

## Atividade 1

COM112 - Algoritmo e Estrutura de Dados

Data: 01/06/2020

Nome:

Matrícula:

Luana de Cássia Freitas

2019009541

Lucas Lima Lordello

2019015941

### 1. Comparação entre métodos de ordenação interna

#### 1.1. Tempo de Execução

Algoritmo	Aleatório	Ordem Crescente	Ordem Decrescente
Bubble Sort	0.254	0.129	0.225000
Selection Sort	0.108	0.126	0.109000
Insertion Sort	0.072	< 0.001	0.137000
Merge Sort	0.001	0.001	0.001000
Quick Sort	0.001	0.001	< 0.001

#### 1.2. Número de Comparações

Algoritmo	Aleatório	Ordem Crescente	Ordem Decrescente	Complexidade Assintótica
Bubble Sort	49995000	49995000	49995000	$O(n^2)$
Selection Sort	50004999	50004999	50004999	$O(n^2)$
Insertion Sort	24973203	9999	50004999	$O(n)$ ou $O(n^2)$
Merge Sort	382155	271632	331840	$O(n \log n)$
Quick Sort	131241	119535	114548	$O(n \log n)$ ou $O(n^2)$

### 1.3. Número de Movimentações

Algoritmo	Aleatório	Ordem Crescente	Ordem Decrescente
Bubble Sort	24963204	0	49995000
Selection Sort	9987	0	5000
Insertion Sort	24963204	0	49995000
Merge Sort	267232	267232	267232
Quick Sort	104550	17712	32712

### 2. Relação entre comparações e complexidade assintótica

Bubble e Selection possuem complexidade  $O(n^2)$ , então não há variação entre os casos testados. Insertion possui complexidade  $O(n)$  no melhor caso e  $O(n^2)$  no pior caso; é possível observar grande diferença entre seus casos. Merge, um algoritmo  $O(n \log n)$ , mostra uma grande diferença para todos os casos em relação a outros algoritmos. Quick tem um desempenho superior, mas não apresenta estabilidade em raros casos onde pode ser  $O(n^2)$ .

### 3. Código

[https://github.com/LucLordello/COM112\\_2019015941/tree/master/Atividade\\_Avaliativa1](https://github.com/LucLordello/COM112_2019015941/tree/master/Atividade_Avaliativa1)