

Relatorio Comparativo de Algoritmos de Ordenacao

Analise Teorica:

Bubble Sort:

- Complexidade de Tempo: $O(n^2)$ para o melhor, medio e pior caso.
- Complexidade de Espaco: $O(1)$.
- Notas: Extremamente ineficiente para grandes volumes de dados.

Merge Sort:

- Complexidade de Tempo: $O(n \log n)$ em todos os casos.
- Complexidade de Espaco: $O(n)$.
- Notas: Eficiente e estavel, mas consome mais memoria.

Quick Sort:

- Complexidade de Tempo: $O(n \log n)$ no caso medio. $O(n^2)$ no pior caso.
- Complexidade de Espaco: $O(\log n)$ a $O(n)$.
- Notas: Um dos mais rapidos na pratica, mas sensivel a entrada no pior caso.

Heap Sort:

- Complexidade de Tempo: $O(n \log n)$ em todos os casos.
- Complexidade de Espaco: $O(1)$.
- Notas: Eficiente, estavel e usa pouca memoria.

Insertion Sort:

- Complexidade de Tempo: $O(n)$ no melhor caso. $O(n^2)$ no caso medio e pior caso.
- Complexidade de Espaco: $O(1)$.
- Notas: Extremamente rapido para dados quase ordenados.

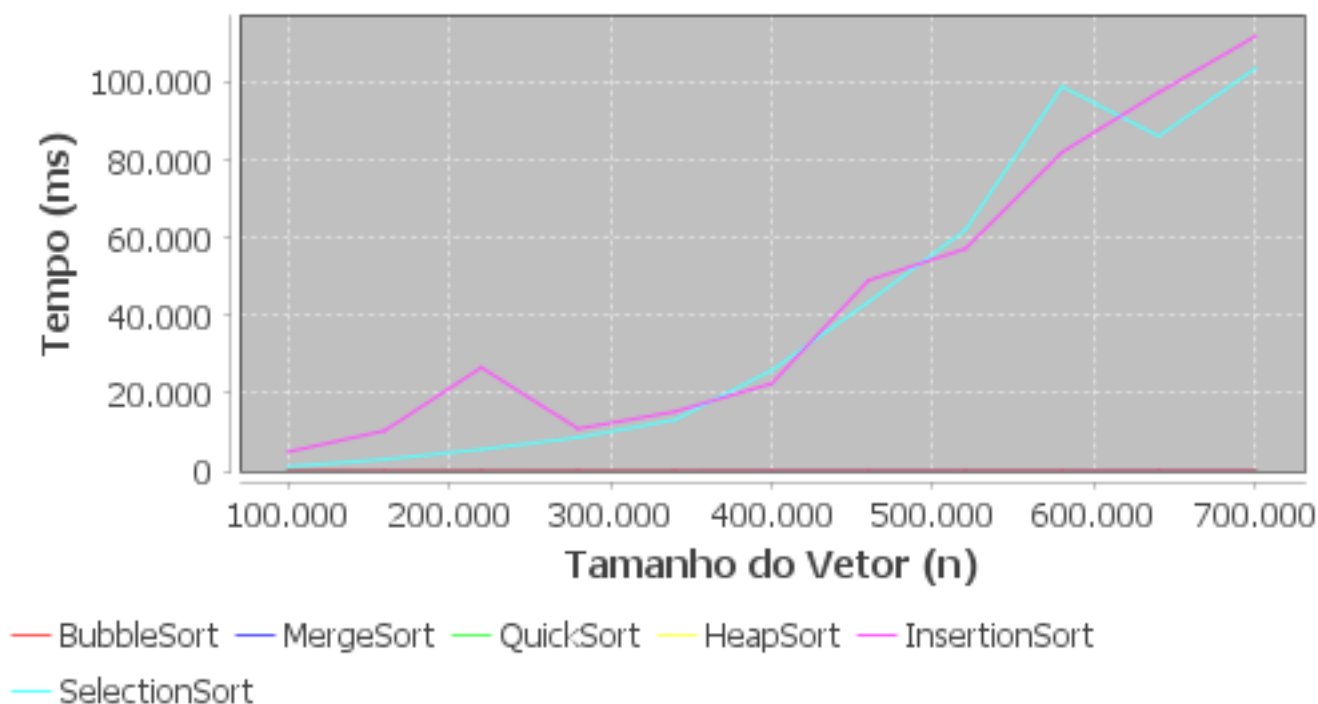
Selection Sort:

- Complexidade de Tempo: $O(n^2)$ em todos os casos.
- Complexidade de Espaco: $O(1)$.
- Notas: Sempre realiza o mesmo numero de comparacoes.

