

OBSERVACIONES DE LA PRACTICA

Raul Santiago Rincon Corba Cod 202120414
Benjamin Raisbeck Garcia Cod 202120398

	Máquina 1	Máquina 2
Procesadores	AMD Ryzen 5 3600 3.6 GHz Processor 3.6 GHz	Intel® Core™ i7-10700 CPU @ 2.90GHz 2.9 GHz
Memoria RAM (GB)	16 GB	16 GB
Sistema Operativo	Windows 10 Pro - 64 bits	Windows 11 - 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]	Quick Sort [ms]	Merge Sort [ms]
0.50%	500	177.65	205.103	15.786	11.530	13.977
5.00%	4300	14304.876	15785.559	224.461	151.324	155.574
10.00%	9000	62278.473	69975.255	537.923	335.342	349.363
20.00%	18000	258055.105	289552.053	1341.513	893.044	754.150
30.00%	26000	537430.596	608930.142	2223.188	1384.374	1151.012
50.00%	41000	1374220.501	1561579.045	3629.749	2572.856	1844.995
80.00%	47000	No TIME	No TIME	4023.148	3188.306	2271.432
100.00%	56000	No TIME	No TIME	5000.496	4159.859	2785.007

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%	500	5996.840	5266.347	463.065	376.720	68.567
5.00%	4300	No Time	No Time	66326.984	45530.023	4316.836
10.00%	9000	No Time	No Time	369282.247	204681.915	19456.286
20.00%	18000	No Time	No Time	1542140.176	1153800.593	90215.256
30.00%	26000	No Time	No Time	No Time	No Time	451984.133
50.00%	41000	No Time	No Time	No Time	No Time	1305045.155
80.00%	47000	No Time	No Time	No Time	No Time	No Time

100.00%	56000	No Time	No Time	No Time	No Time	No Time
---------	-------	---------	---------	---------	---------	---------

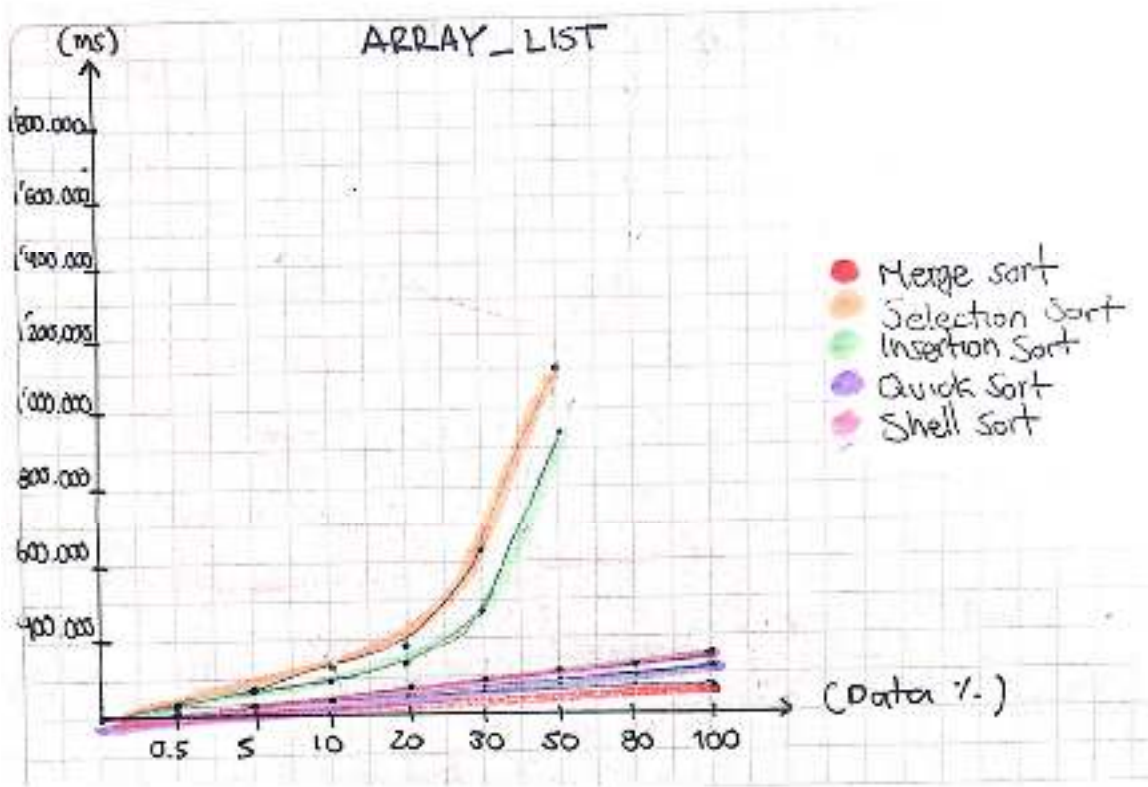
Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

