

OBSERVACIONES DE LA PRÁCTICA

Juan David Salguero 201923136
David Molina 202125176

Ambientes de pruebas

	Máquina 1	Máquina 2
Procesadores	AMD Ryzen 7	Intel Core i5
Memoria RAM (GB)	8GB	8GB
Sistema Operativo	Windows 11- 64 bits	macOs Catalina - 64 bits

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]
0.50%		0000.0	0000.0	0000.0
5.00%	219	55	55	8
10.00%	439	199	202	24
20.00%	878	905	949	40
30.00%	1317	2183	2247	70
50.00%	2195	5634	5907	150
80.00%	3512	15566	13222	269
100.00%	4391	21598	25028	365

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

Porcentaje de la muestra [pct]	Tamaño de la muestra (SINGLE_LINKED)	Insertion Sort [ms]	Selection Sort [ms]	Shell Sort [ms]
0.50%	21	1	0.000	0.000
5.00%	219	642	574	73
10.00%	439	5803	4.998	490
20.00%	878	24568	21599	1241
30.00%	1317	117589	137353	3327
50.00%	2195	627663	439717	10525
80.00%	3512	2072746	1753310	30413
100.00%	4391	tiempo excedido	tiempo excedido	

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

Algoritmo	Arreglo (ARRAY_LIST)	Lista enlazada (LINKED_LIST)
-----------	----------------------	------------------------------

<i>Insertion Sort</i>	Lento.	Más Lento
<i>Selection Sort</i>	Más lento.	Lento
<i>Shell Sort</i>	Más rápido.	Más rápido.

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

Maquina 2

Resultados

Porcentaje de la muestra [pct]	Tamaño de la muestra (ARRAY_LIST)	Insertion Sort [ms]	Selection Sort [ms]	Shell Sort [ms]
0.50%	21	0000.12	0000.12	0000.12
5.00%	219	0046.28	0053.32	0007.83
10.00%	439	0179.93	0206.93	0018.33
20.00%	878	0713.72	0831.12	0043.46
30.00%	1317	1687.30	1893.22	0070.76
50.00%	2195	4601.40	5171.57	0137.78
80.00%	3512	11910.50	13540.35	0246.89
100.00%	4391	19599.61	20990.44	0331.53

Tabla 5. Comparación de tiempos de ejecución para los ordenamientos en la representación arreglo.

Porcentaje de la muestra [pct]	Tamaño de la muestra (SINGLE_LINKED)	Insertion Sort [ms]	Selection Sort [ms]	Shell Sort [ms]
0.50%	21	0001.66	0001.31	0001.97
5.00%	219	0621.08	0581.06	0079.54
10.00%	439	4982.78	4467.77	0445.09
20.00%	878	39957.76	36945.41	2204.89
30.00%	1317	142550.43	128764.76	5738.81
50.00%	2195	666914.10	606240.90	18573.73
80.00%	3512			55005.90
100.00%	4391			

Tabla 6. Comparación de tiempos de ejecución para los ordenamientos en la representación lista enlazada.

