OBSERVACIONES DE LA PRACTICA

Juan Esteban Currea Vergara Cod 201922133

María José Cely Cod 202011803

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | 1,4 GHz Intel Core i5 de cuatro núcleos | 1,4 GHz Intel Core i5 de cuatro núcleos |
| Memoria RAM (GB) | 8 GB 2133 MHz LPDDR3 | 8 GB 2133 MHz LPDDR3 |
| Sistema Operativo | macOS Big Sur version 11.2.1 | macOS Big Sur version 11.2.1 |

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

# **Maquina 1**

## **Resultados**

|  |  |  |  |
| --- | --- | --- | --- |
| Tamaño de la muestra (ARRAY\_LIST) | Insertion Sort (ms) | Selection Sort (ms) | Shell Sort (ms) |
| 1000 | 0.66 | 0.73 | 0.05 |
| 2000 | 2.60 | 2.91 | 0.09 |
| 4000 | 10.71 | 11.75 | 0.20 |
| 8000 | 43.14 | 52.82 | 0.49 |
| 16000 | 184.48 | 241.55 | 1.05 |
| 32000 | Se demora mucho | Se demora mucho | 2.52 |
| 64000 | Se demora mucho | Se demora mucho | 6.90 |
| 128000 | Se demora mucho | Se demora mucho | 14.86 |
| 256000 | Se demora mucho | Se demora mucho | 38.33 |
| 512000 | No alcanzan los datos | No alcanzan los datos | No alcanzan los datos |

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) |
| 1000 | 46.36 | 43.33 | 2.39 |
| 2000 | Se demora mucho | Se demora mucho | 11.16 |
| 4000 | Se demora mucho | Se demora mucho | 50.47 |
| 8000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 16000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 32000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 64000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 128000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 256000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 512000 | No alcanzan los datos | No alcanzan los datos | No alcanzan los datos |

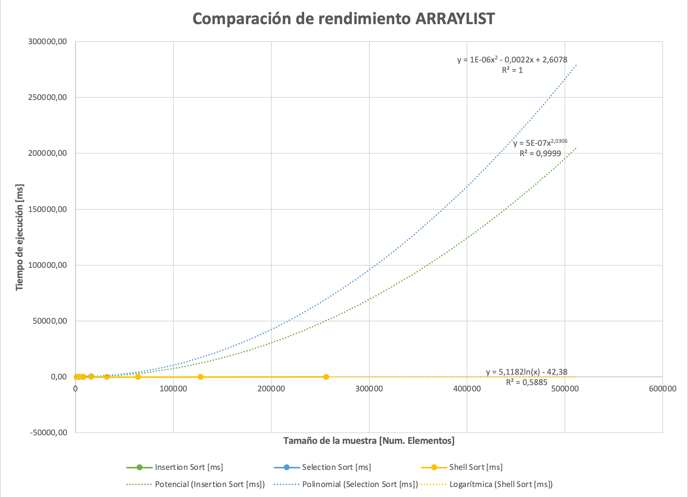
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| Insertion sort | 2 | 3 |
| Selection sort | 3 | 2 |
| Shell sort | 1 | 1 |

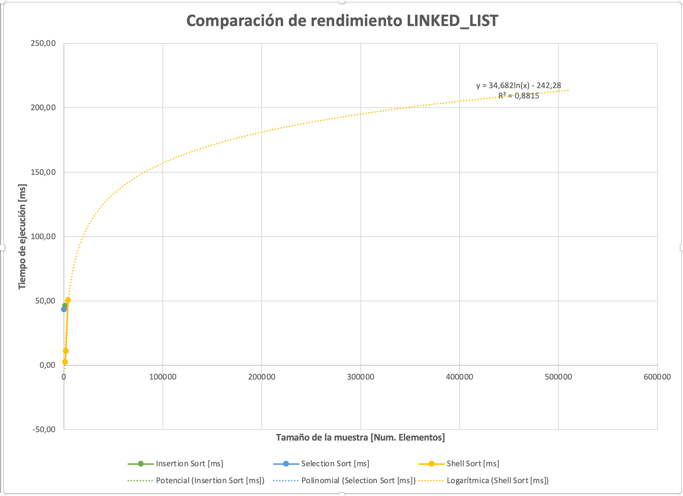
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

