OBSERVACIONES DE LA PRACTICA

Andrés Beltrán Cod 202014143

Juan Montealegre Cod 202012723

Lina Maríg Gómez Cod 201923531

|  |  |  |  |
| --- | --- | --- | --- |
|  | Máquina 1 | Máquina 2 | Máquina 3 |
| Procesadores | Intel® CoreTM i7- 1065G7 CPU @ 1.30GHz, 1.50GHz | Intel® Core™ i7-8550U CPU @ 1.8GHZ, 2.00GHz | Intel® Core™ i5-8250U CPU @ 1.6GHZ, 1800 Mhz |
| Memoria RAM (GB) | 12GB | 16 GB | 8.0 GB |
| Sistema Operativo | Windows 10 Pro-64-bits | Windows 10 Home-64-bits | Windows 10 Pro 64-bits |

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 | 33765.625 | 30968.75 | 1609.375 |
| 2000 | 418593.75 | 387500.0 | 11703.125 |
| 4000 | Excede los 25 min | Excede los 25 min | 51406.25 |
| 8000 |  |  | 257625.0 |
| 16000 |  |  | Excede los 25 min |
| 32000 |  |  |  |
| 64000 |  |  |  |
| 128000 |  |  |  |
| 256000 |  |  |  |
| 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) |
| 1000 | 32093.75 | 31921.875 | 1734.375 |
| 2000 | 310734.375 | 260406.25 | 11468.75 |
| 4000 |  |  | 58515.625 |
| 8000 |  |  | 319640.625 |
| 16000 |  |  | Excede los 25 min |
| 32000 |  |  |  |
| 64000 |  |  |  |
| 128000 |  |  |  |
| 256000 |  |  |  |
| 512000 |  |  |  |

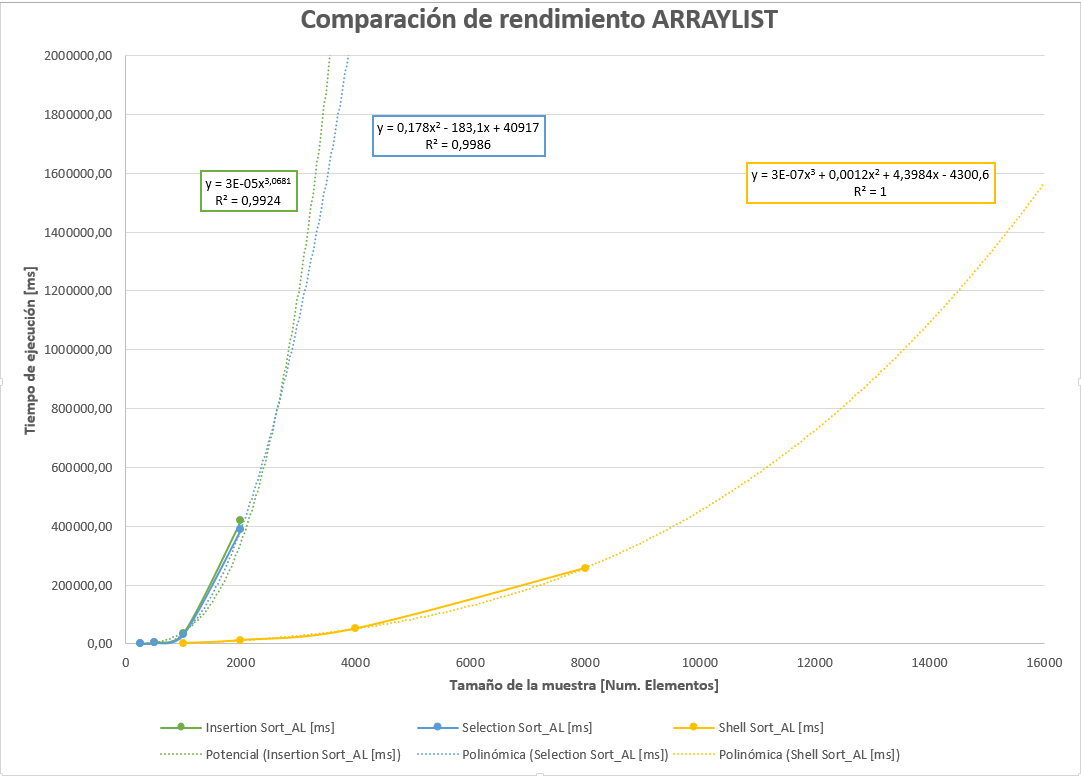
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

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

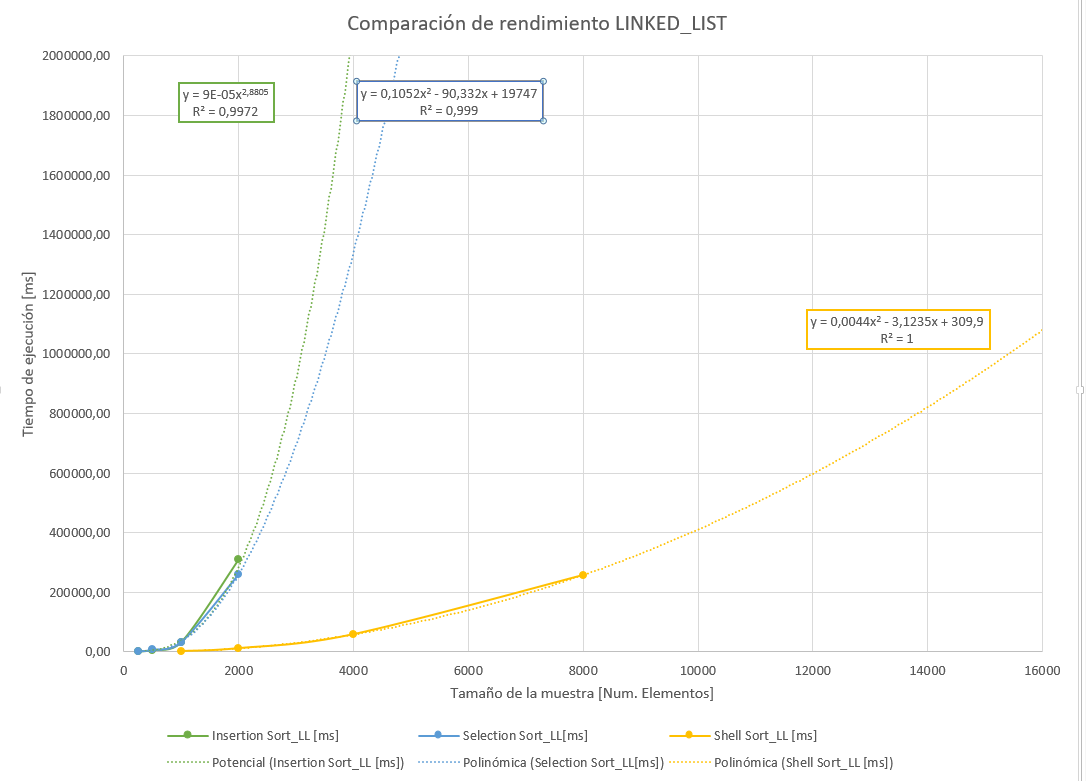
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