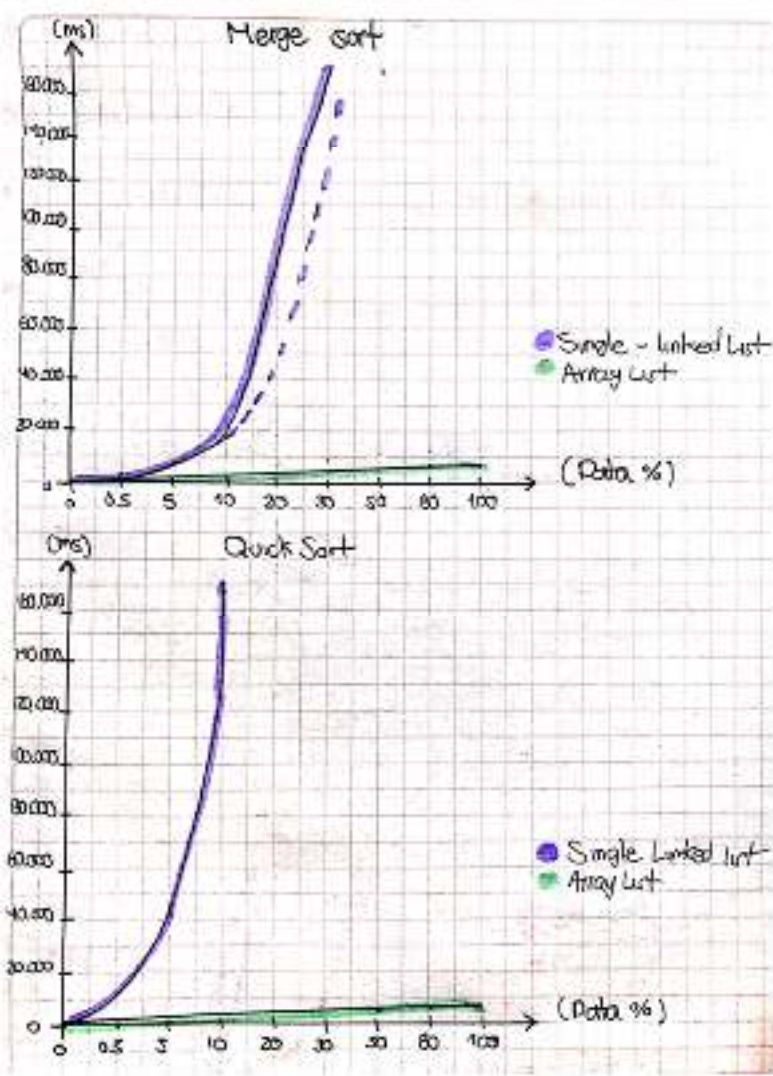
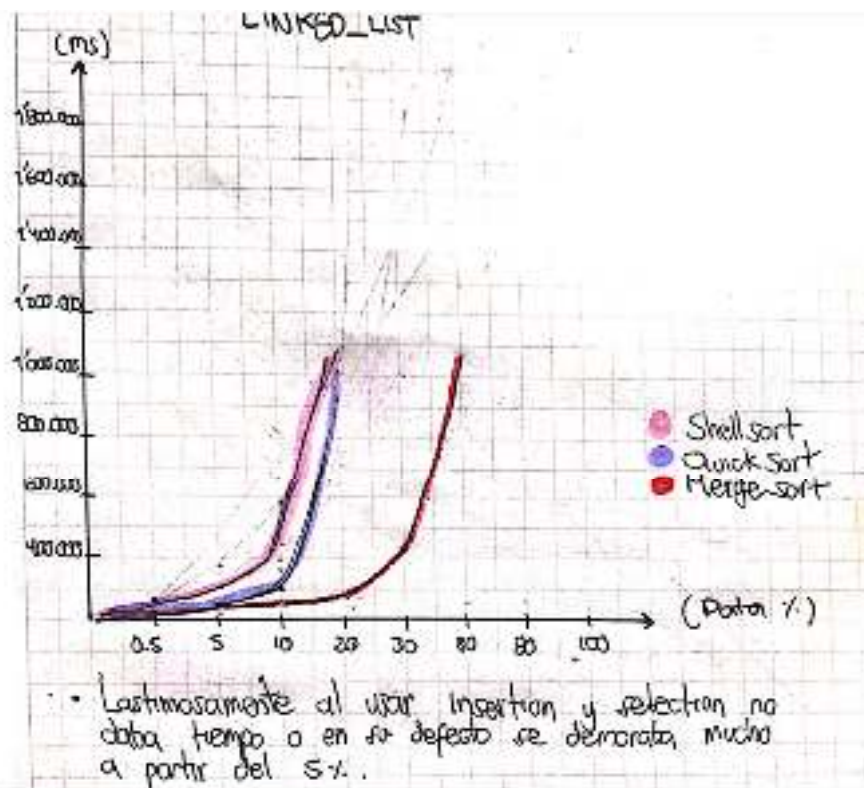
Algoritmo	Arreglo (ARRAYLIST)	Lista enlazada (LINKED_LIST)
Merge sort		x
Quick sort	x	