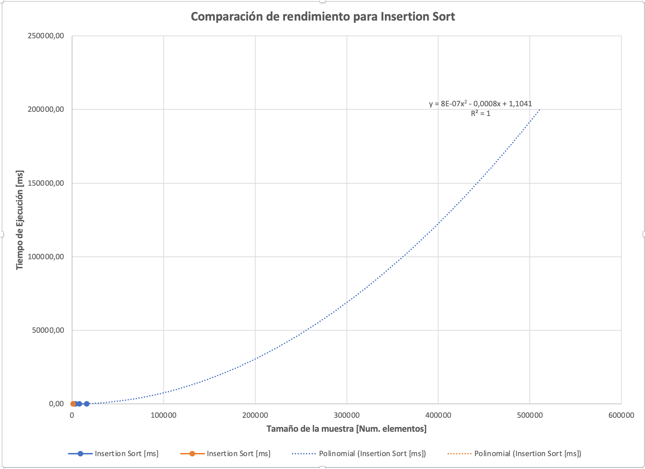
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 1.**
  + Comparación de rendimiento ARRAYLIST.



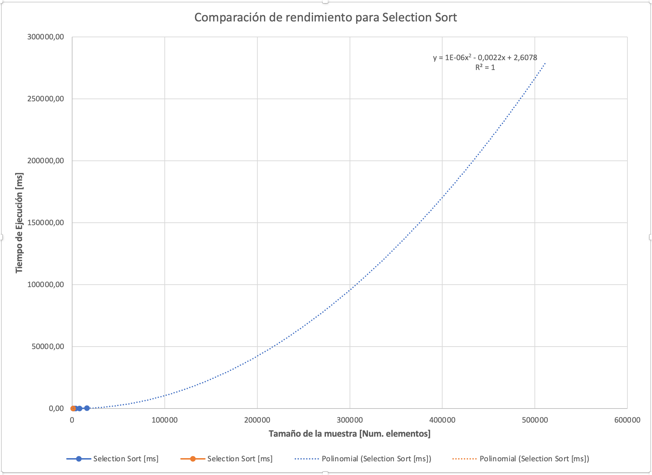
* + Comparación de rendimiento LINKED\_LIST.



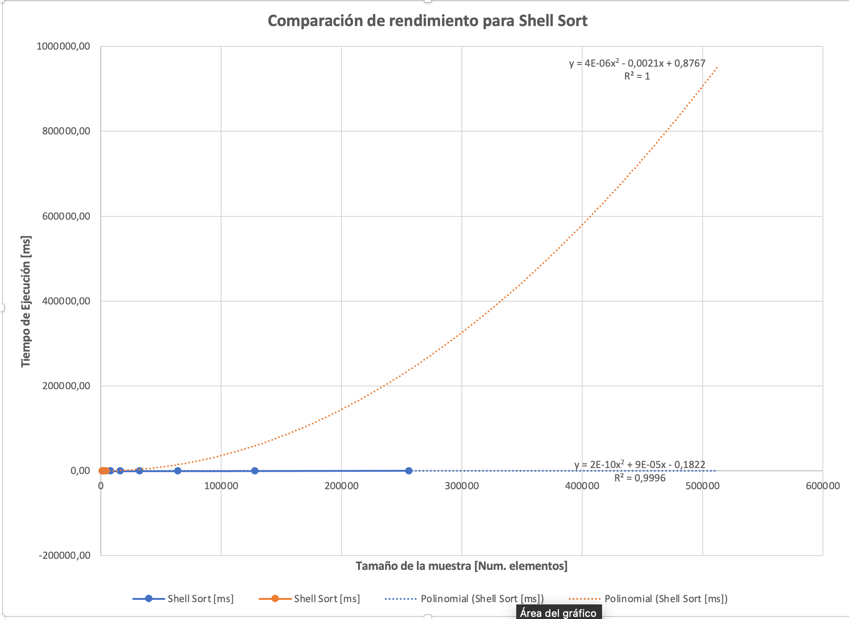
* + Comparación de rendimiento para Insertion Sort.



