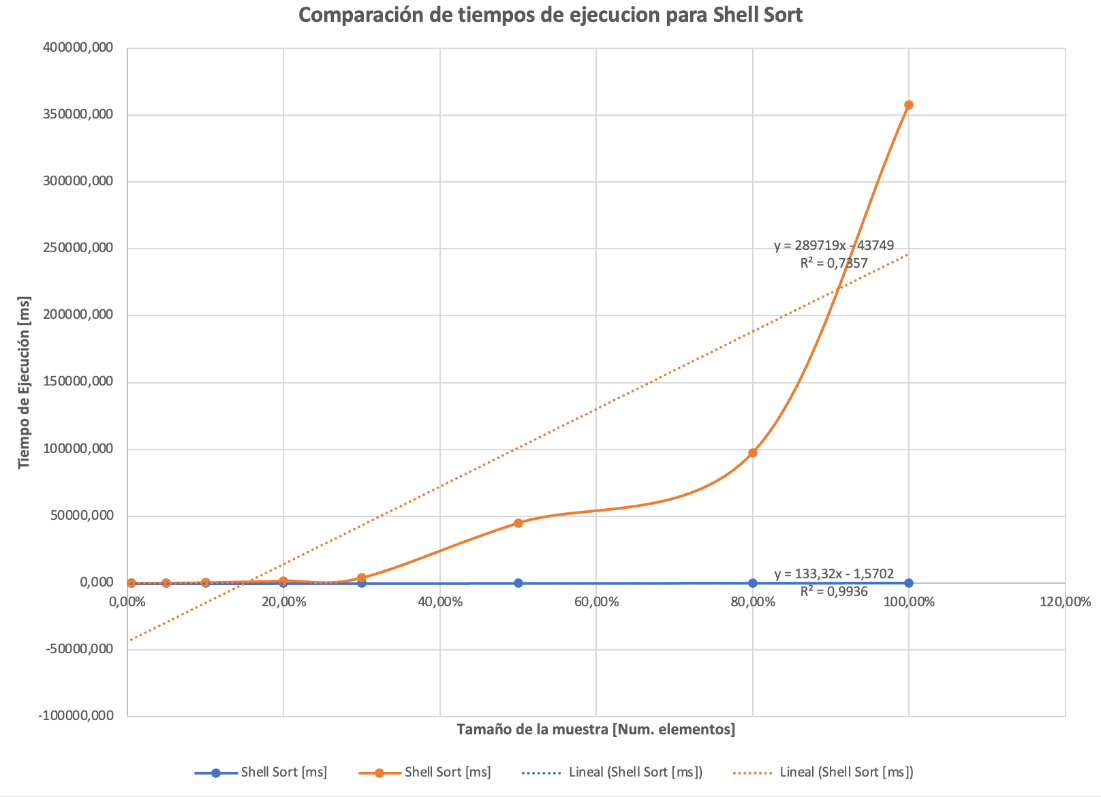
* + Comparación de rendimiento para Insertion Sort.



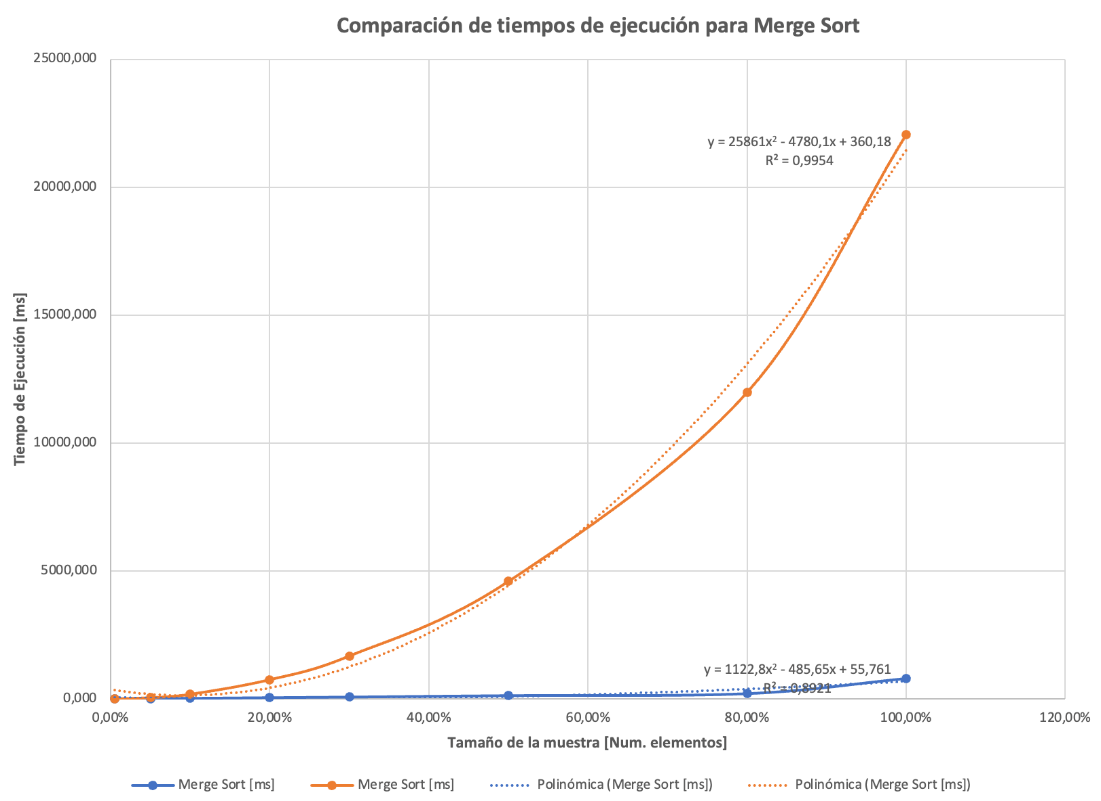
* + Comparación de rendimiento para Selection Sort.



