

Alexandre Aparecido Scrocaro Junior
R.A.: 2135485

a)

Não consegui rodar o de 500000

TEMPO EM SEGUNDOS (VETOR DESORDENADO)

	1000	10000	100000	500000
selection sort	0,003246	0,351476	36,121166	
bubble sort	0,005917	0,614614	69,678238	
insertion sort	0,001963	0,179811	19,003204	
merge sort	0,000351	0,002803	0,032995	
quick sort	0,000219	0,002343	0,031723	

b)

TEMPO EM SEGUNDOS (VETOR DESORDENADO)

	1000	10000	100000	500000
selection sort	0,001589	0,174975	16,173788	
bubble sort	0,002002	0,183582	19,505308	
insertion sort	0,000024	0,000164	0,001567	
merge sort	0,000316	0,002670	0,031668	
quick sort	0,007973	0,830956	85,166603	

c)

i) O algoritmo insertion sort, pois se mostrou o mais rápido de todos nessas circunstâncias;

ii) Usaria o merge sort, pois tem a complexidade mais rápida;

iii) Usaria o algoritmo merge sort, porque foi rápido em qualquer situação;

iv) sim, o algoritmo quick sort se mostrou ineficiente quando se tem um vetor já ordenado (ou quase), mas se tratando de um vetor completamente aleatório, é mais eficiente.