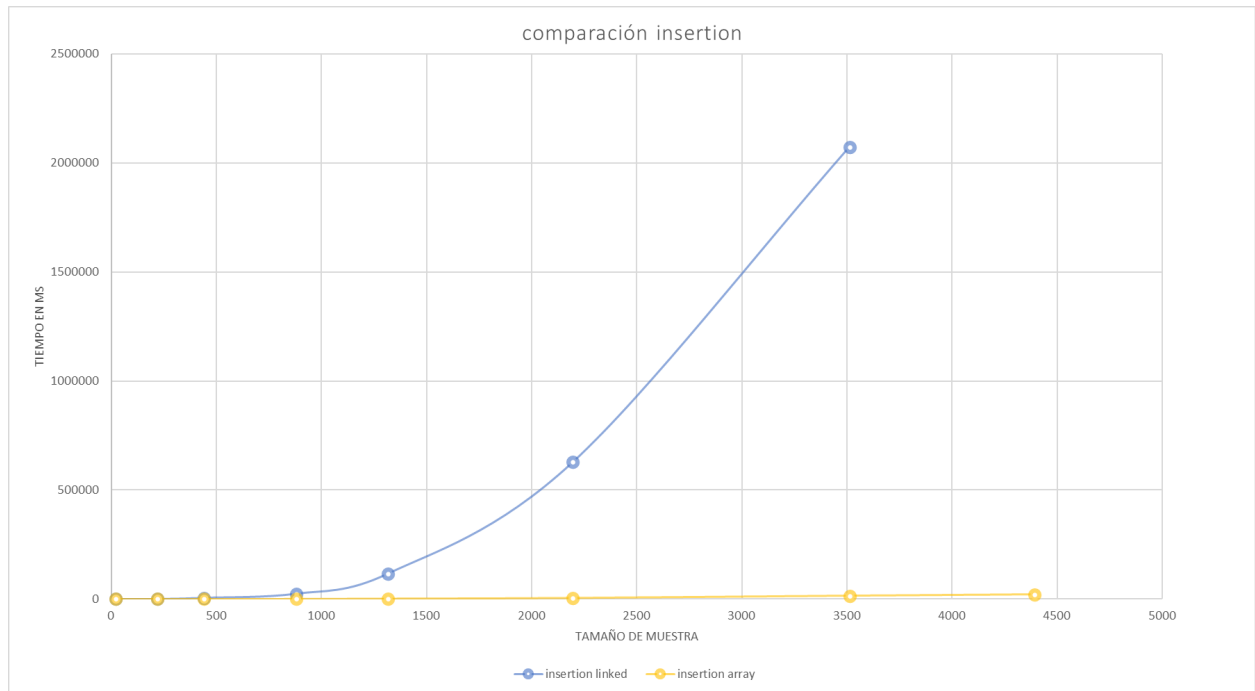
Algoritmo	Arreglo (ARRAY_LIST)	Lista enlazada (LINKED_LIST)
<i>Insertion Sort</i>	Lento.	Más Lento
<i>Selection Sort</i>	Más lento.	Lento
<i>Shell Sort</i>	Más rápido.	Más rápido.