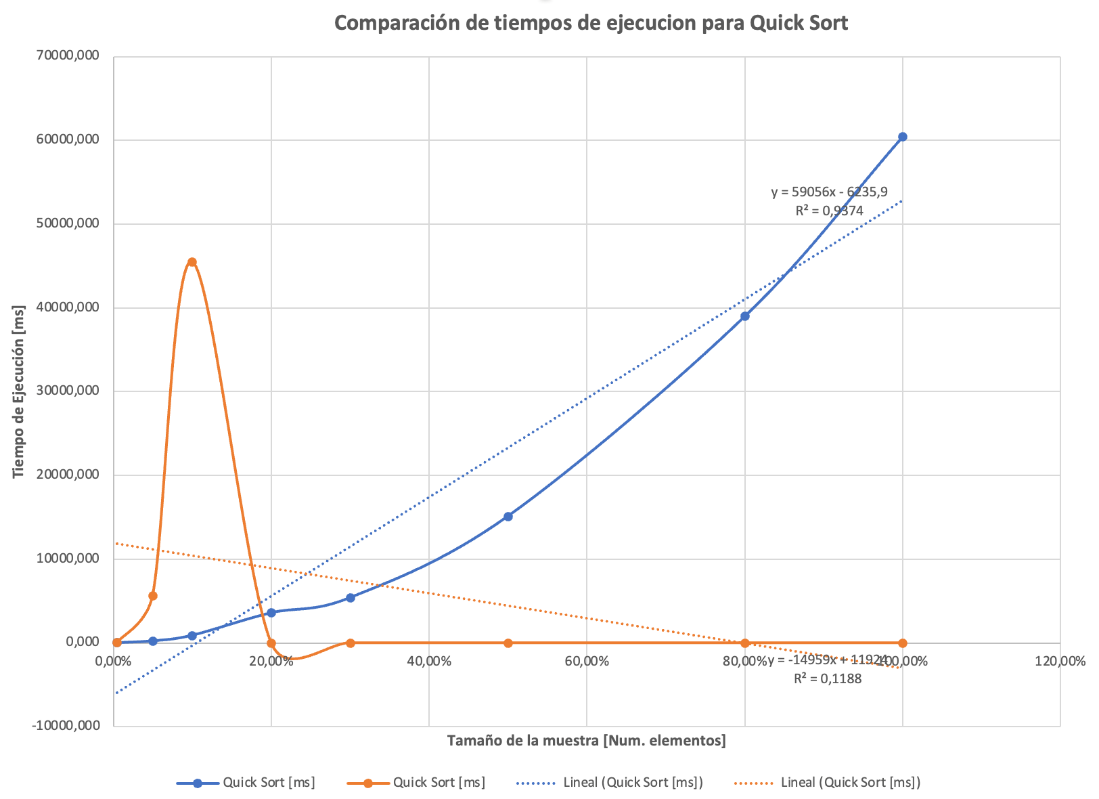
* + Comparación de rendimiento para Shell Sort.



* + Comparación de rendimiento para MergeSort.



* + Comparación de rendimiento para QuickSort.