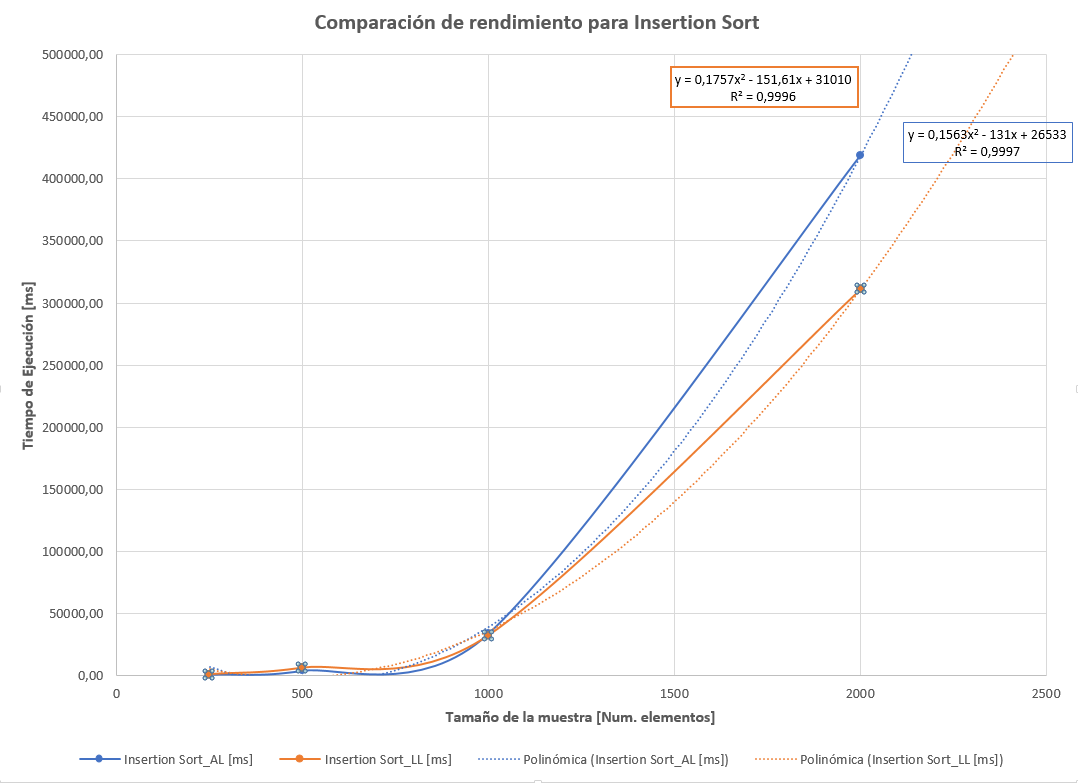
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 1.**
  + Comparación de rendimiento ARRAYLIST.



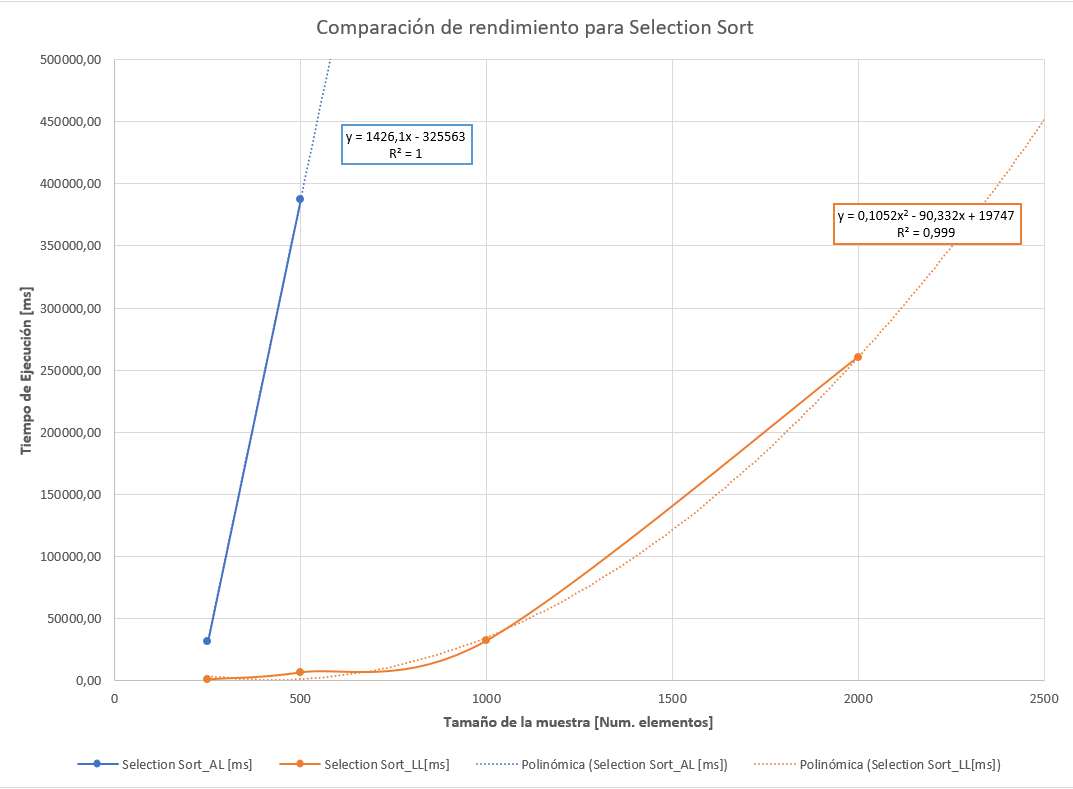
* + Comparación de rendimiento LINKED\_LIST.