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%	500	127.680	155.411	13.3	8.474	9.276
5.00%	4000	10,500.192	11,100.568	155.9	93.45	99.68
10.00%	8000	36,729.751	44,162.341	356.5	210.83	210.4
20.00%	15000	133,654.891	158,345.272	798.7	474.97	464.4
30.00%	21000	247,534.126	277,054.458	981.5	719.36	621.6
50.00%	30000	477,595.934	565,907.256	1,615.5	1,168	874.9
80.00%	45000			2,515.1	2,315	1,333
100.00%	55000			3,187	2,980	1,666

Tabla 5. 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%	500	3,644.227	3,227.979	262.749	215.49	42.843
5.00%	4000			33,820.1	22,522	2,237.97
10.00%	8000			163,293.1	97,119	8,971.88
20.00%	15000				414,084.2	32,839.7
30.00%	21000					62,678.3
50.00%	30000					127,615
80.00%	45000					313,080
100.00%	55000					

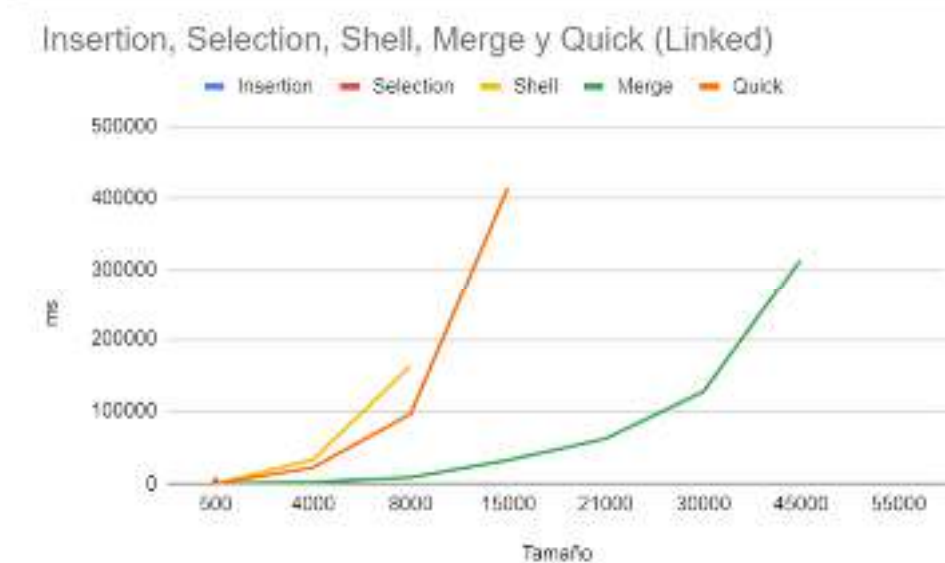
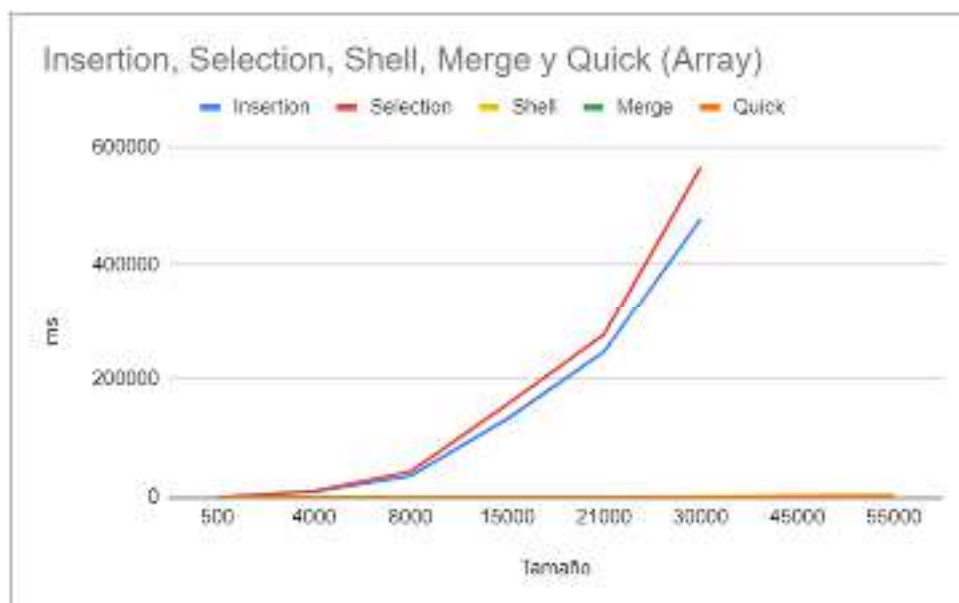
Tabla 6. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

Algoritmo	Arreglo (ARRAYLIST)	Lista enlazada (LINKED_LIST)
Merge sort	x	
Quick sort	x	

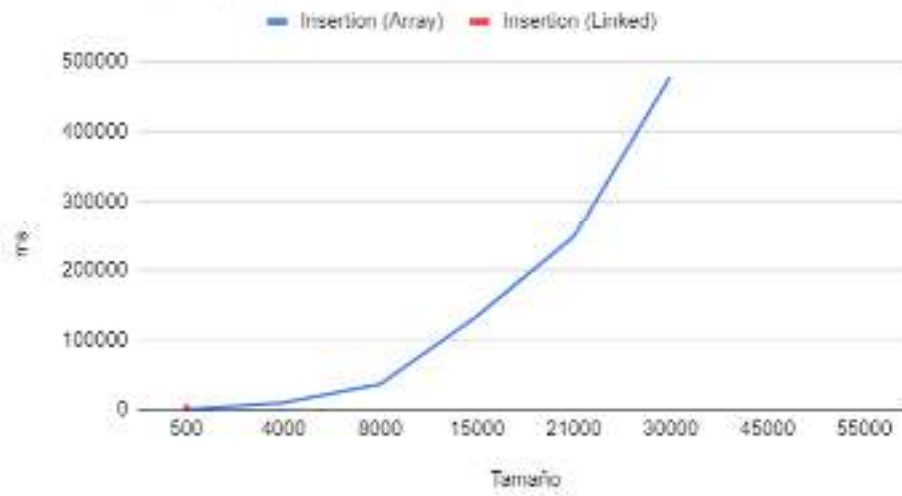
Tabla 7. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