# **Maquina 2**

## **Resultados**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (ARRAY\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% | 228 | 1.195 | 7.894 | 1.957 | 28.911 | 3.858 |
| 5.00% | 1148 | 3.283 | 162.628 | 6.412 | 470.19 | 27.326 |
| 10.00% | 2298 | 4.818 | 646.088 | 18.610 | 1780.825 | 53.713 |
| 20.00% | 4598 | 6.396 | 2576.003 | 38.119 | 7821.907 | 96.170 |
| 30.00% | 6898 | 13.707 | 6035.858 | 54.017 | 21183.656 | 164.813 |
| 50.00% | 11489 | 22.843 | 18606.143 | 89.928 | 48964.329 | 264.004 |
| 80.00% | 18397 | 32.647 | 52931.455 | 157.809 | 95342.712 | 425.635 |
| 100.00% | 22998 | 38.598 | 87666.544 | 203.692 | 196843.214 | 571.423 |

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

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (LINKED\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% | 228 | 2.653 | 58.765 | 5.679 | 198.532 | 11.471 |
| 5.00% | 1148 | 37.287 | 5452.907 | 119.582 | 16406.965 | 101.458 |
| 10.00% | 2298 | 105.892 | 42975.377 | 501.261 | 135891.728 | 455.923 |
| 20.00% | 4598 | 407.148 | 362212.771 | 2785.805 | - | 1791.822 |
| 30.00% | 6898 | 870.538 | 1272025.975 | 5712.181 | - | 4030.321 |
| 50.00% | 11489 | 2445.027 | - | 114460.005 | - | 9574.488 |
| 80.00% | 18397 | 6299.062 | - | 324235.278 | - | 28249.090 |
| 100.00% | 22998 | 21891.43 | - | 540810.96 | - | 48428.539 |

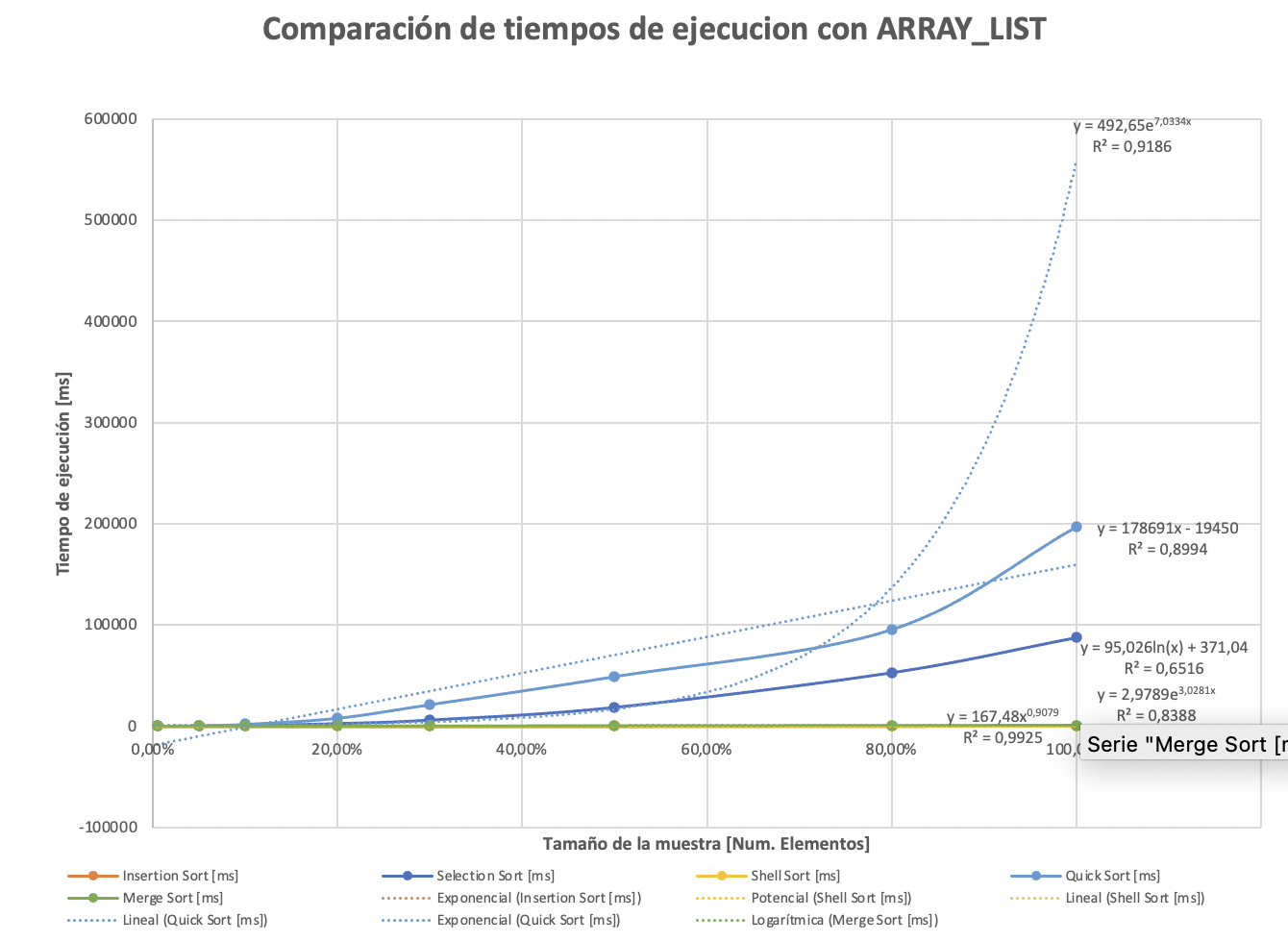
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

