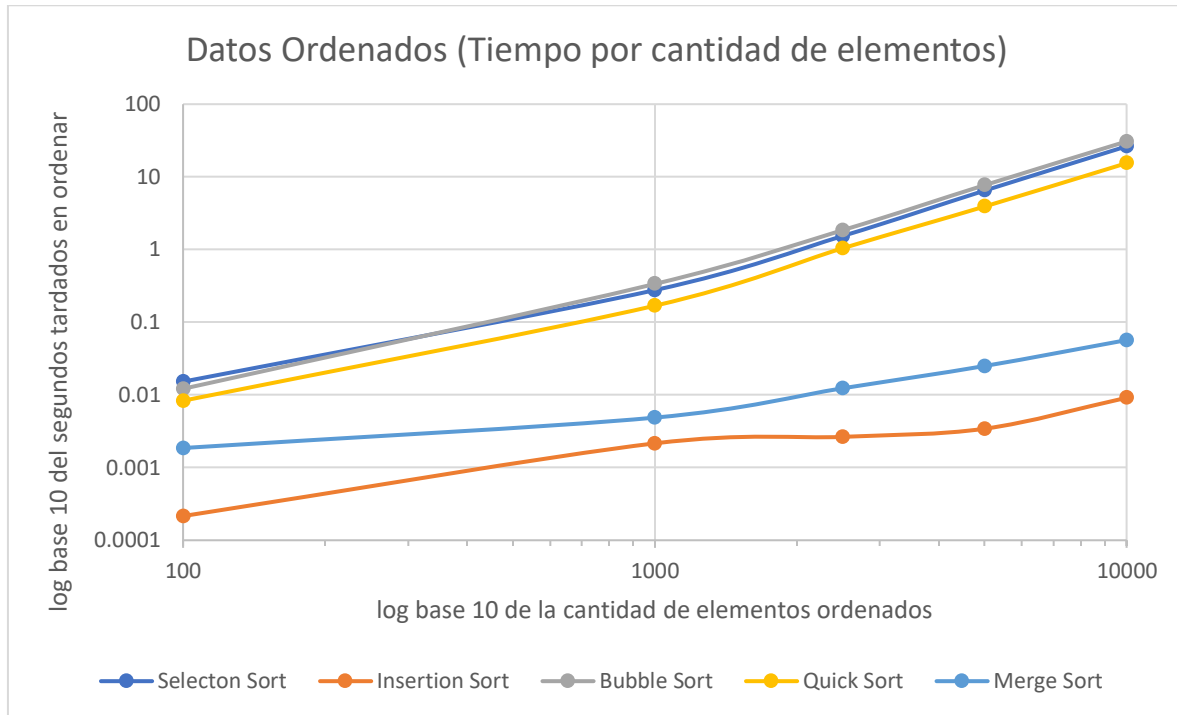
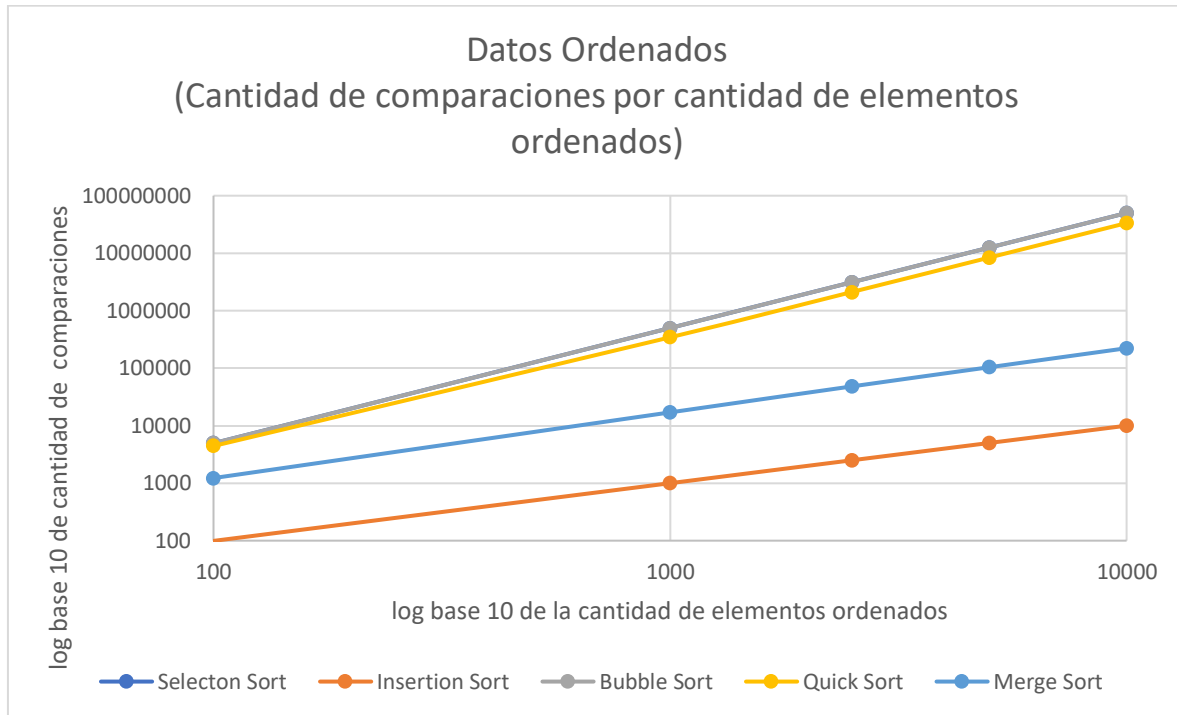