Grafico de Comparacao: Crescente com repeticao

Grafico de Comparacao: Decrescente com repeticao

Tempo de Execucao - Crescente com repeticao



Tempo de Execucao - Decrescente com repeticao

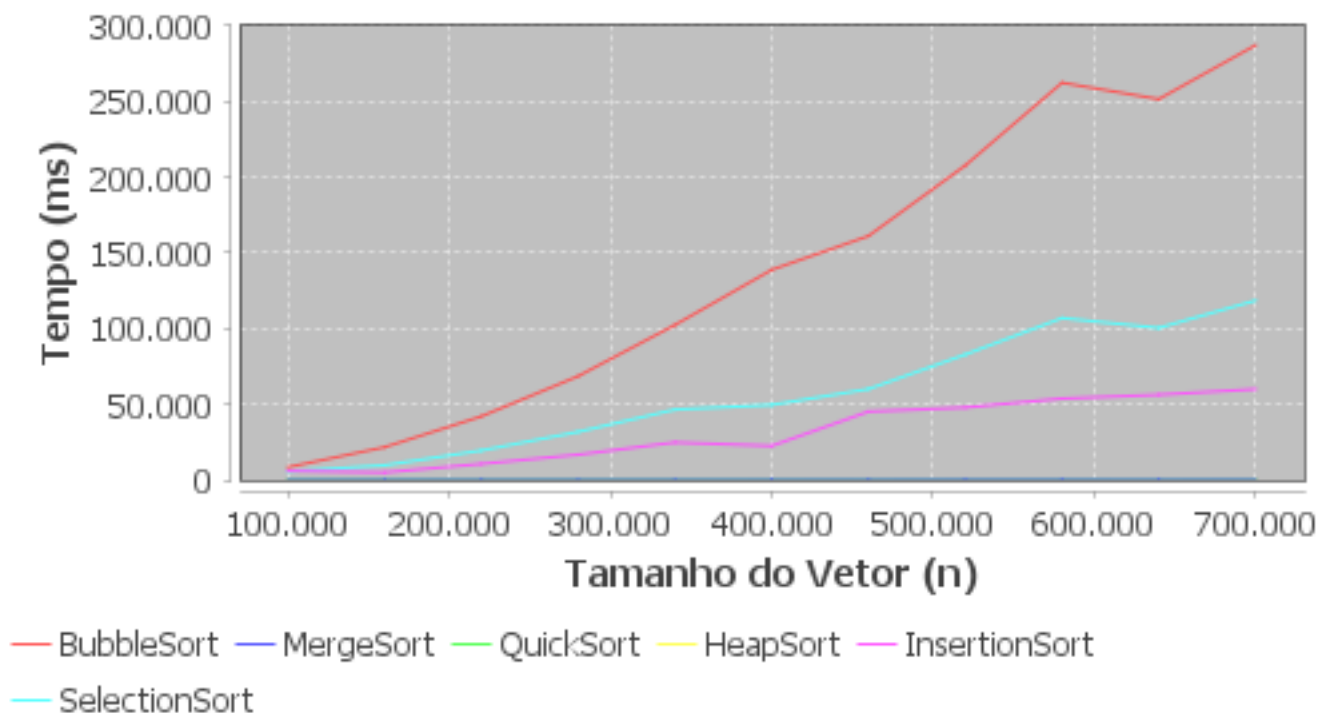
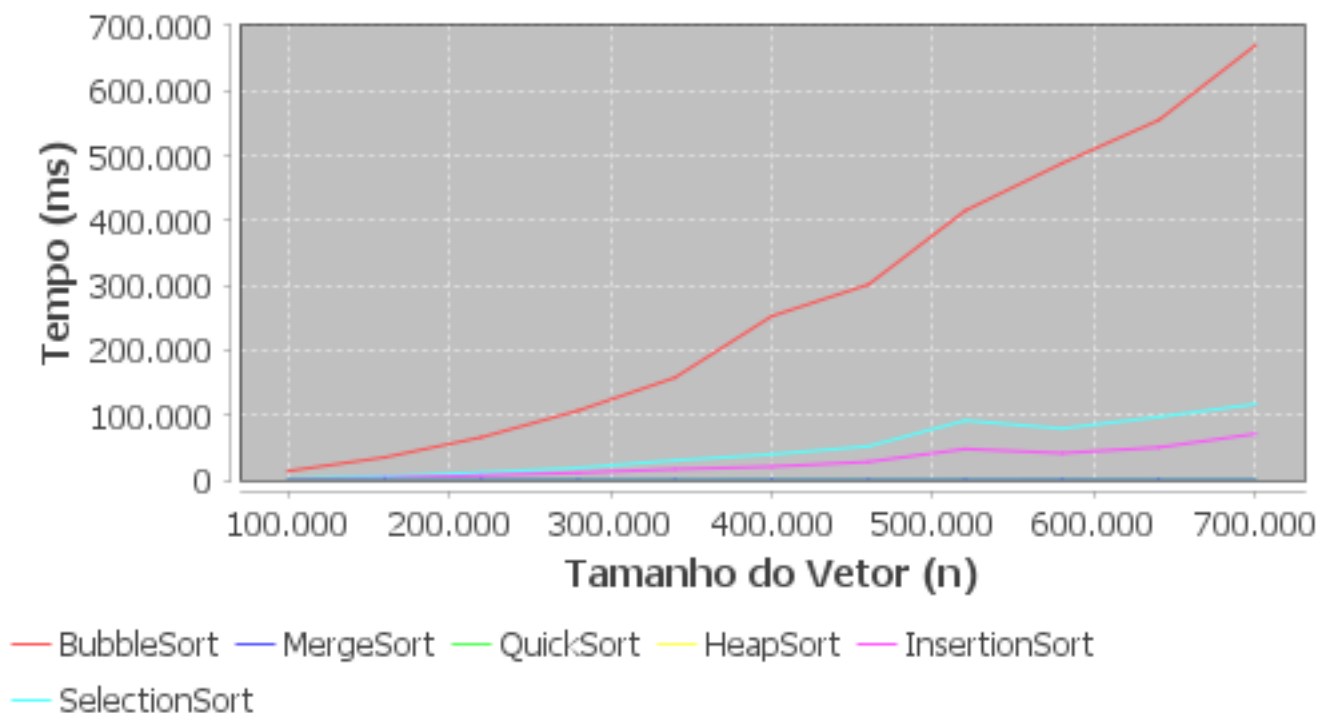