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

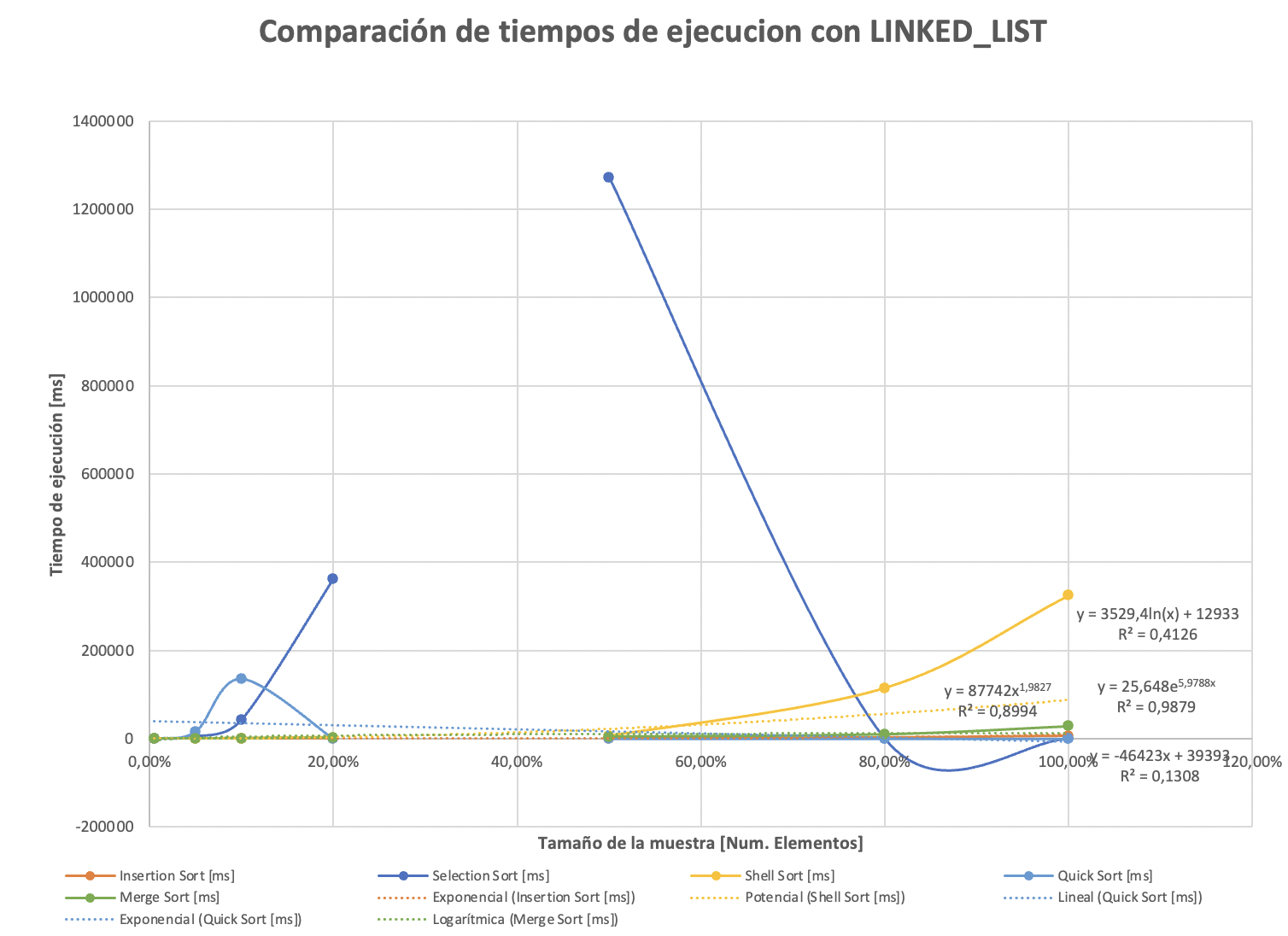
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