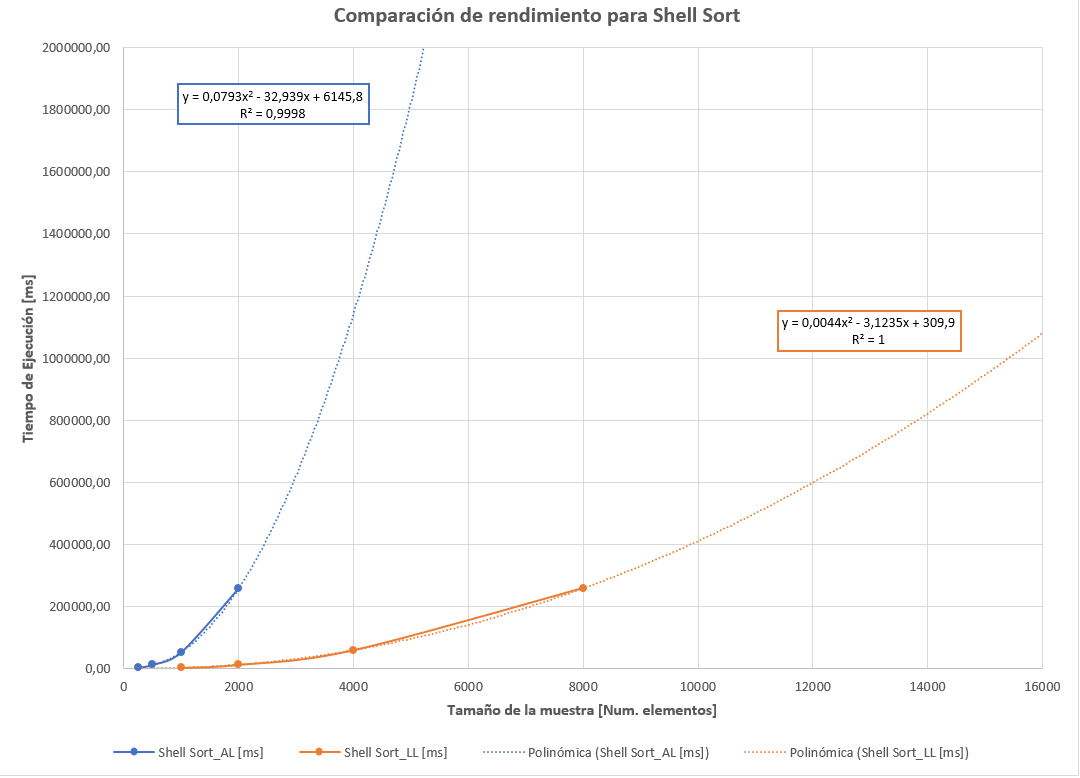
* + Comparación de rendimiento para Insertion Sort.



* + Comparación de rendimiento para Selection Sort.



* + Comparación de rendimiento para Shell Sort.



# **Maquina 2**

## **Resultados**

|  |  |  |  |
| --- | --- | --- | --- |
| Tamaño de la muestra (ARRAY\_LIST) | Insertion Sort (ms) | Selection Sort (ms) | Shell Sort (ms) |
| 1000 | 62031.25 | 67890.625 | 3546.8 |
| 2000 | 619828.125 | 594656.25 | 12390.6 |
| 4000 | Excede los 45 minutos | Excedelos 45 minutos | 68812.5 |
| 8000 |  |  | 543312.5 |
| 16000 |  |  | 2017437.5 |
| 32000 |  |  |  |
| 64000 |  |  |  |
| 128000 |  |  |  |
| 256000 |  |  |  |
| 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) |
| 1000 | 99593.75 | 69140.62 | 2828.15 |
| 2000 | 636390.625 | 567015.625 | 16718.75 |
| 4000 |  |  | 82656 |
| 8000 |  |  | 413859.3 |
| 16000 |  |  | 2008281.2 |
| 32000 |  |  |  |
| 64000 |  |  |  |
| 128000 |  |  |  |
| 256000 |  |  |  |
| 512000 |  |  |  |