Tabla 7. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

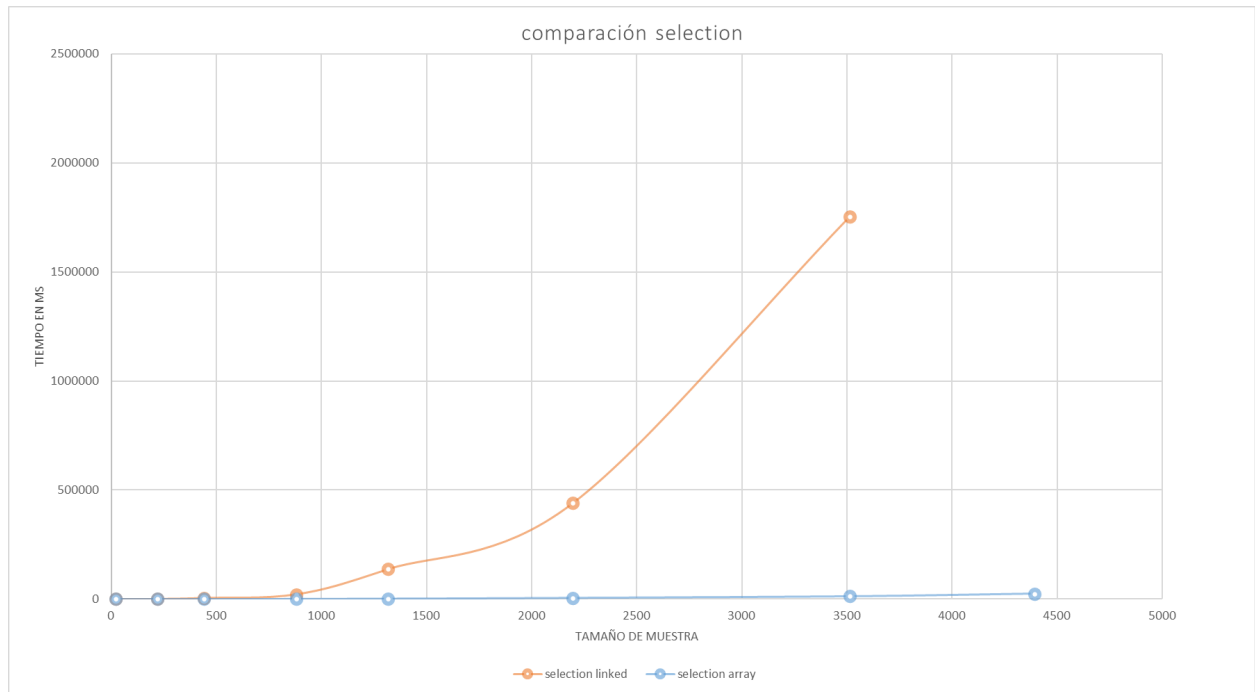
Gráficas comparativas

Maquina 1

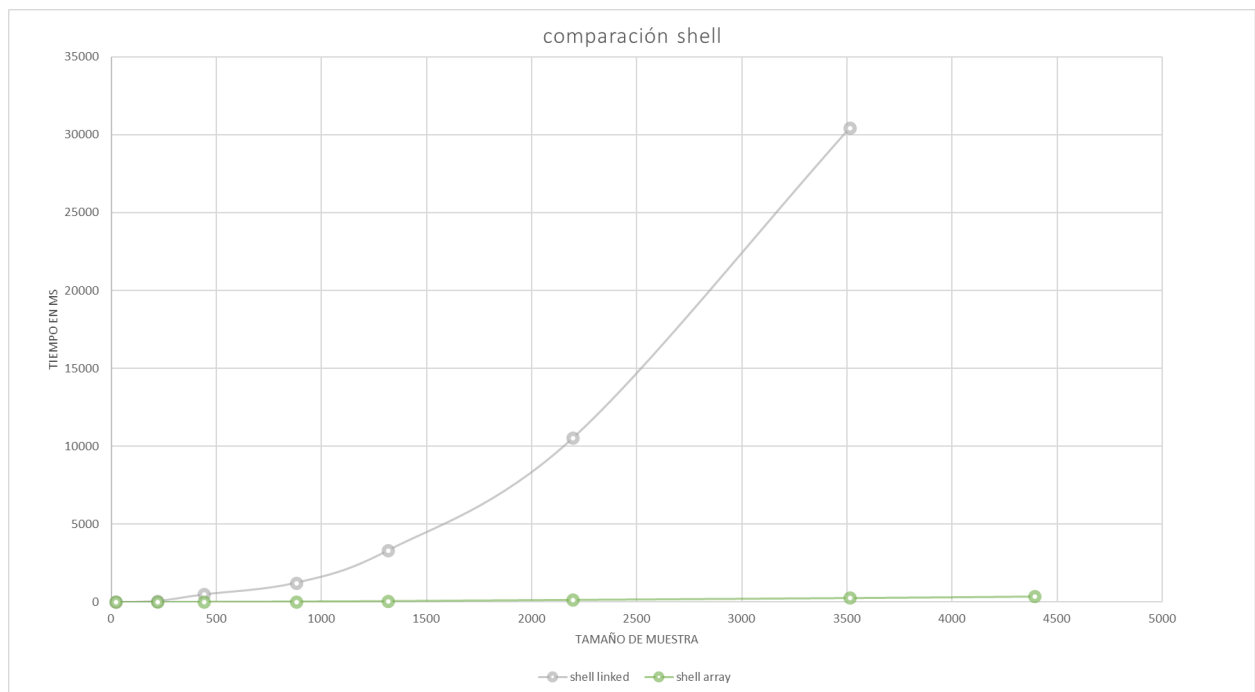
Insertion



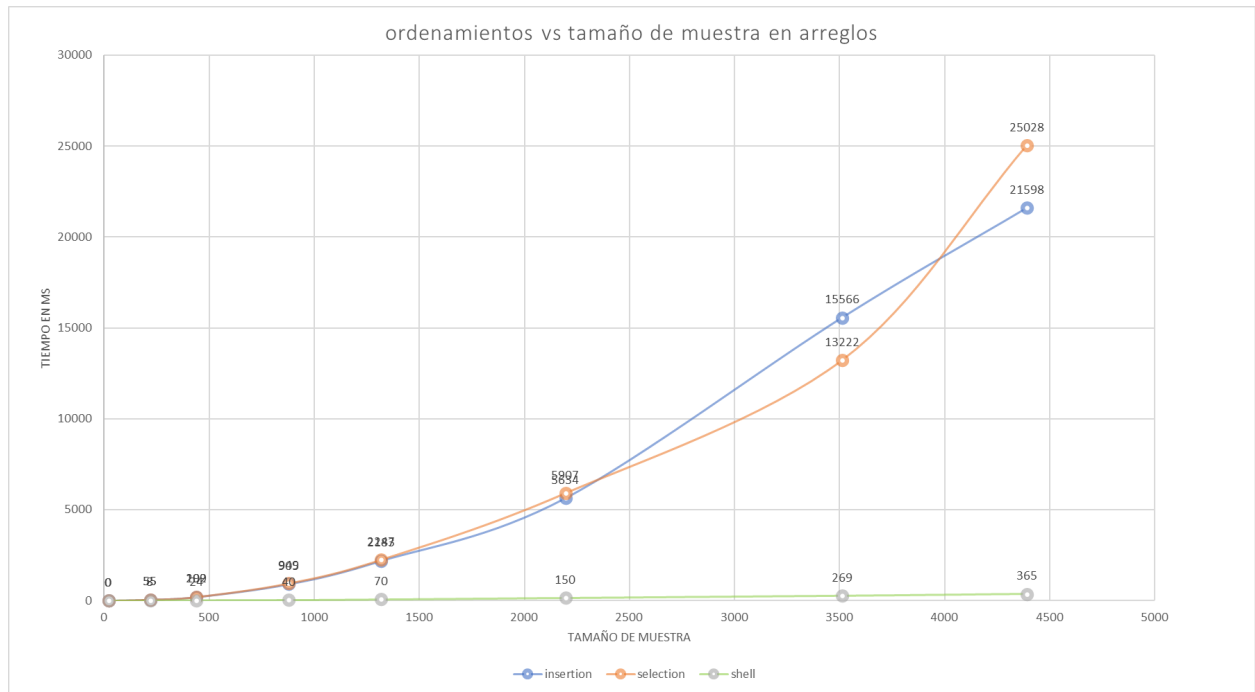
Selection



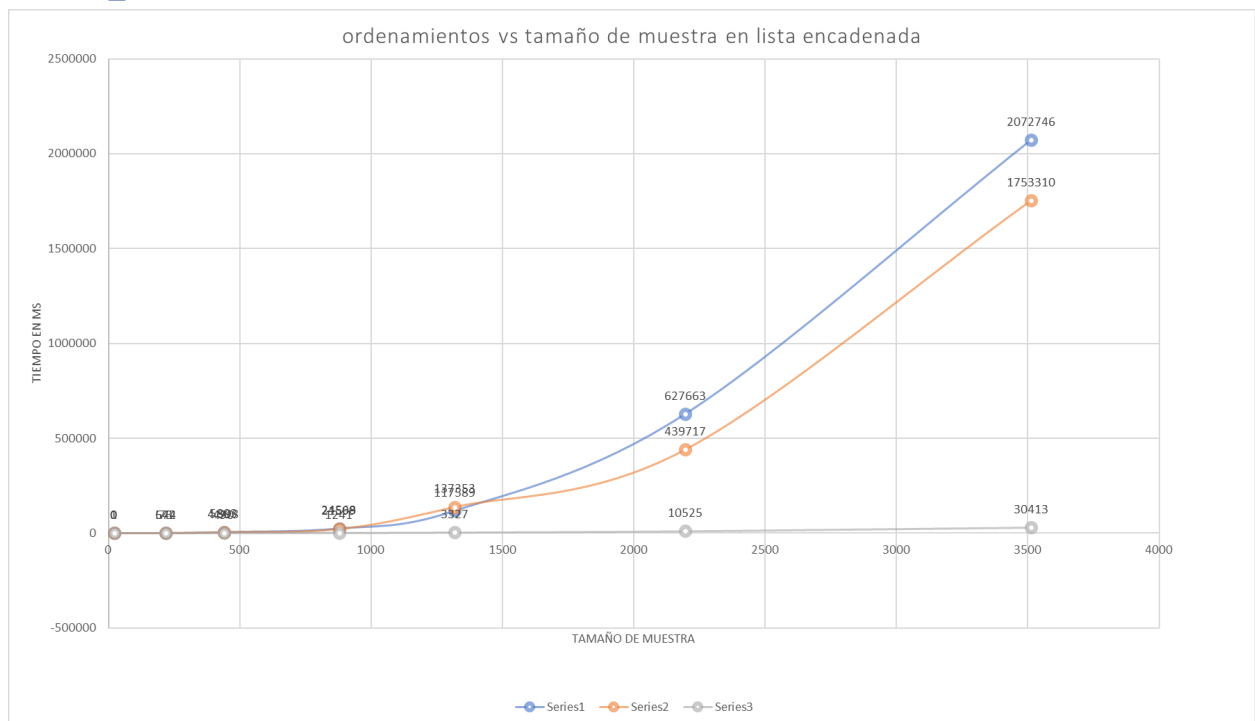
Shell

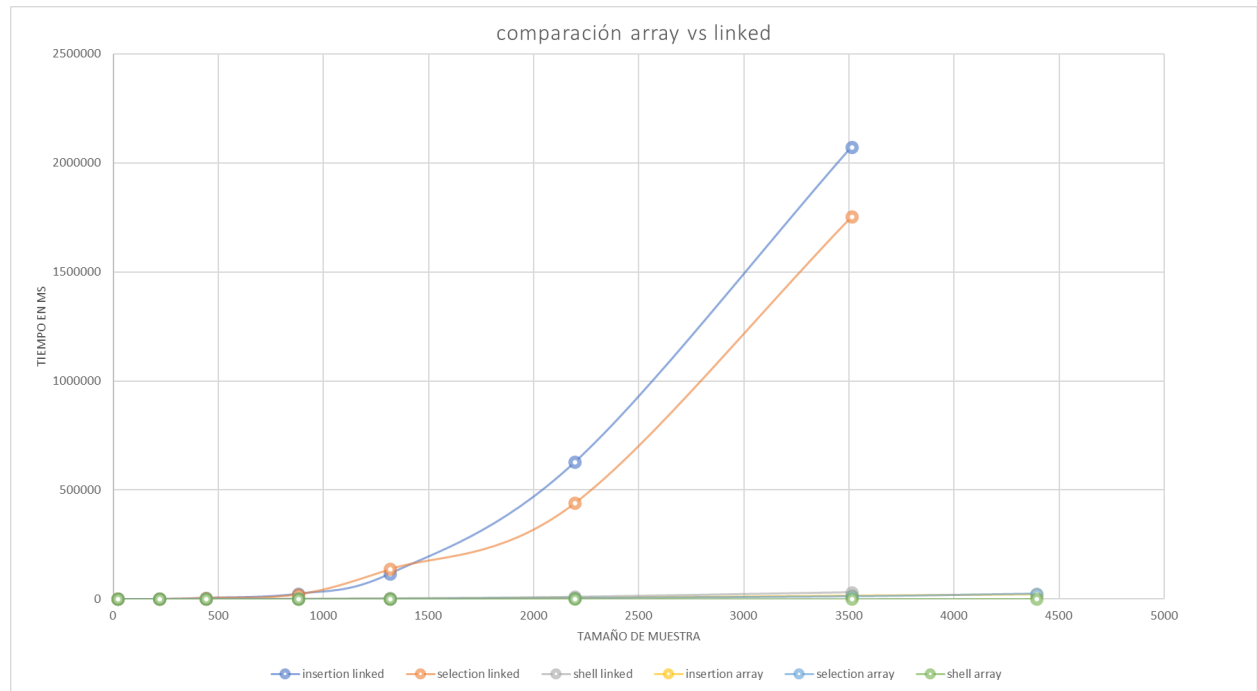


ARRAY_LIST



LINKED_LIST

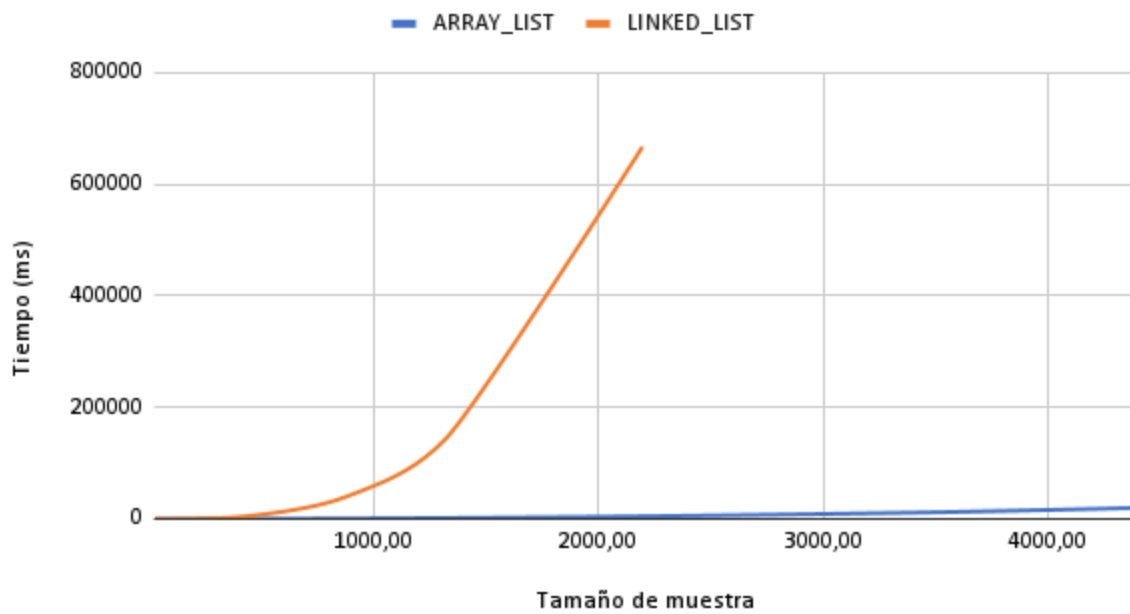