* + Comparación de rendimiento para Selection Sort.



* + Comparación de rendimiento para Shell Sort.



# **Maquina 2**

## **Resultados**

|  |  |  |  |
| --- | --- | --- | --- |
| Tamaño de la muestra (ARRAY\_LIST) | Insertion Sort (ms) | Selection Sort (ms) | Shell Sort (ms) |
| 1000 | 0.55 | 0.62 | 0.04 |
| 2000 | 2.26 | 2.51 | 0.09 |
| 4000 | 8.73 | 11.07 | 0.17 |
| 8000 | 40.42 | 48.04 | 0.43 |
| 16000 | 154.26 | 202.01 | 0.89 |
| 32000 | 654.87 | 928.24 | 2.16 |
| 64000 | Se demora mucho | Se demora mucho | 5.25 |
| 128000 | Se demora mucho | Se demora mucho | 11.52 |
| 256000 | Se demora mucho | Se demora mucho | 30.73 |
| 512000 | No alcanzan los datos | No alcanzan los datos | No alcanzan los datos |

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) |
| 1000 | 41.14 | 38.32 | 2.21 |
| 2000 | 340.22 | 306.94 | 38.32 |
| 4000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 8000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 16000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 32000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 64000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 128000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 256000 | Se demora mucho | Se demora mucho | Se demora mucho |
| 512000 | No alcanzan los datos | No alcanzan los datos | No alcanzan los datos |

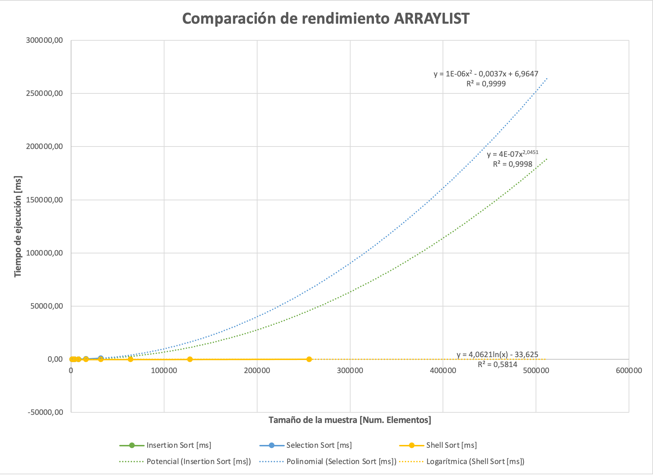
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| Insertion sort | 2 | 3 |
| Selection sort | 3 | 2 |
| Shell sort | 1 | 1 |

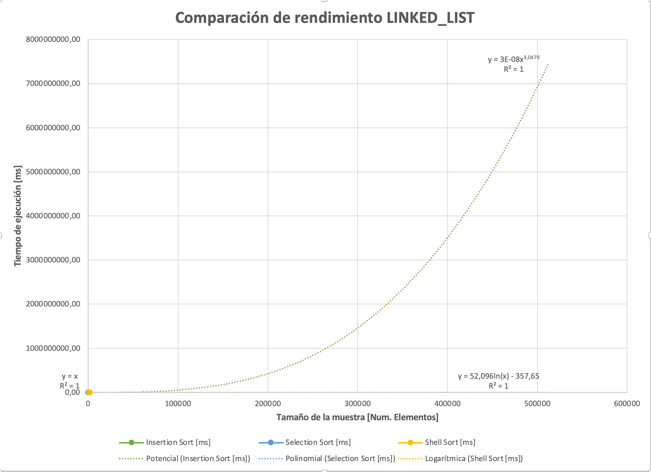
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

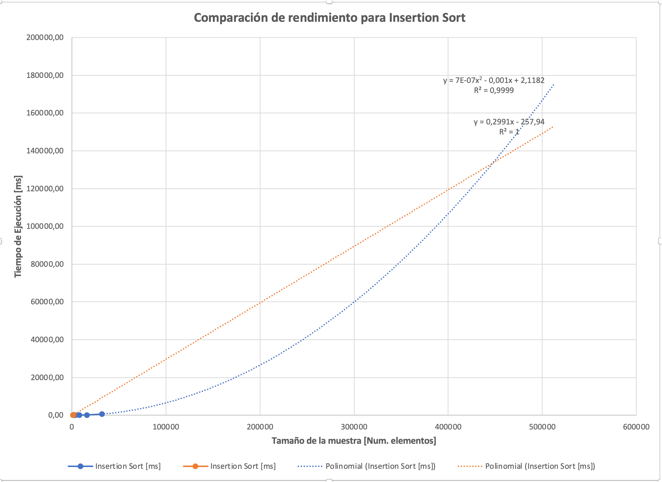
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 2.**
  + Comparación de rendimiento ARRAYLIST.