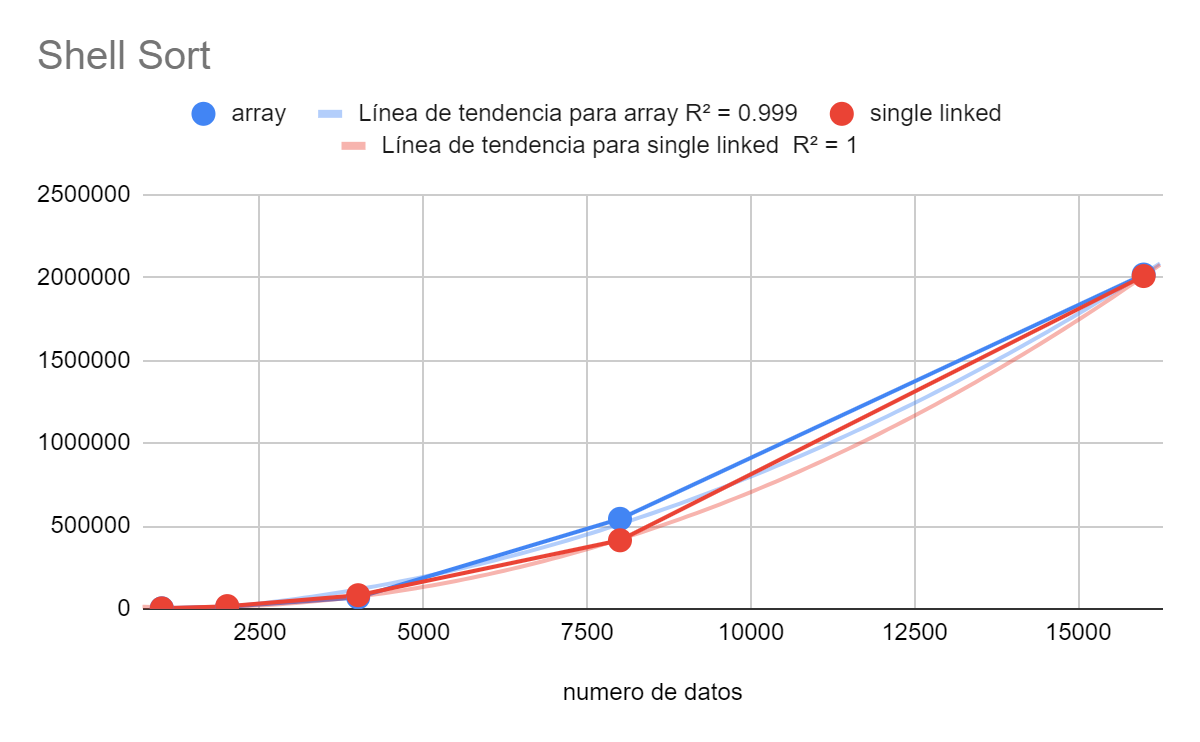
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

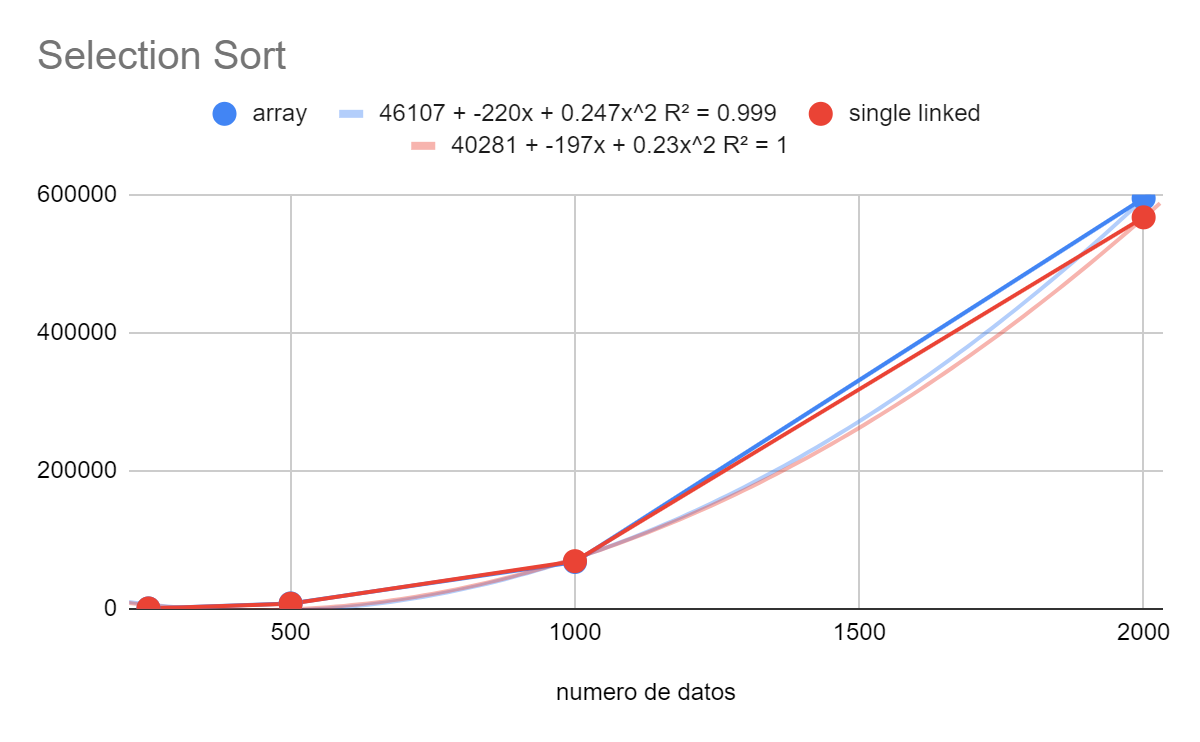
|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| Insertion sort | x |  |
| Selection sort | x |  |
| Shell sort | x |  |

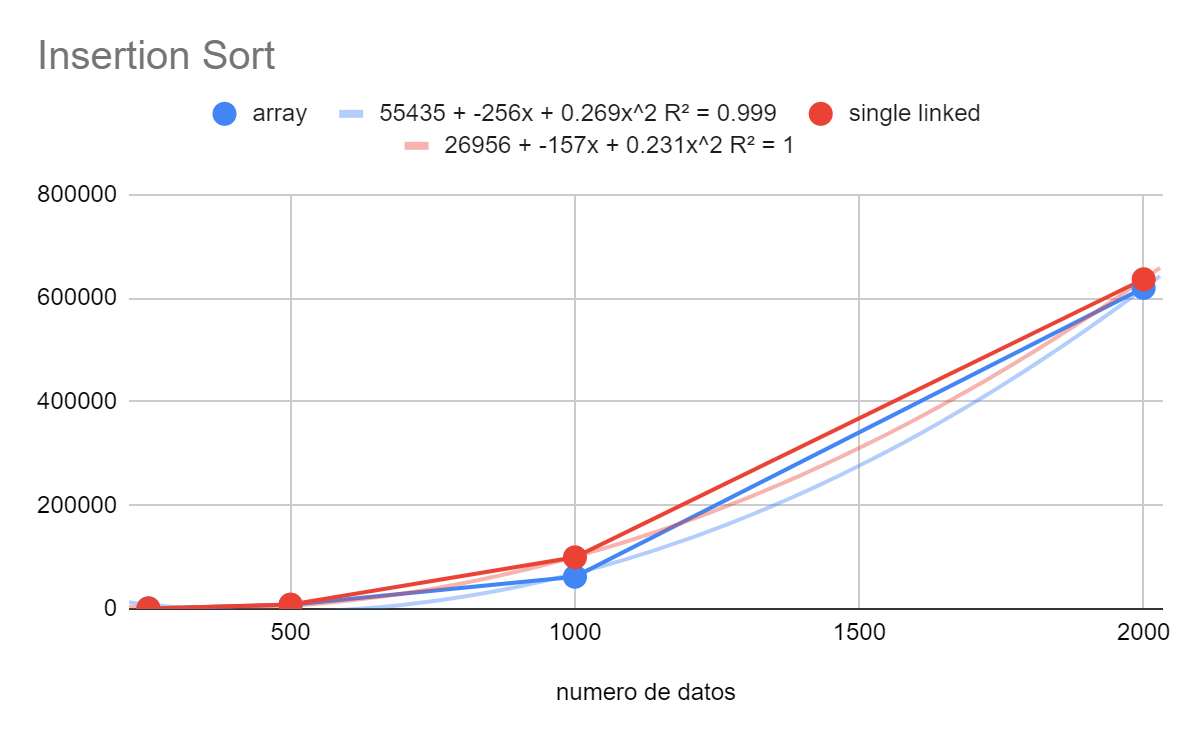
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

