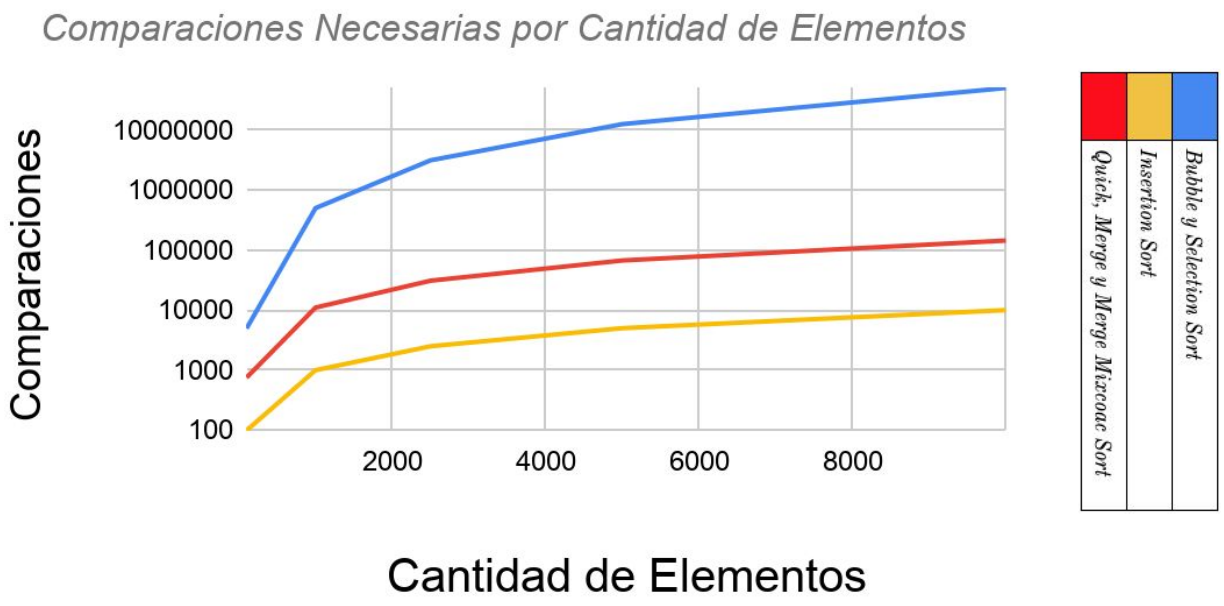
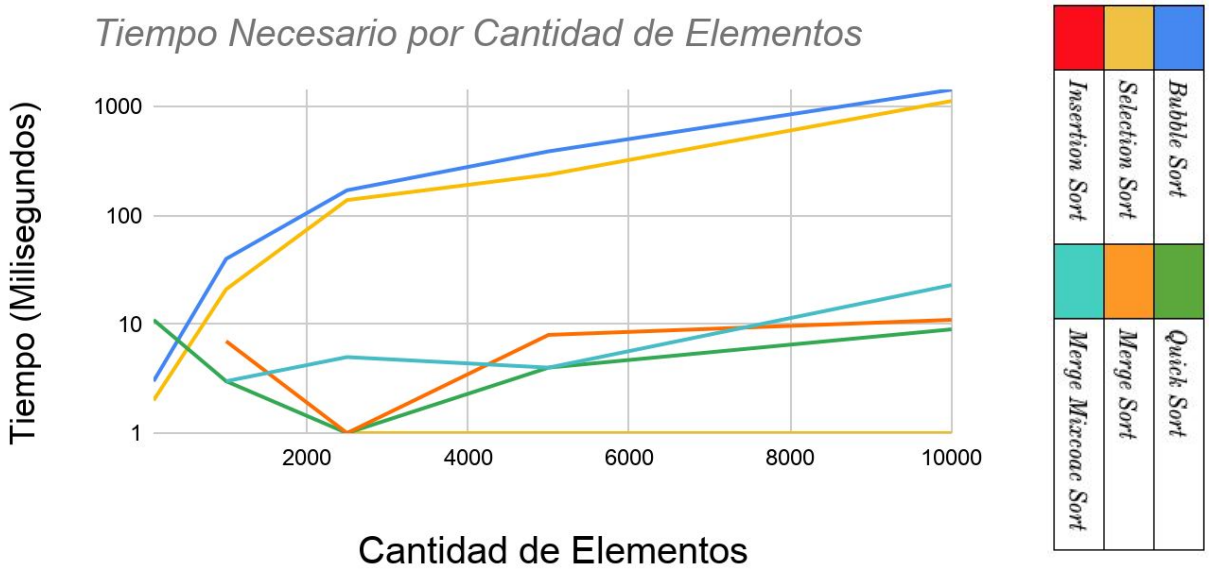
Jorge Edgar Rodriguez Ortiz Loyola 181334