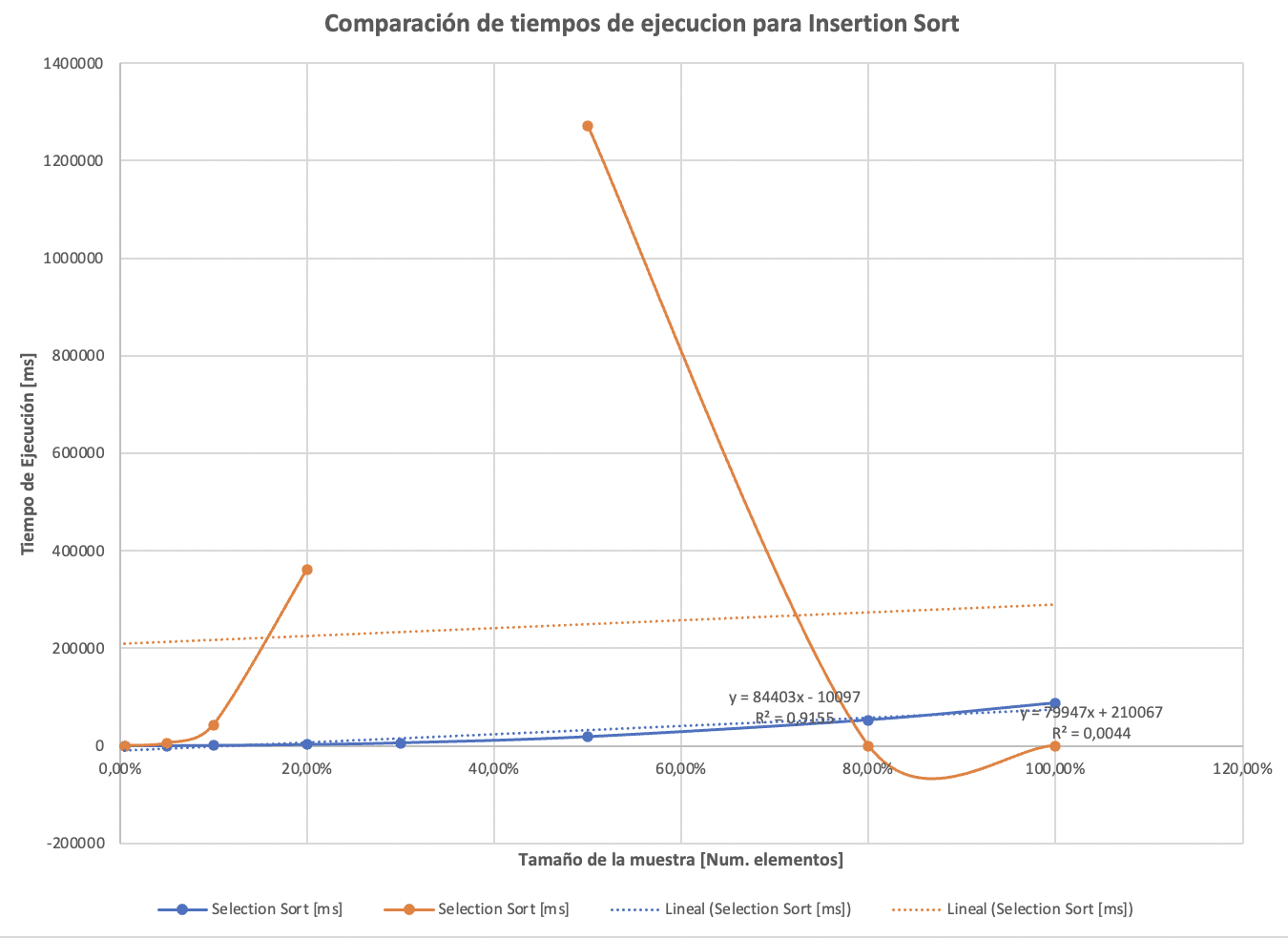
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 2.**
  + Comparación de rendimiento ARRAYLIST.