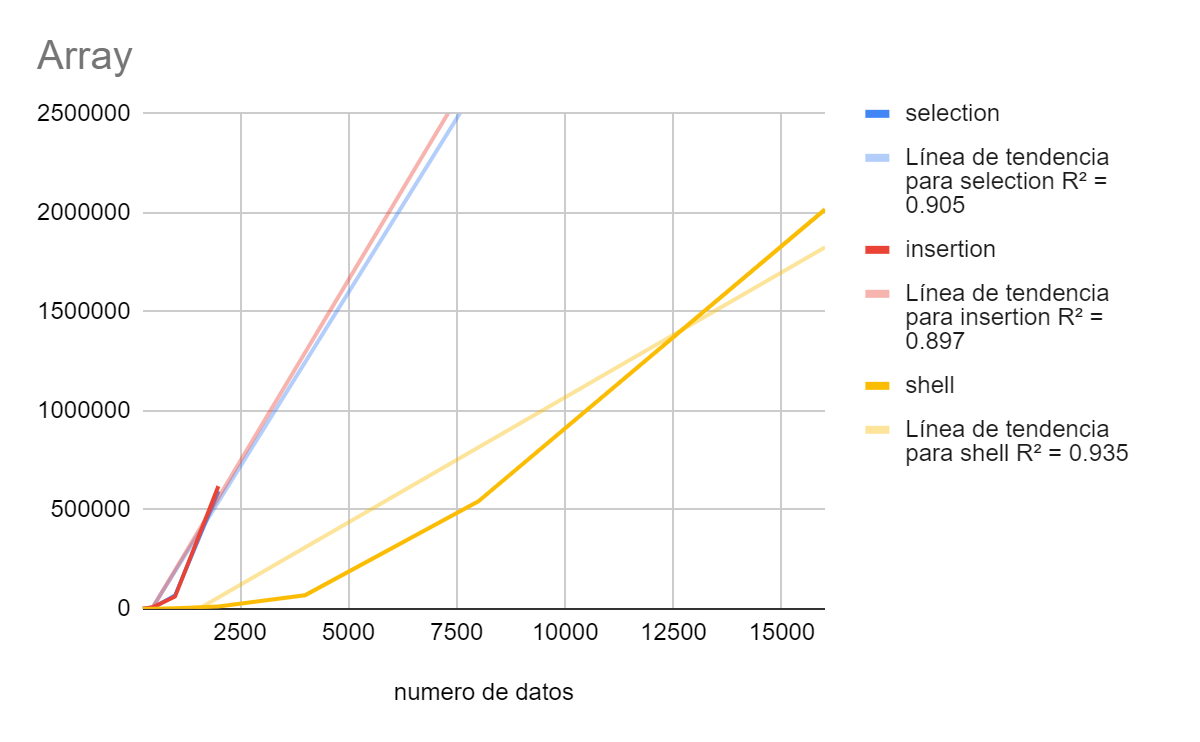
## **Graficas**

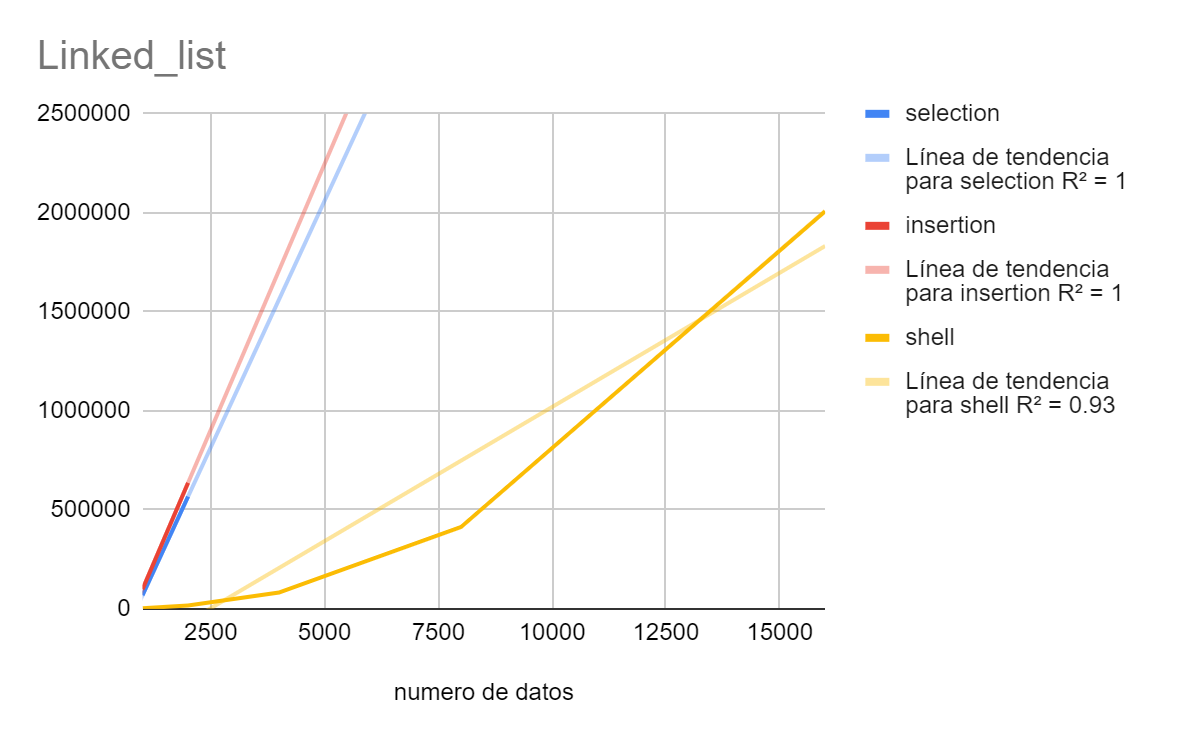
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 2.**
  + Comparación de rendimiento ARRAYLIST.
  + Comparación de rendimiento LINKED\_LIST.
  + Comparación de rendimiento para Insertion Sort.
  + Comparación de rendimiento para Selection Sort.
  + Comparación de rendimiento para Shell Sort.