Graficas

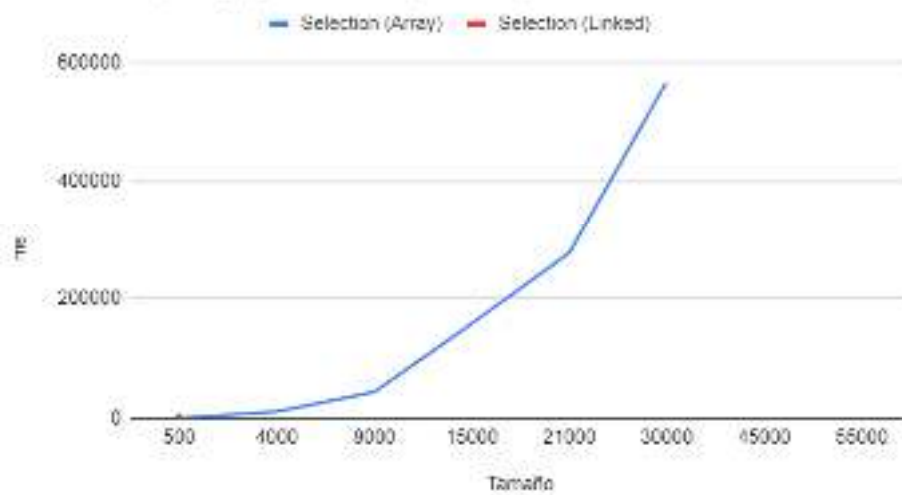
- Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 2**.
 - Comparación de rendimiento ARRAYLIST.
 - Comparación de rendimiento LINKED_LIST.
 - Comparación de rendimiento para Insertion Sort.
 - Comparación de rendimiento para Selection Sort.
 - Comparación de rendimiento para Shell Sort.
 - Comparación de rendimiento para MergeSort.
 - Comparación de rendimiento para QuickSort.



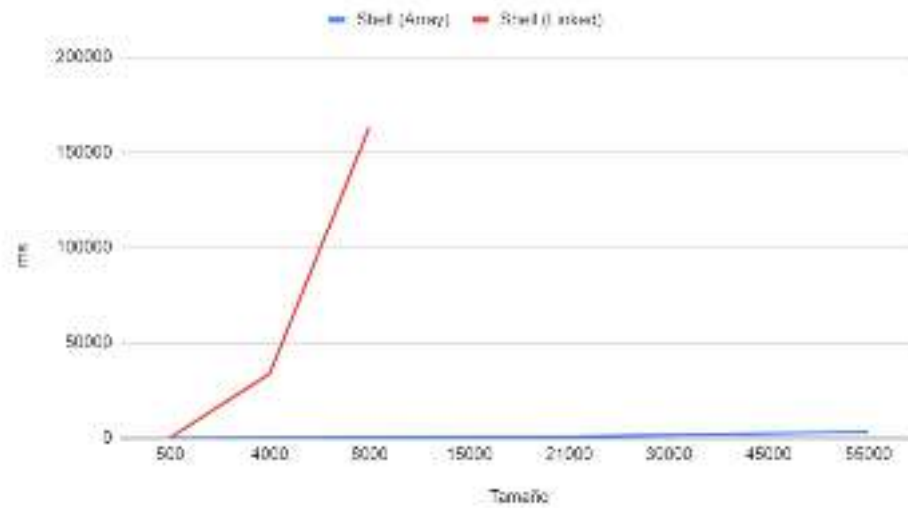
Insertion (Array) y Insertion (Linked)