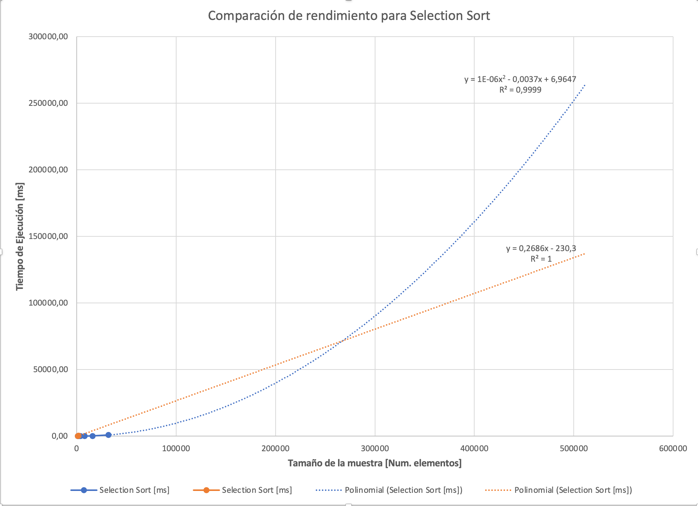
* + Comparación de rendimiento LINKED\_LIST.



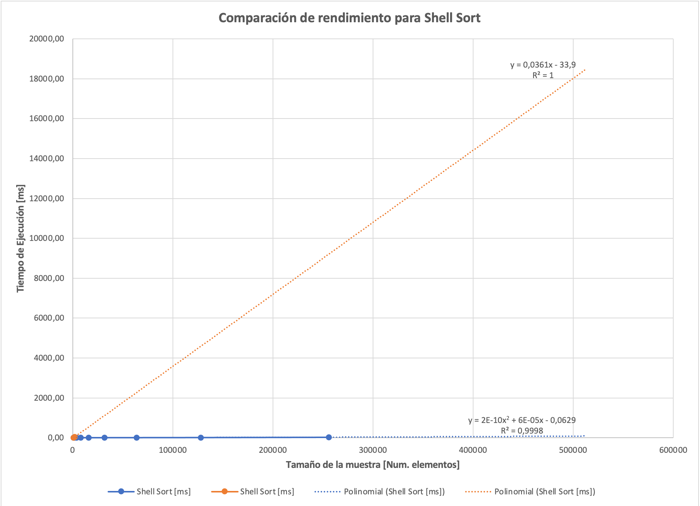
* + Comparación de rendimiento para Insertion Sort.



* + Comparación de rendimiento para Selection Sort.



* + Comparación de rendimiento para Shell Sort.



# **Preguntas de análisis**

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

Si, pues segun lo visto en clase el mejor algoritmo es el shellsort por su orden de complejidad en el peor caso que es cn3/2  frente a selectionsort y insetionsort que es ½ n2.

1. ¿Existe alguna diferencia entre los resultados obtenidos al ejecutar las pruebas en diferentes máquinas?
   * Tenemos el mismo computador así que no existen demasiadas, sin embargo, las hay.
2. De existir diferencias, ¿A qué creen ustedes que se deben dichas diferencias?
   * Creeria que a la cantidad de applicaciones abiertas, pues se cuenta con el mismo computador.
3. ¿Cuál Estructura de Datos es mejor utilizar si solo se tiene en cuenta los tiempos de ejecución de los algoritmos?
   * La mejor estructura es ARRAY\_LIST pues el tiempo con cualquiera de los 3 tipos de ordenamiento, es menor que con la LINKED\_LIST.