Maquina 2

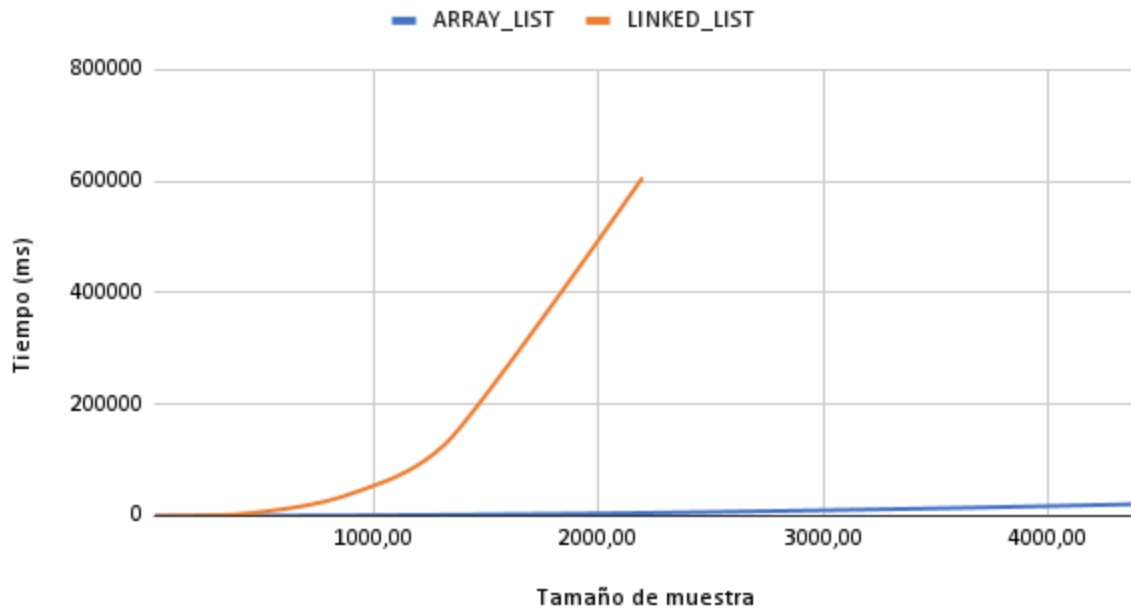
Insertion

Comparación Insertion Sort



Selection

Comparación Selection Sort



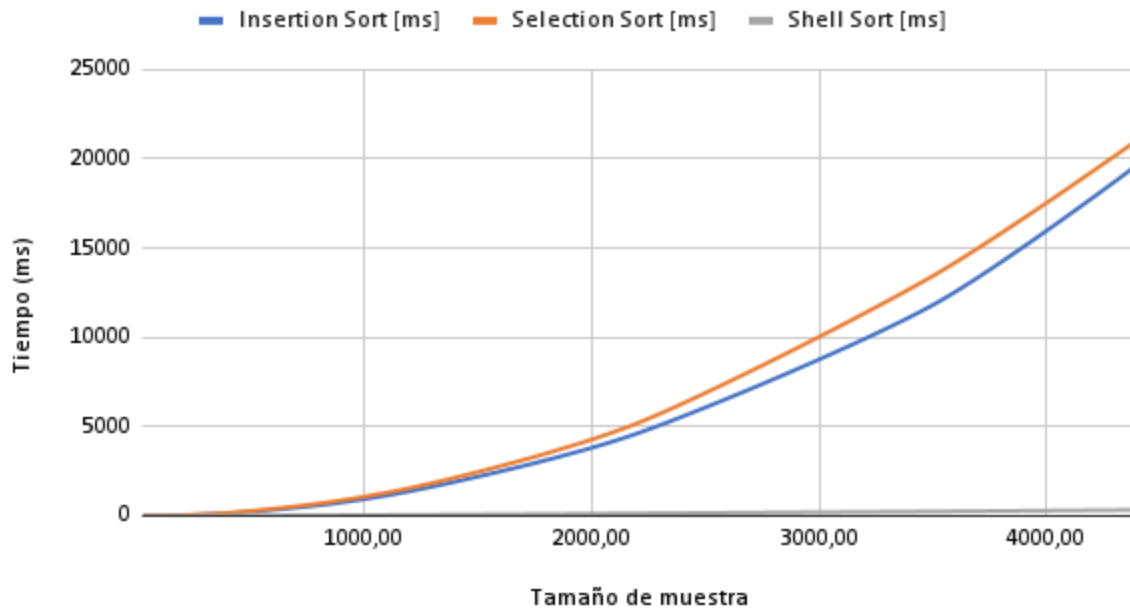
Shell

Comparación Shell Sort



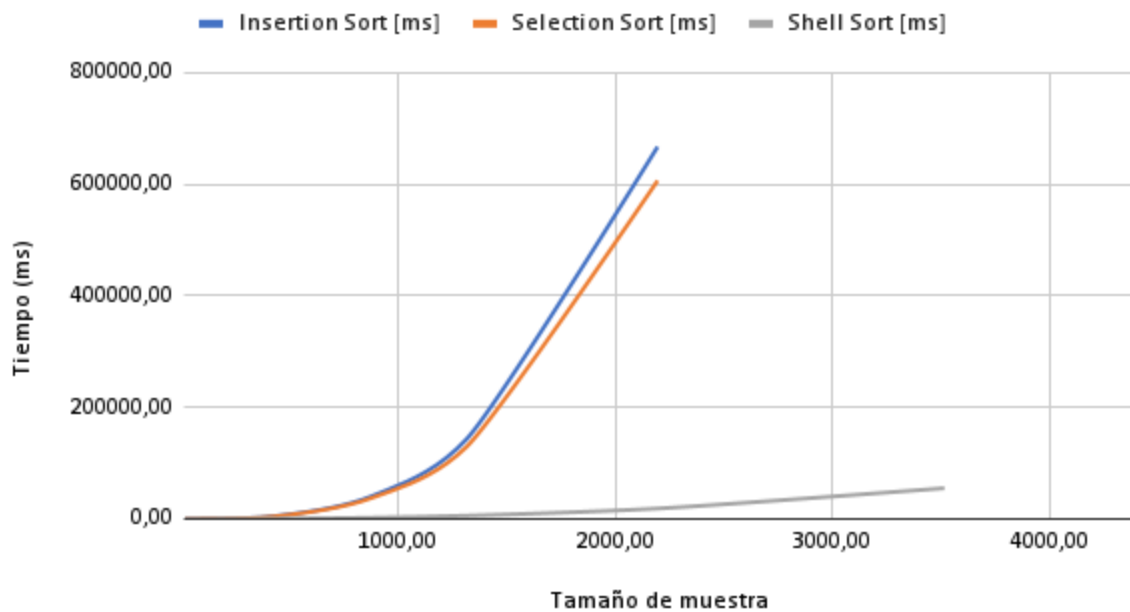
ARRAY_LIST

Desempeño de algoritmos en ARRAY_LIST

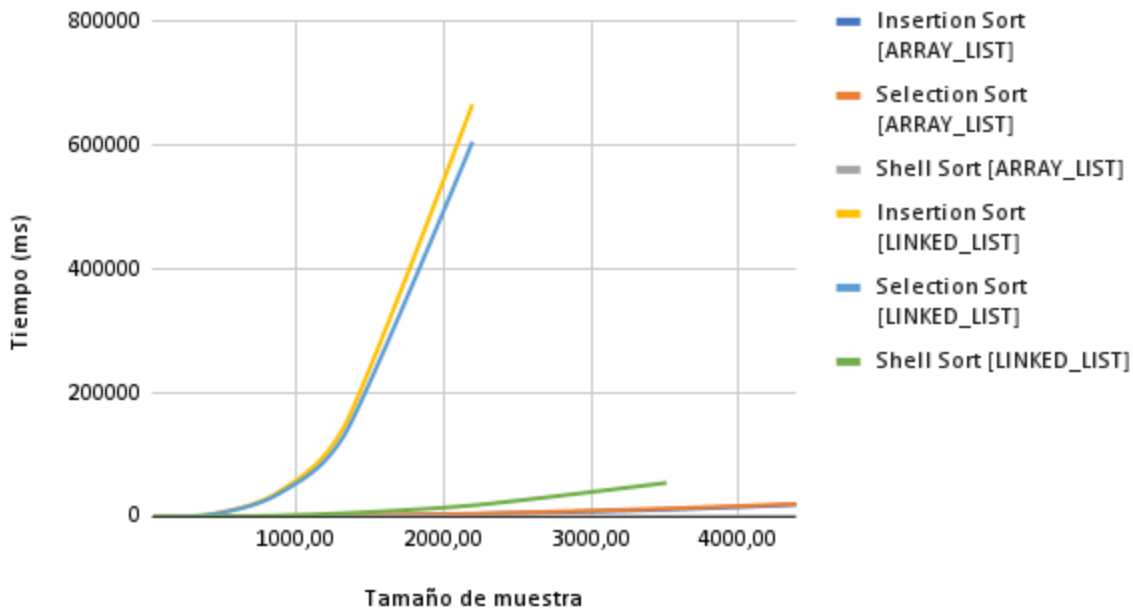


LINKED_LIST

Desempeño de algoritmos en LINKED_LIST