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

Si los comportamientos presentados por los algoritmos son acordesc on la teoria

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

SI principal mente los tiempos de respuestas cambiaron, pero en general las gráficas guardan una correlación

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

Son muchos factores los cuales puede depender estas diferencias, el principal los componentes del computador, la arquitectura del procesador, la cantidad de aplicaciones que se ejecutan en segundo plano, pueden influir en los resultados

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

Se es evidente que el array list es la mas eficiente, y el metodo de ordenamierto el Shell Sort teniendo en cuenta solo los tiempos.

# **Maquina 3**

## **Resultados**

|  |  |  |  |
| --- | --- | --- | --- |
| Tamaño de la muestra (ARRAY\_LIST) | Insertion Sort (ms) | Selection Sort (ms) | Shell Sort (ms) |
| 1000 | 49510,4167 | 44385,4167 | 2270,83 |
| 2000 | 389968,7500 | 371796,875 | 10432,29 |
| 4000 | Excede el tiempo de 15min | Excede el tiempo de 30 min | 50088,5417 |
| 8000 | - | - | 241296,875 |
| 16000 | - | - | 1333531,25 |
| 32000 | - | - | Excede el tiempo de 15 min |
| 64000 | - | - | - |
| 128000 | - | - | - |
| 256000 | - | - | - |
| 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) |
| 1000 | 52125 | 50526,04 | 2864,375 |
| 2000 | 430906,25 | 430687,5 | 13702,95 |
| 4000 | Tiempo mayor a 15 min | Tiempo mayor a 30 min | 56718,75 |
| 8000 | - | - | 289171,875 |
| 16000 | - | - | 1294656,25 |
| 32000 | - | - | Excede el tiempo de 30 min |
| 64000 | - | - | - |
| 128000 | - | - | - |
| 256000 | - | - | - |
| 512000 | - | - | - |

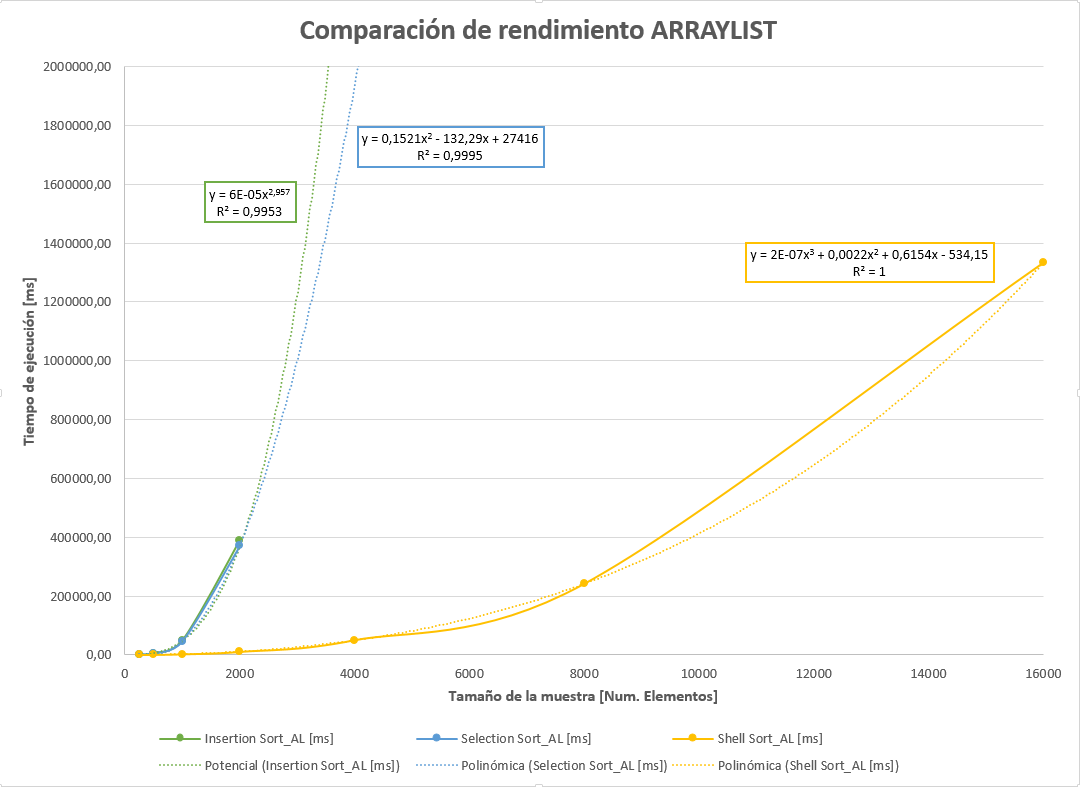
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