Grafico de Comparacao: Aleatorio com repeticao

Grafico de Comparacao: Crescente sem repeticao

Tempo de Execucao - Aleatorio com repeticao



Tempo de Execucao - Crescente sem repeticao

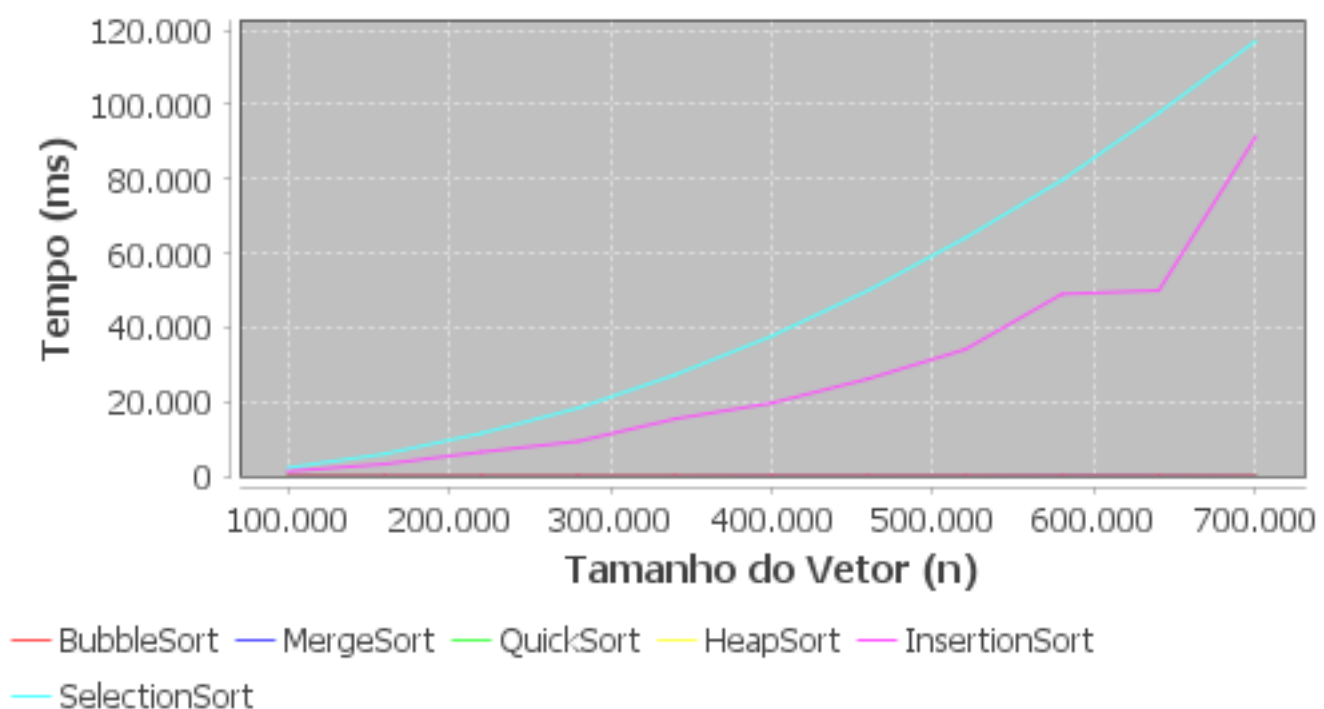