Comparación general según tipo de listas.



Preguntas de análisis

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

El comportamiento si es acorde a lo enunciado. Los tiempos de carga organizando una LINKED_LIST son considerablemente mayores a los de cualquier ARRAY_LIST. Por otra parte, hay poca diferencia entre los algoritmos de organización al ser aplicados a pocos datos; con muchos datos, sin embargo, la eficiencia y rapidez de algoritmos como shell sale a relucir.

- 2) ¿Existe alguna diferencia entre los resultados obtenidos al ejecutar las pruebas en diferentes máquinas?

No parece haber una variable que denote alguna diferencia constante. En algunos casos la máquina 1 ejecuta los procedimientos más velozmente que la máquina 2, y viceversa. Siendo así, la efectividad de cada algoritmo de ordenamiento se mantiene constante independientemente de la máquina que lo ejecuta.

- 3) De existir diferencias, ¿A qué creen ustedes que se deben dichas diferencias?

Las pocas diferencias probablemente se deben a la memoria ram disponible al ejecutarse el programa (valor el cual aunque es por predeterminado el mismo en las dos máquinas, puede

variar de acuerdo al estado de uso de las mismas.) Por otro lado, que tan velozmente el hardware de cada máquina logra realizar cada procedimiento, tendrá algún efecto.

- 4) ¿Cuál Estructura de Datos es mejor utilizar si solo se tiene en cuenta los tiempos de ejecución de los algoritmos?

Shell Sort, con gran ventaja, se muestra según las pruebas como el algoritmo más veloz en cada ocasión.