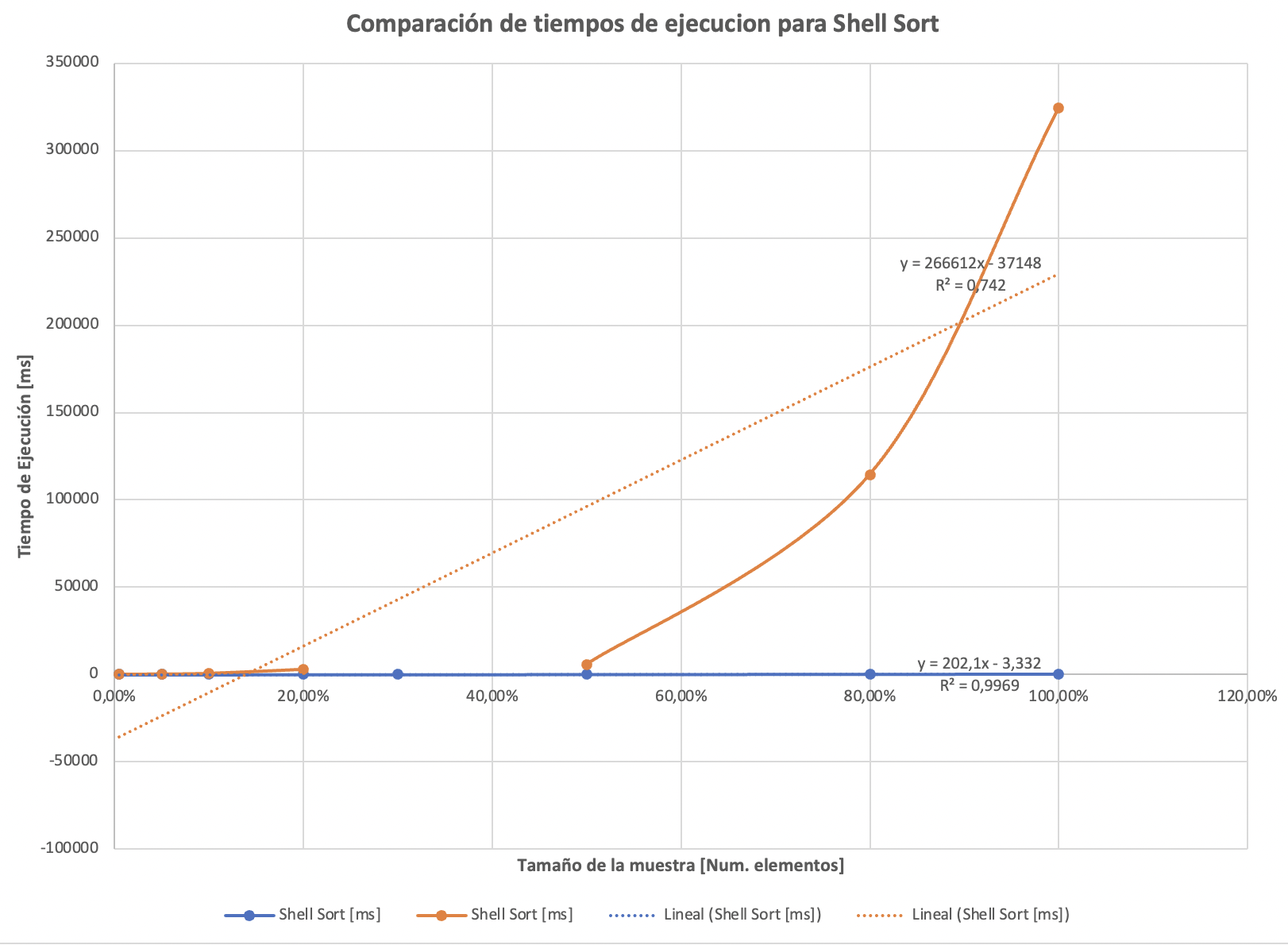
* + Comparación de rendimiento LINKED\_LIST.



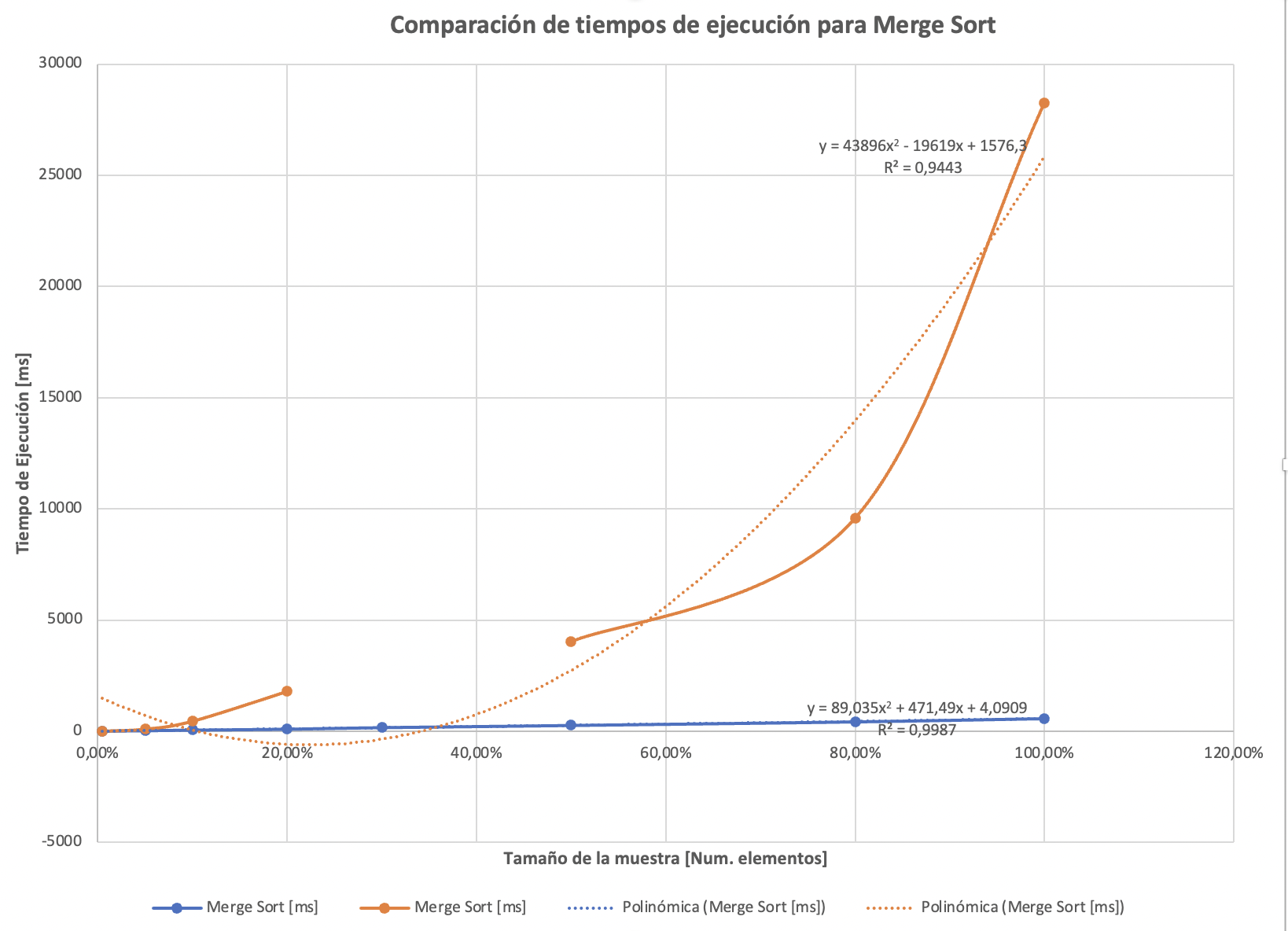
* + Comparación de rendimiento para Insertion Sort.
  + Comparación de rendimiento para Selection Sort.