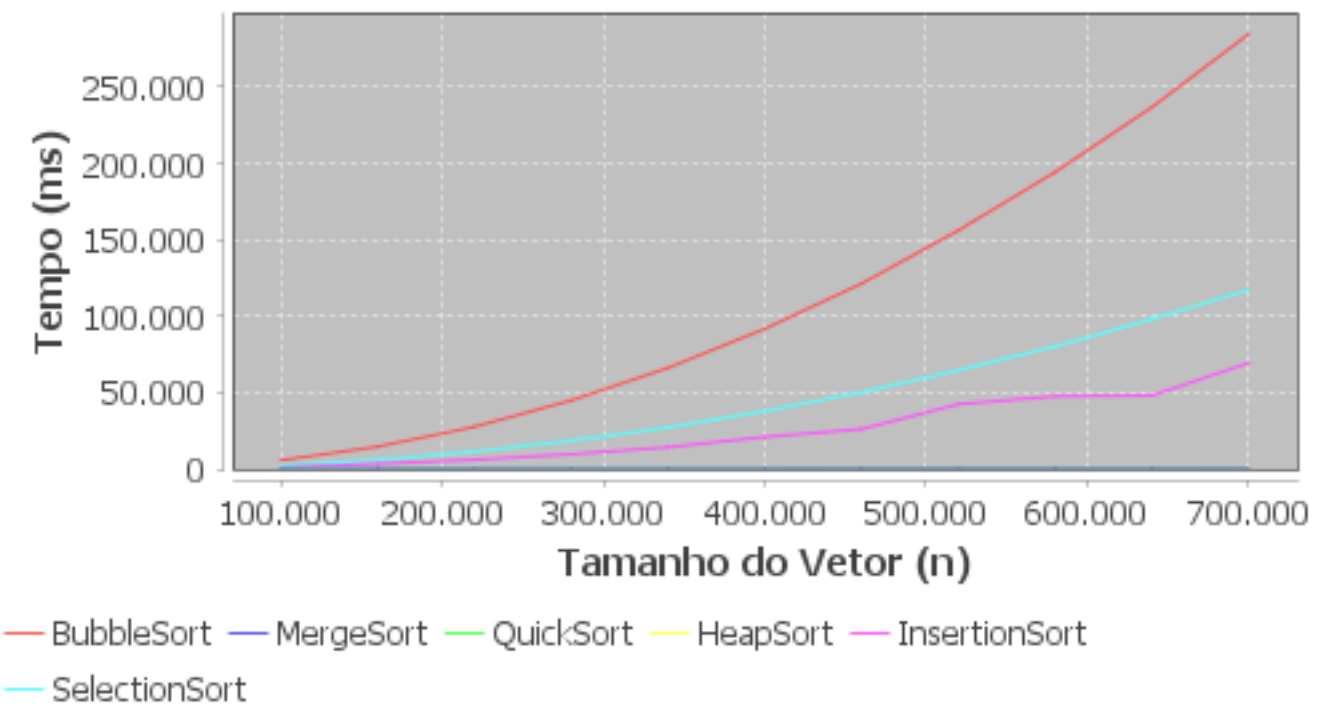


Grafico de Comparacao: Decrescente sem repeticao

Grafico de Comparacao: Aleatorio sem repeticao

Tempo de Execucao - Decrescente sem repeticao



Tempo de Execucao - Aleatorio sem repeticao