## Tarea 1

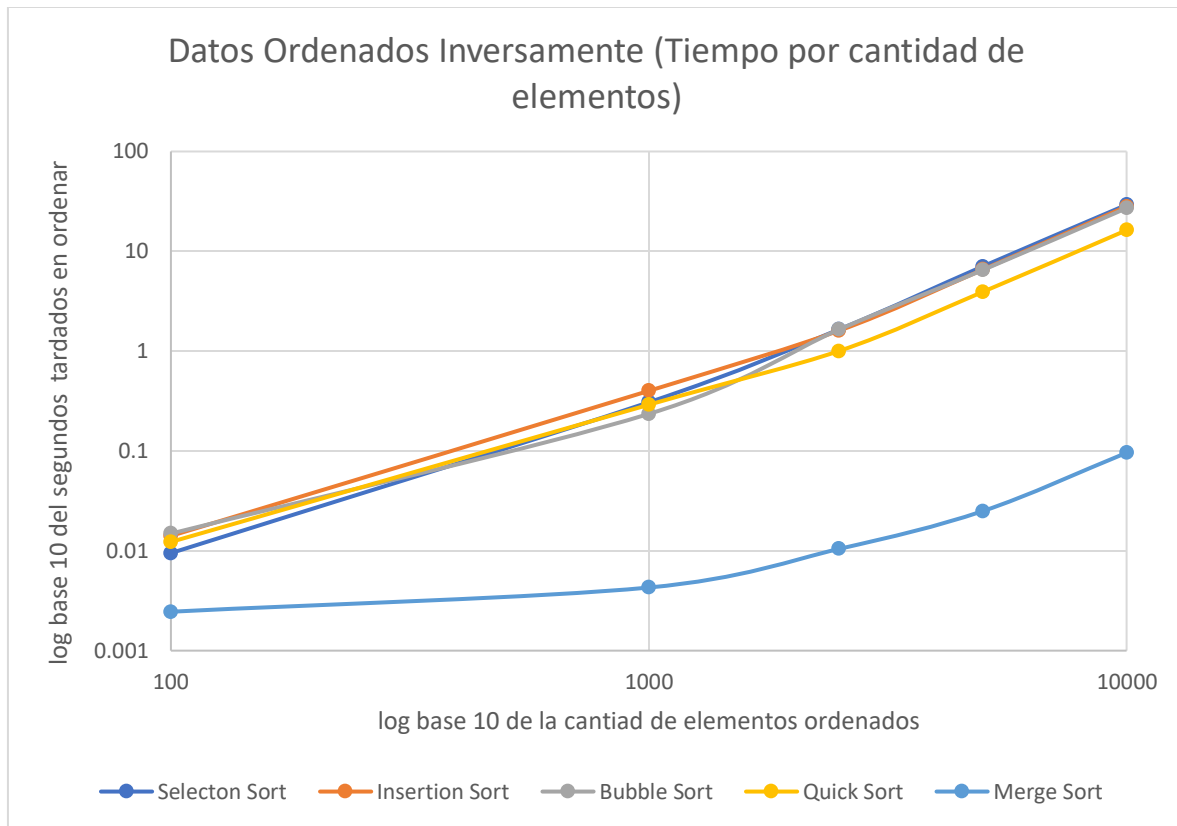
Gráfica 1:



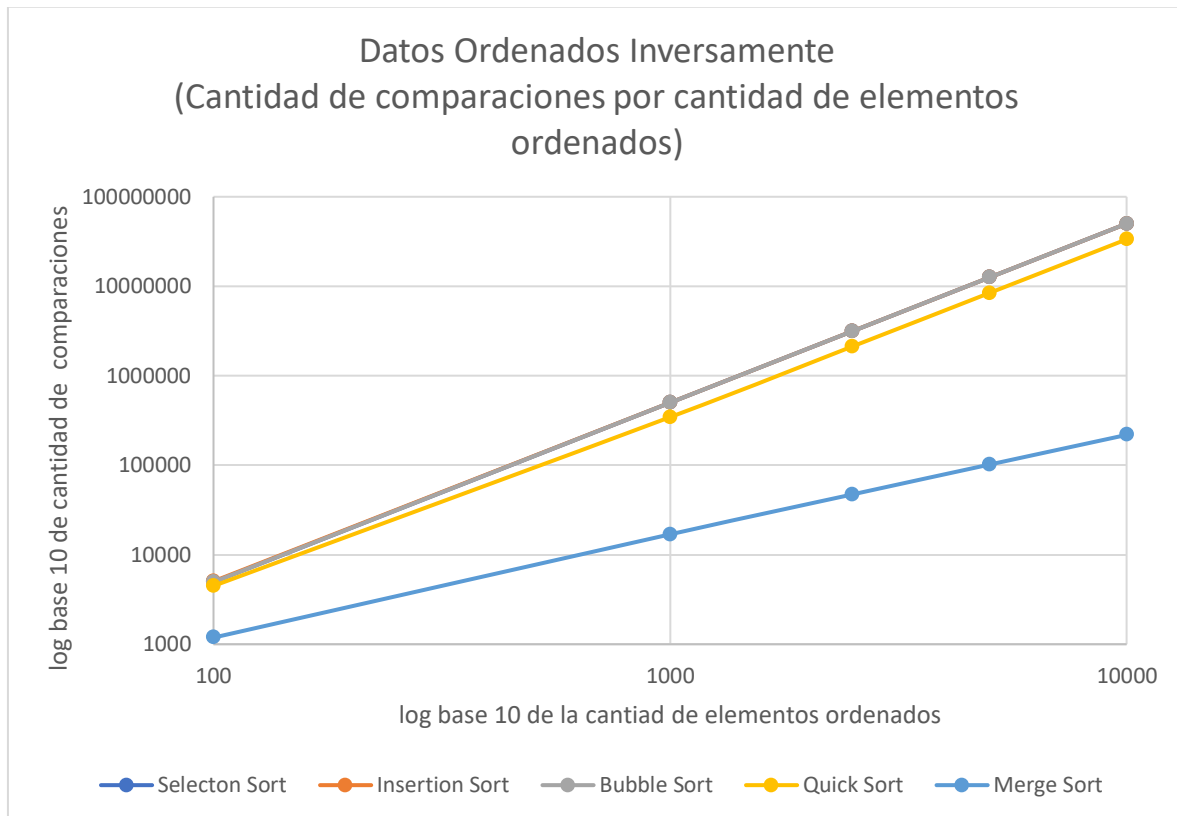
Gráfica 2:



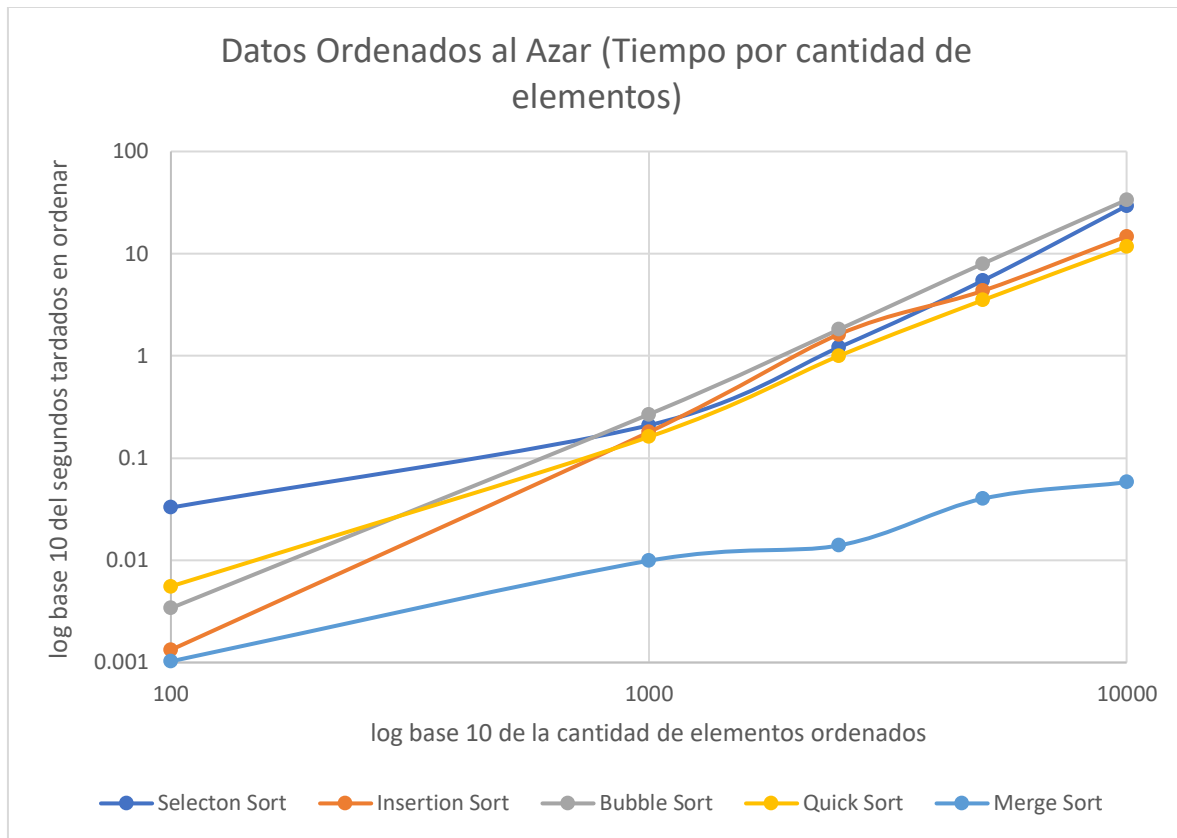
Gráfica 3:



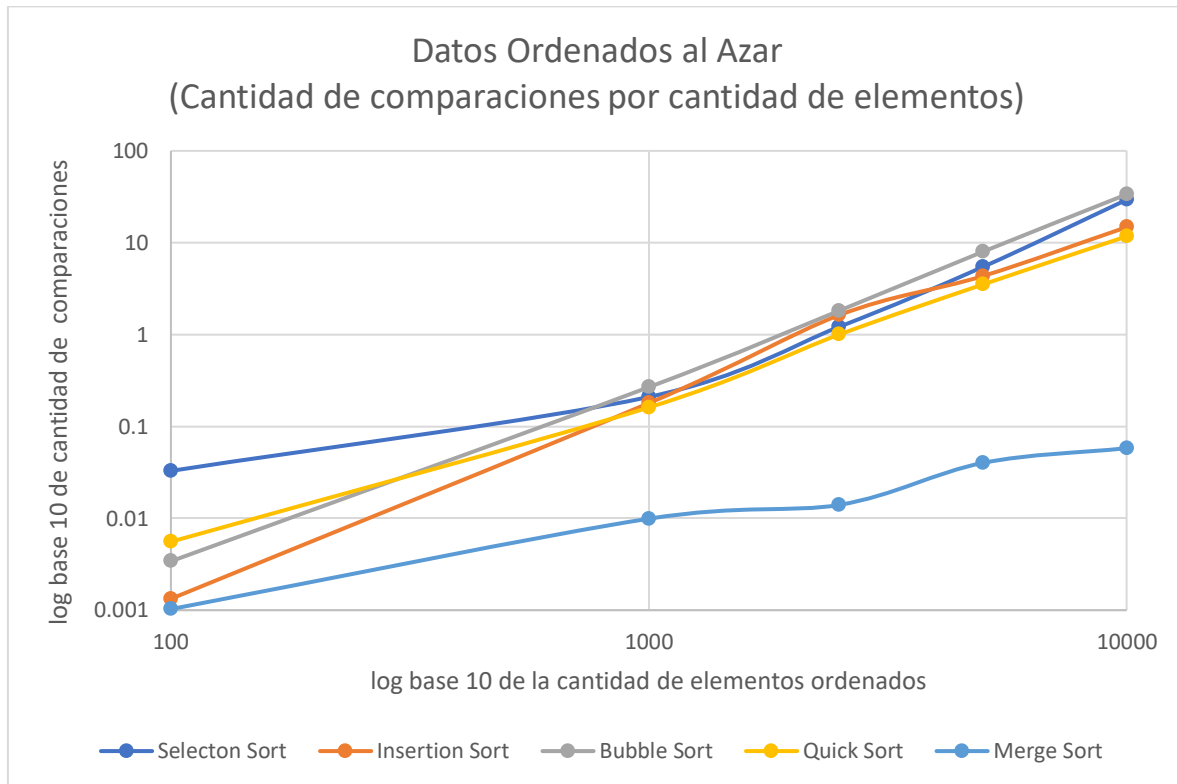
Gráfica 4:



Gráfica 5:



Gráfica 6:



## Interpretaciones:

El algoritmo más efectivo en todos los casos (tanto en tiempo como en comparaciones) fue mergesort. En general todos los algoritmos escalan de manera que un incremento de 100% a la cantidad de datos que tienen que ordenar significa un incremento desde 100% (en el mejor de los casos) hasta 1000% en la cantidad de comparaciones y tiempo requeridos para ordenarlos. No hubo un algoritmo que tuviera siempre el peor desempeño, en general la mayoría de los algoritmos tendían a una línea similar en las gráficas, ya que todos ellos casi siempre están en complejidad  $O(n^2)$

Como se puede ver en la gráfica 1 y 2, insertionsort está en su caso ideal cuando los datos están ordenados, es por esto por lo que escala linealmente, con complejidad  $O(n)$ .

Creo que, al momento de elegir un algoritmo de ordenamiento, uno se debe de fijar principalmente en la estructura de sus datos, desde algo tan simple como: ¿qué tan ordenados están ya los datos? Hasta cosas más complicadas como la distribución de los datos. No me cuesta trabajo imaginar un algoritmo de ordenamiento que aprovecha que los datos tengan una distribución normal para hacerlo más eficiente, o que puede asegurar que al menos el 99% de los datos quedarán ordenados ganando eficiencia con esa incertidumbre.

## Datos brutos:

TIEMPO					
ORDENADOS	100	1000	2500	5000	10000
Selecton Sort	0.01521736	0.27626826	1.53383471	6.46491395	26.2848325
Insertion Sort	0.00021509	0.00215368	0.00264634	0.00342371	0.00912252
Bubble Sort	0.01213331	0.33576442	1.84525611	7.68364195	30.7184988
Quick Sort	0.00831303	0.16859637	1.03989622	3.90493959	15.4296058
Merge Sort	0.00185749	0.00486528	0.01232492	0.02497357	0.05671023
INVERTIDOS	100	1000	2500	5000	10000
Selecton Sort	0.00948309	0.30738602	1.64969579	7.0601363	29.2268106
Insertion Sort	0.01410088	0.39984507	1.60232409	6.55458839	28.1694321
Bubble Sort	0.01487245	0.23496338	1.64422414	6.48675786	27.0600818
Quick Sort	0.01226852	0.29036177	1.00272392	3.92341718	16.2761463
Merge Sort	0.00245233	0.00428736	0.01043993	0.02498698	0.09591486
	100	1000	2500	5000	10000
Selecton Sort	0.03280273	0.20946866	1.21159161	5.44652776	29.3369447
Insertion Sort	0.00132908	0.17928221	1.62404959	4.33334773	14.7725481
Bubble Sort	0.00340548	0.26748841	1.81492813	7.96748324	33.594627
Quick Sort	0.00555689	0.16094628	0.99851807	3.50741664	11.7103418
Merge Sort	0.00102698	0.00989847	0.01397582	0.0400646	0.05798535

COMPARACIONES					
ORDENADOS	100	1000	2500	5000	10000
Selecton Sort	4950	499500	3123750	12497500	49995000
Insertion Sort	99	999	2499	4999	9999
Bubble Sort	4950	499500	3123750	12497500	49995000
Quick Sort	4432	344179	2110245	8387435	33441188
Merge Sort	1226	17018	48154	103810	222622
INVERTIDOS	100	1000	2500	5000	10000
Selecton Sort	4950	499500	3123750	12497500	49995000
Insertion Sort	5049	500499	3126249	12502499	50004999
Bubble Sort	4950	499500	3123750	12497500	49995000
Quick Sort	4483	344680	2111496	8389936	33446189
Merge Sort	1186	16906	47054	101610	218222
ALEATORIO	100	1000	2500	5000	10000
Selecton Sort	4950	499500	3123750	12497500	49995000
Insertion Sort	2529	248413	1559240	6244789	24915605
Bubble Sort	4950	499500	3123750	12497500	49995000
Quick Sort	5075	466898	3031909	11795764	23733971
Merge Sort	1247	17430	49311	106395	228343