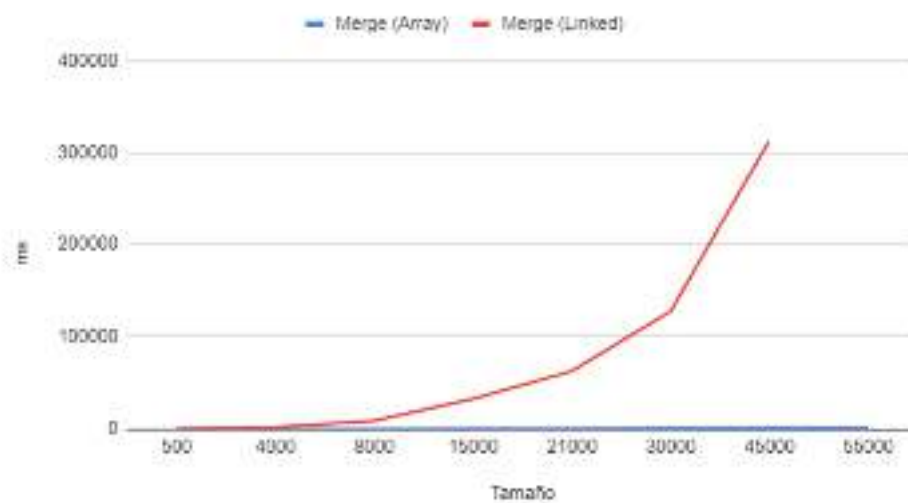
Selection (Array) y Selection (Linked)

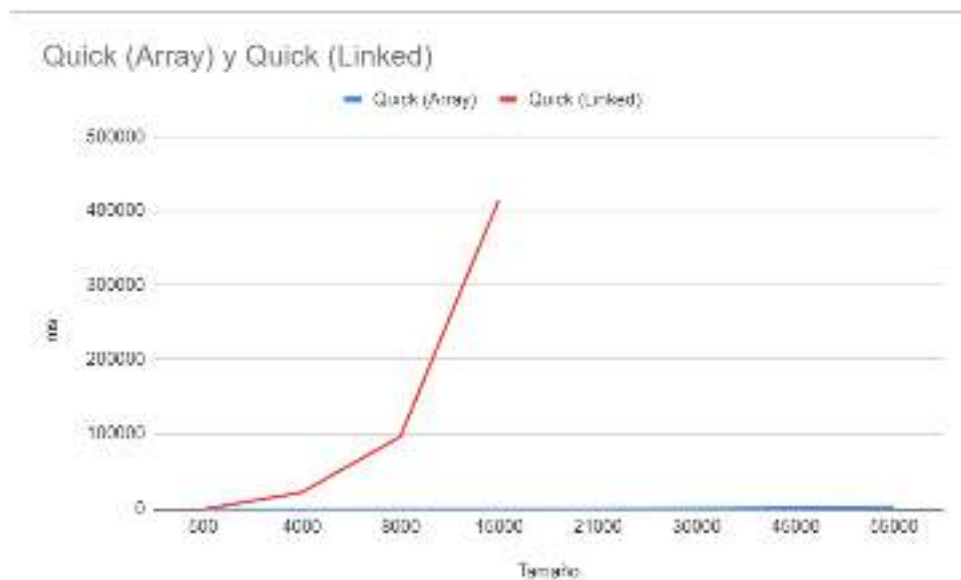


Shell (Array) y Shell (Linked)



Merge (Array) y Merge (Linked)





Preguntas de análisis

- ¿El comportamiento de los algoritmos es acorde a lo enunciado teóricamente?
 - Si aun teniendo algunos porcentajes de datos donde los algoritmos se demoraban mucho.
- ¿Existe alguna diferencia entre los resultados obtenidos al ejecutar las pruebas en diferentes máquinas?
 - Si. En la maquina 2 va mucho más rápido los algoritmos de ordenamiento que en la primera. Además, en linked los resultado son muy distintos.
- De existir diferencias, ¿A qué creen ustedes que se deben dichas diferencias?
 - Principal a la diferencia de arquitecturas entre maquinas y segundo probablemente a los procesos corriendo en segundo plano o los diferentes servicios en segundo plano de las maquinas
- ¿Cuál Estructura de Datos es mejor utilizar si solo se tiene en cuenta los tiempos de ejecución de los algoritmos?
 - En este caso como no necesitamos agregar ni quitar datos a la estructura la mejor es el Array List ya que en cualquiera de los 5 algoritmos de ordenamiento va mucho más rápido que usando Single Linked List.
- 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.

EFICIENCIA ALGORITMOS DE MEJOR A PEOR (EN TIEMPOS DE EJECUCIÓN)	
1 (MEJOR)	MERGE
2	QUICK
3	SHELL
4	INSERTION
5 (PEOR)	SELECTION