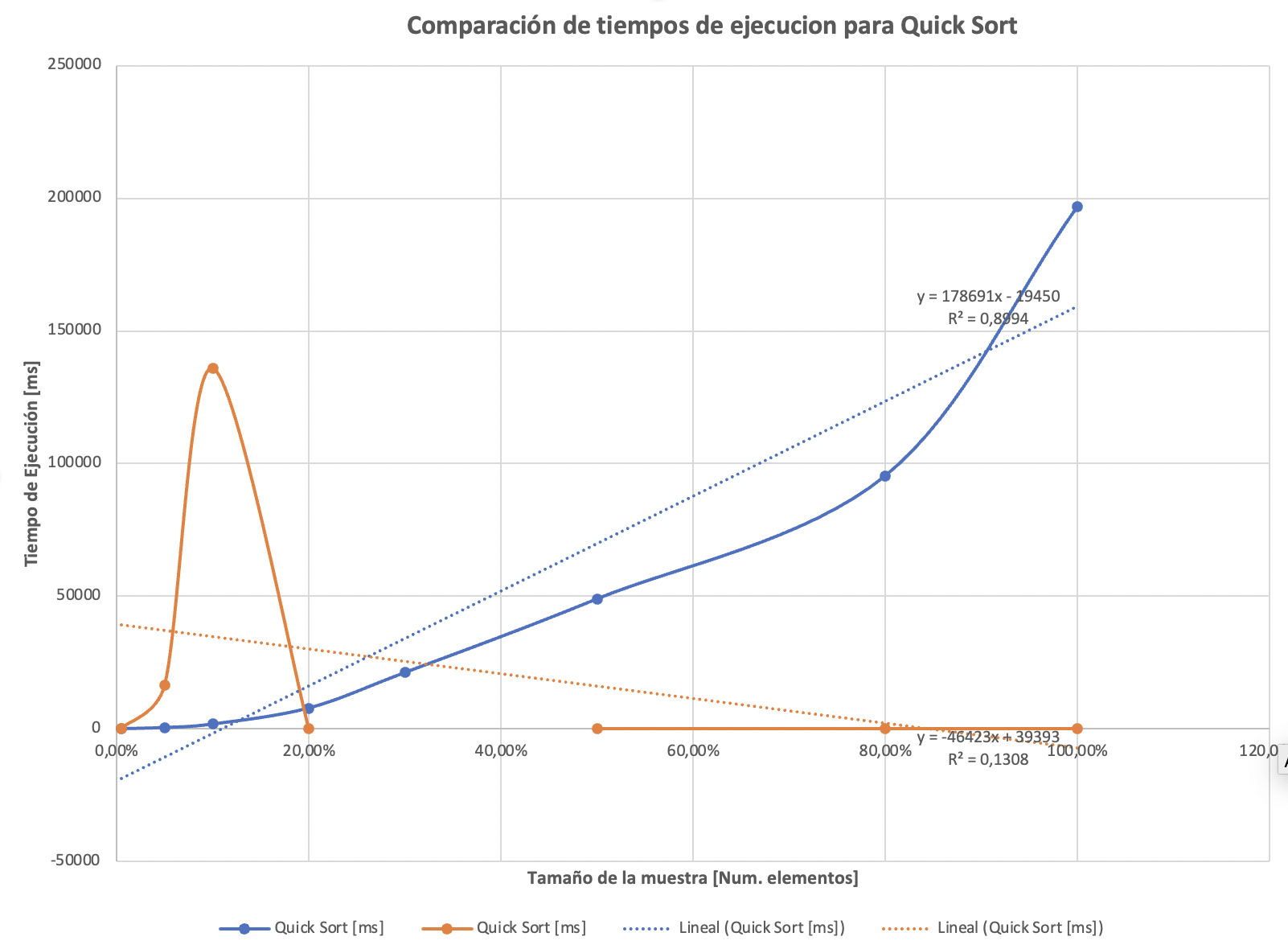
* + Comparación de rendimiento para Shell Sort.