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

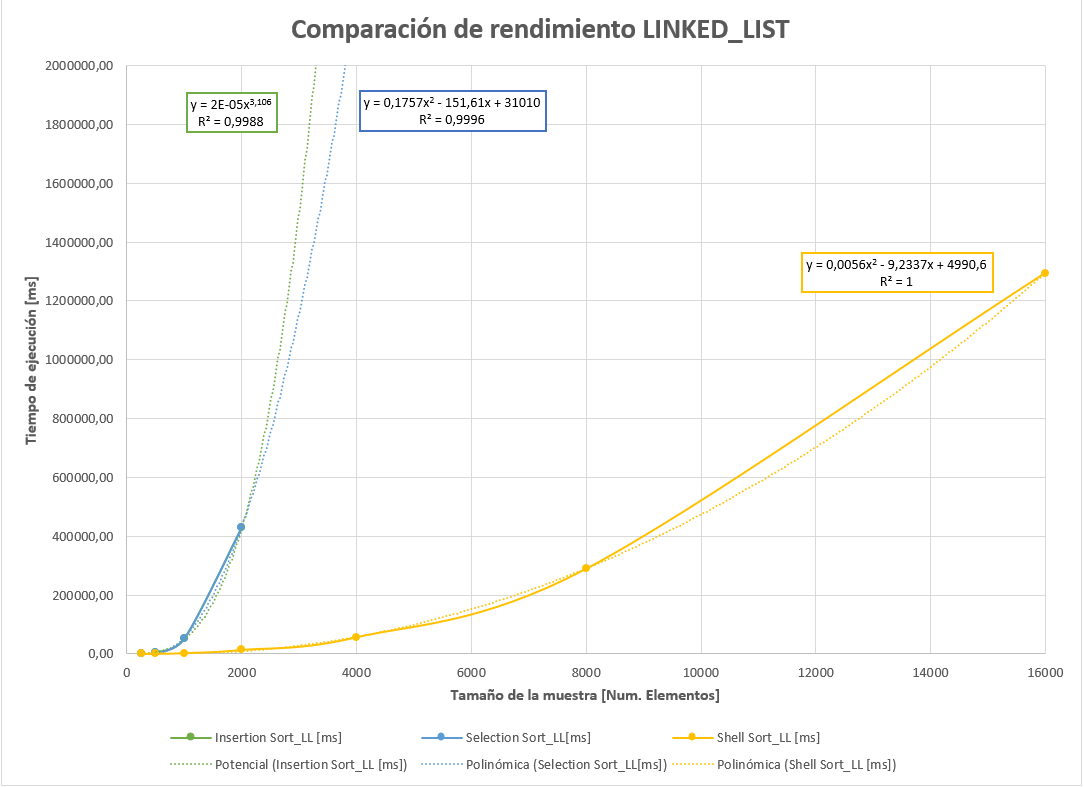
Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

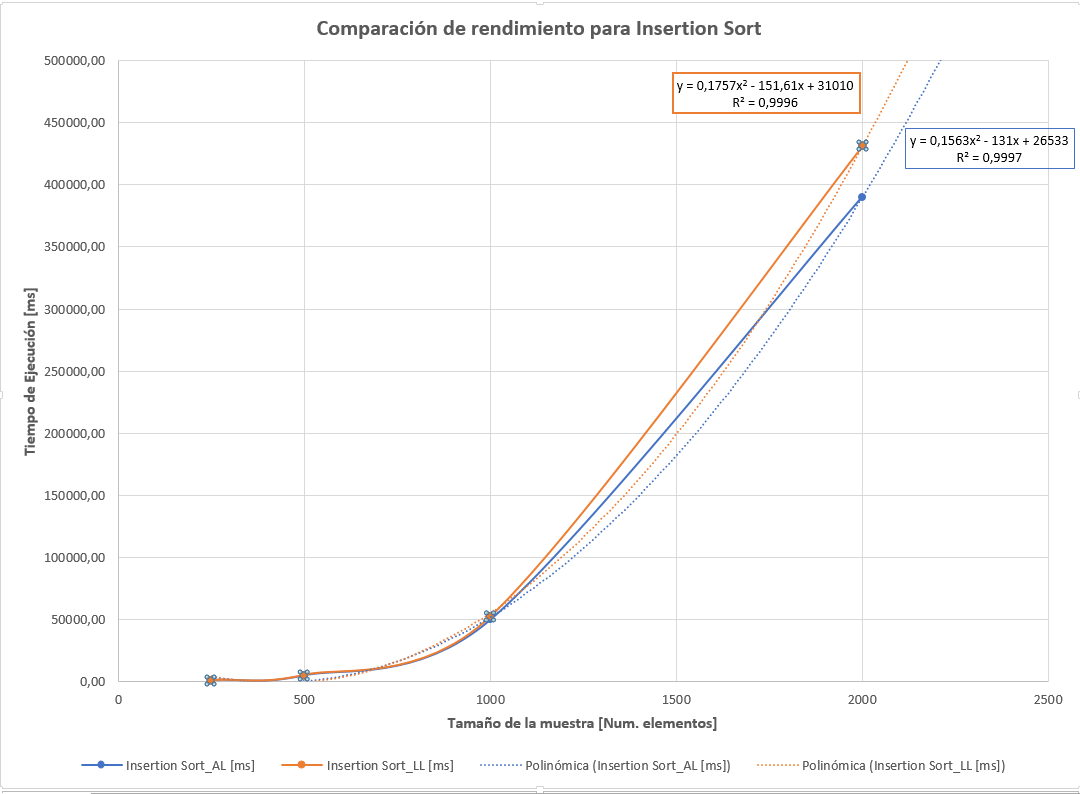
* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 2.**
  + Comparación de rendimiento ARRAYLIST.