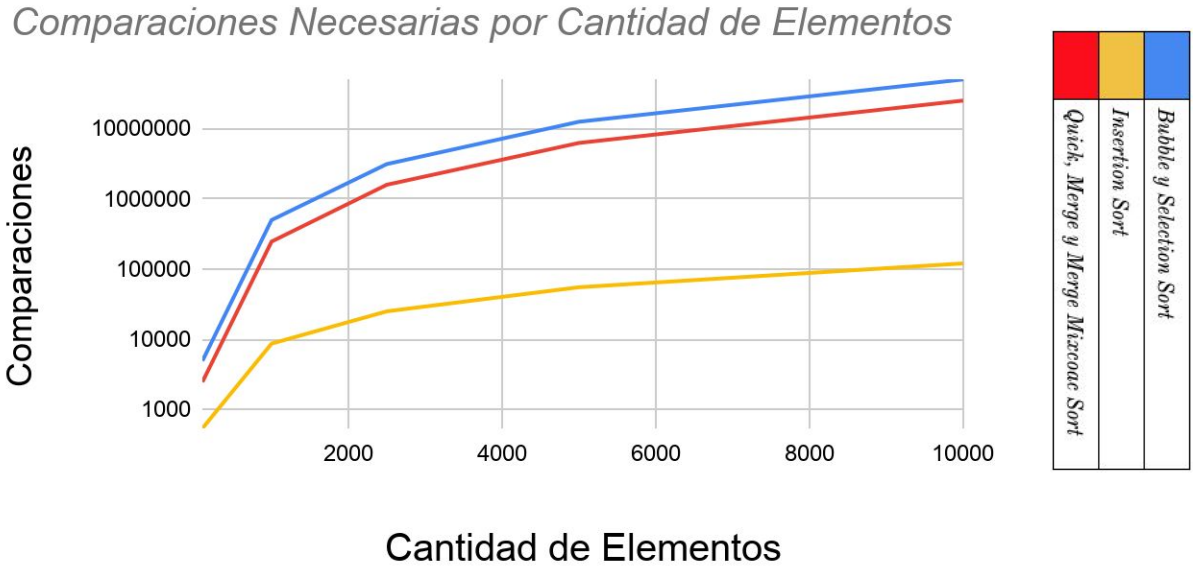
Estructuras de Datos Avanzadas
Tarea

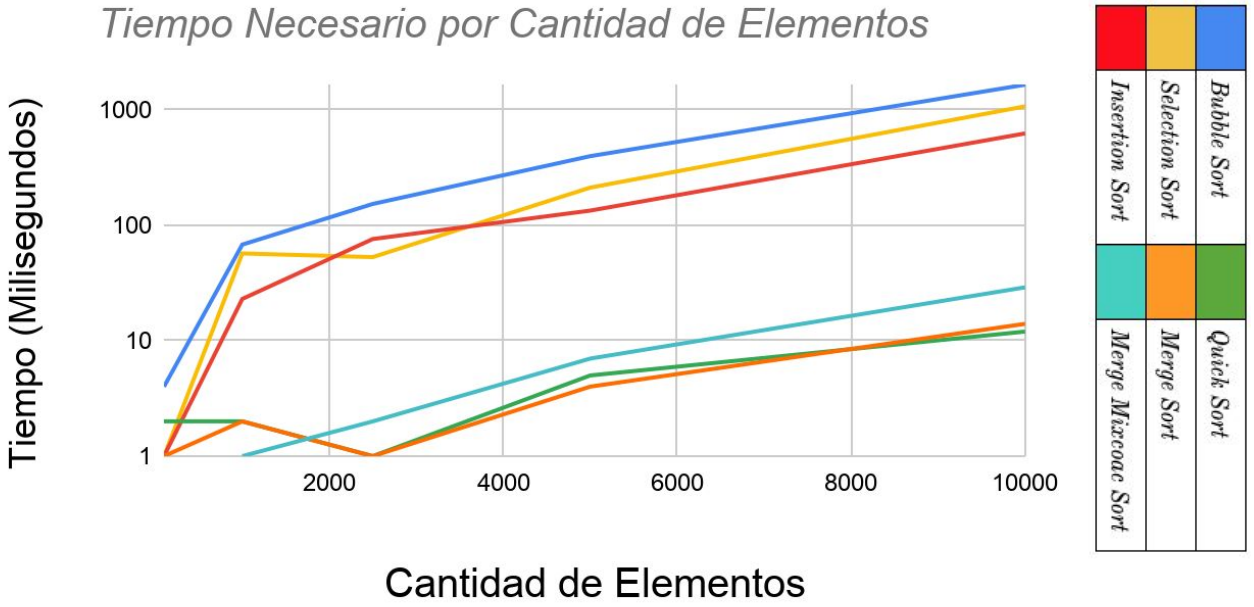
Datos Ordenados



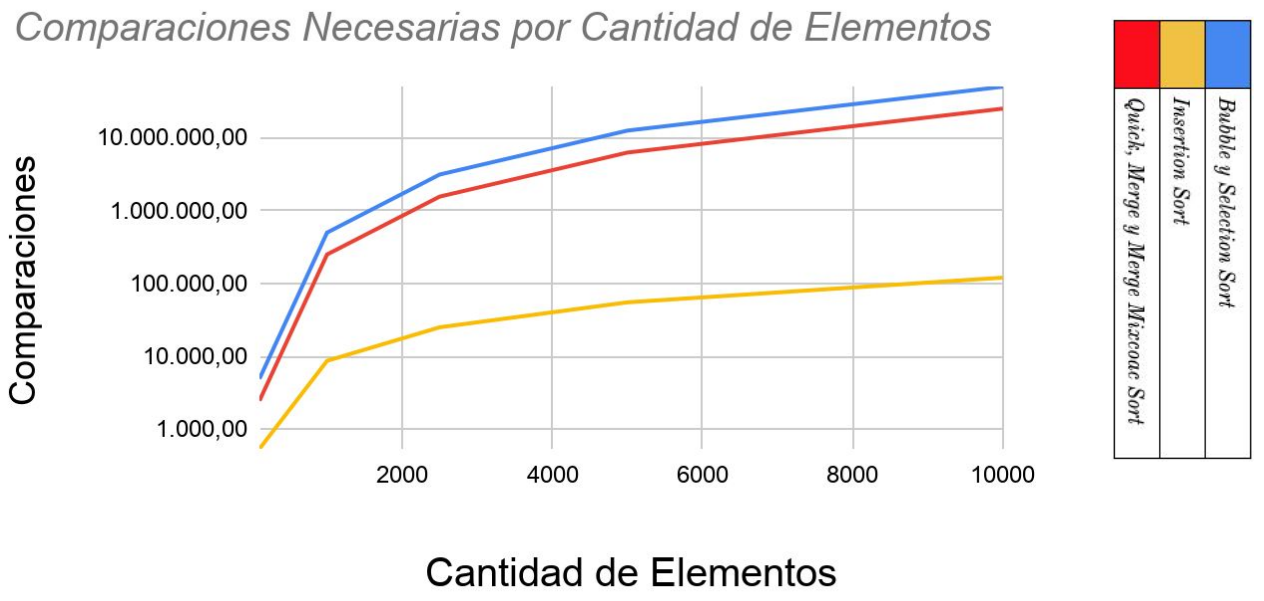


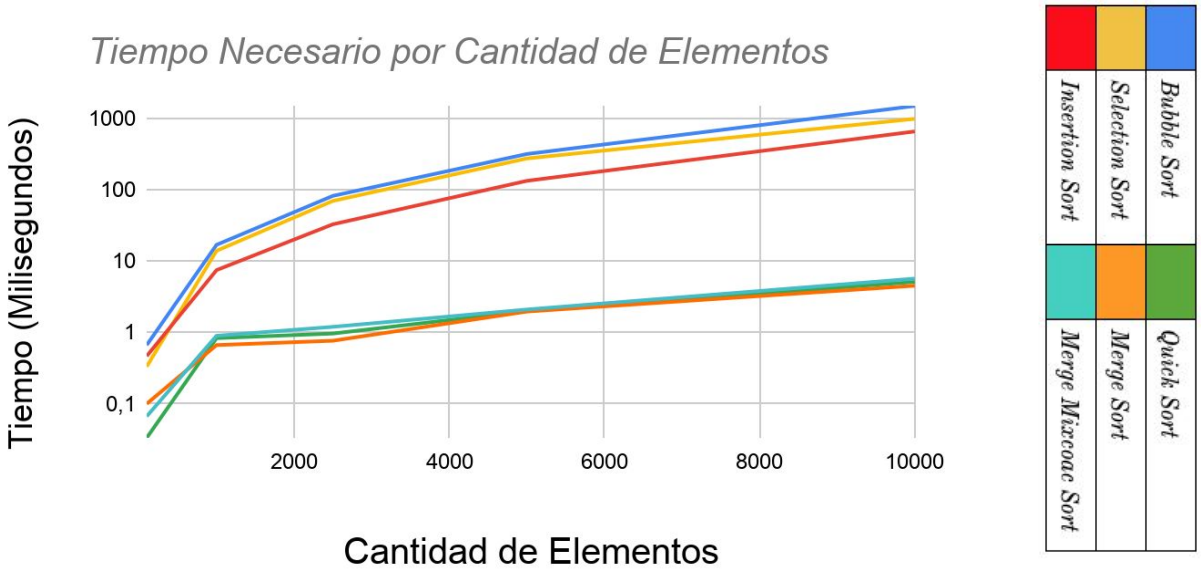
Orden Inverso





Orden Aleatorio

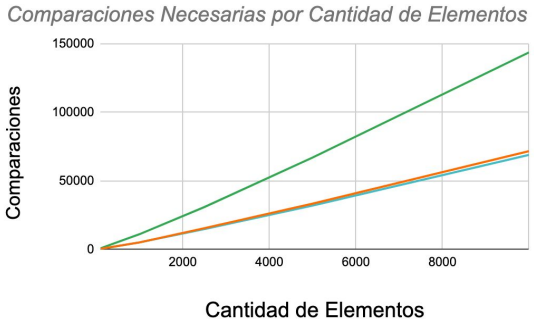




Interpretación

Viendo rápidamente las 6 gráficas podemos notar que algoritmos son los mejores, ya sea tomando en cuenta el número de comparaciones o el tiempo. Los mejores son las dos variaciones de Merge y Quick sort. Bubble Sort y Selection Sort son los peores, pues los dos realizan el mismo mayor número de comparaciones, aunque SelectionSort es más rápido. Insertion Sort es un caso interesante, pues es un poco mejor que Bubble Sort y Selection Sort excepto cuando ya está ordenado el arreglo, caso en el que es mucho mejor que todos los demás algoritmos. Analizaremos posteriormente el porqué.