* + Comparación de rendimiento para MergeSort.



* + Comparación de rendimiento para QuickSort.



# **Maquina 3**

## **Resultados**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (ARRAY\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% | 228 | 1.80 | 18.46 | 1.49 | 6.77 | 11.53 |
| 5.00% | 1148 | 2.50 | 393.91 | 13.86 | 47.78 | 57.90 |
| 10.00% | 2298 | 9.21 | 1667.47 | 43.32 | 94.55 | 111.05 |
| 20.00% | 4598 | 19.37 | 6152.27 | 210.12 | 230.54 | 230.27 |
| 30.00% | 6898 | 27.94 | 15560.85 | 131.85 | 333.48 | 332.60 |
| 50.00% | 11498 | 35.76 | 38249.30 | 264.69 | 542.18 | 491.62 |
| 80.00% | 18397 | 67.65 | 102358.30 | 536.19 | 674.25 | 826.85 |
| 100.00% | 22998 | 84.34 | 155968.08 | 403.31 | 1034.88 | 879.96 |

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

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (LINKED\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% | 228 | 6.52 | 178.10 | 7.65 | 254.60 | 17.65 |
| 5.00% | 1148 | 78.29 | 13389.08 | 286.44 | 1600.45 | 213.50 |
| 10.00% | 2298 | 268.64 | 109341.88 | 1271.88 | 6541.95 | 803.22 |
| 20.00% | 4598 | 1260847.34 | 861531.67 | 40809.87 | 34520.80 | 3471.91 |
| 30.00% | 6898 | 2834.25 | - | 115337.91 | 73312.46 | 8095.51 |
| 50.00% | 11498 | 9314.73 | - | 490725.65 | 200014.89 | 20232.23 |
| 80.00% | 18397 | 20050.82 | - | 941614.90 | 560489.60 | 44495.73 |
| 100.00% | 22998 | - | - | - | 902421.33 | 59672.28 |

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 |  |
| Quick sort | X |  |

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



Gráfico, Gráfico de líneas

Descripción generada automáticamenteGráfico, Gráfico de líneas

Descripción generada automáticamenteGráfico, Gráfico de líneas

Descripción generada automáticamenteGráfico, Gráfico de líneas

Descripción generada automáticamenteGráfico, Gráfico de líneas

Descripción generada automáticamenteGráfico, Gráfico de líneas

Descripción generada automáticamente

## **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 del algoritmo Merge sort, es acorde a el enunciado teóricamente ya que tiene complejidad Temporal de O (n log (n)), es estable y su complejidad espacial es de O(n). El algoritmo Quick sort también es acorde ya que tiene complejidad Temporal, en el peor caso de O (n ^ 2). Su Complejidad Temporal en el Caso promedio es de O (n log(n)), no es estable y su complejidad espacial es de O (log (n)).

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

Existen diferencias entre los resultados al ejecutar las pruebas ya que tienen tiempos de ejecución diferentes.

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

Estas diferencias se deben a que tienen un procesador y sistema operativo distinto y, por ende, unas máquinas toman más tiempo en recorrer ciertos códigos que otras.

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

Si solo te toma en cuenta los tiempos de ejecución de los algoritmos, la mejor estructura de datos para utilizar en este caso es el ARRAY\_LIST, ya que organiza los datos de una forma que es más eficiente y rápida en este caso con los datos del reto dados.

1. Para el caso analizado de ordenamiento de los artistas, teniendo en cuenta los resultados de tiempo reportados por todos los algoritmos de ordenamiento (iterativos y recursivos), proponga un ranking de los algoritmos de ordenamiento (de mayor eficiencia a menor (en tiempos de ejecución) para ordenar la mayor cantidad de artistas.

Según nuestros resultados, el ranking sería el siguiente:

* Insertion sort
* Shell sort
* Merge sort
* Selection sort
* Quick sort