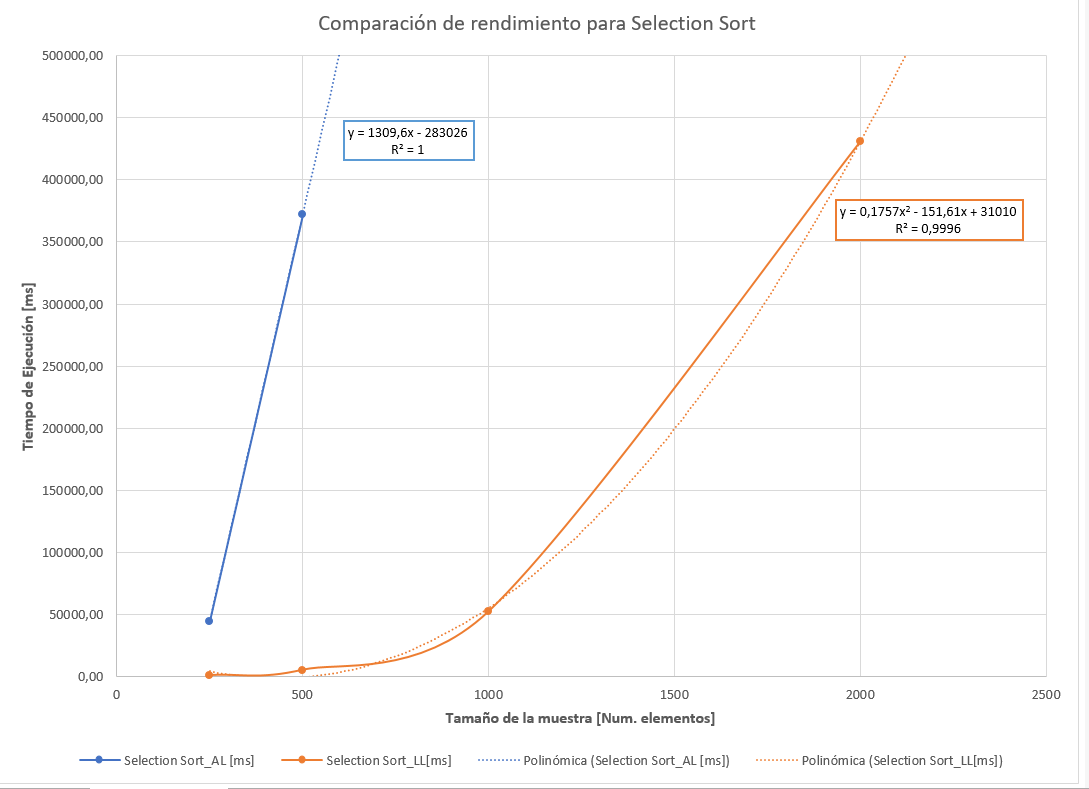
* + Comparación de rendimiento LINKED\_LIST.



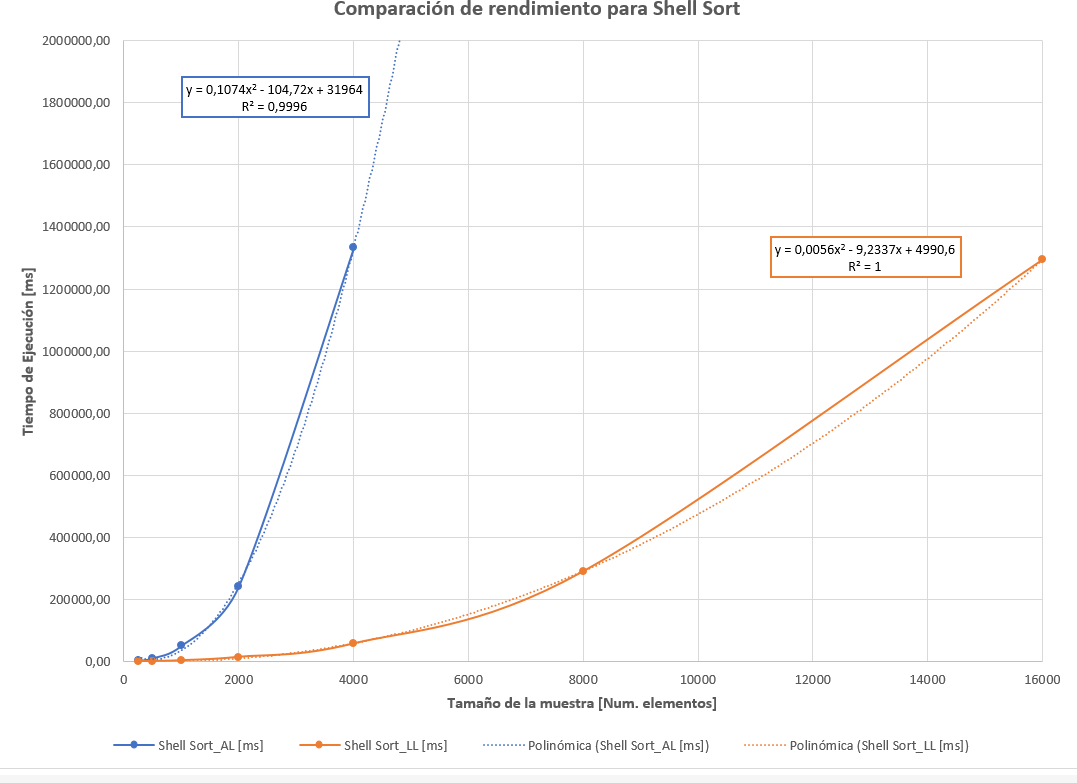
* + Comparación de rendimiento para Insertion Sort.



* + Comparación de rendimiento para Selection Sort.



* + Comparación de rendimiento para Shell Sort.



# **Preguntas de análisis**

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

Sí pues cada uno se terminó coportando de acuerdo a lo visto en clase. Shell short siendo el mejor con ~ O(n^1.5) seguido de selection e insertion con ~ O(n^2).

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

Si se obtuvieron distintos resultados dependiendo de la máquina. Especialmente con el tiempo en ms y con el tamaño de la muestra. Para algunas máquinas con muestras pequeñas (<1000) algunos ordenamientos actuaron más rápido; no obstante, se obtuvo en general que el ARRAY\_LIST s comportó de mejor manera.

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

Estas diferencias se pueden deber a las variables específicas a cada sistema mencionadas al inicio del documento como lo puede ser la memoria (RAM), procesador y el tipo de sistema.

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

Es mejor ARRAY\_LIST