Debido a las escalas tan grandes utilizadas en las gráficas, no es posible observar muchas diferencias importantes entre el mismo algoritmo al ordenar arreglos ya ordenados, invertidos o aleatoriamente ordenados. La única evidente es que el algoritmo Insertion Sort fue por mucho el mejor con el arreglo ya ordenado, pues lo implementamos de manera que no tuviese que estar comparando repetitivamente si ya estaba ordenado. En este caso, el número de comparaciones que realiza InsertionSort es n -1.



Aunque Quick Sort, Merge Sort y Merge Mixcoac Sort parecen igual de eficientes en las primeras gráficas (si se utiliza una escala no logarítmica), viendo esta gráfica con una mejor escala (pues solo se incluyen estos tres algoritmos) podemos observar que las variaciones de Merge si son casi igual de eficientes, aunque ambas son bastante mejores que Quicksort.

En conclusión, los mejores algoritmos son los que tienen una menor complejidad teórica. Las pequeñas variaciones que encontramos entre algoritmos que teóricamente son igual de complejos, se deben a cambios que realizamos a la hora de implementarlos. Podemos implementar la ventaja que tiene Insertion Sort con arreglos ordenados en los demás algoritmos para hacerlos más eficientes.

Algoritmo	Complejidad
Bubble Sort	O(n^2)
Selection Sort	O(n^2)
Insertion Sort	O(n^2)
Quick Sort	O(n log n).
Merge Sort	O(n log n).

Hoja de cálculo con todos los datos:
https://docs.google.com/spreadsheets/d/1lHAGCv9H9jsIRpIu3ED3ZjQGwNiAqzeIfkl25Ik7HBg/edit?usp=sharing