

BACHARELADO EM ENGENHARIA DE SOFTWARE
ALGORITMO E PROGRAMAÇÃO ORIENTADO OBJETO II

Acadêmicos: Aryslene Santos Bitencourt e Matheus Lima Machado

RELATÓRIO TRABALHO DE ORDENAÇÃO

A seguir será apresentados os dados coletados após a implementação e execução dos seguintes algoritmos:

- Bubble Sort;
- Bubble Sort Otimizado;
- Insertion Sort;
- Insertion Sort Otimizado;
- Selection Sort;
- Merge Sort;
- Quick Sort; e
- Cocktail Sort.

Suas respectivas complexidades, assim como as complexidades dos algoritmos implementados para busca e criação dos vetores, estão ilustrados nas Tabela 1 e Tabela 2:

Tabela 1. Complexidade dos algoritmos de Ordenação			
	Pior caso	Caso médio	Melhor caso
Bubble	$O(n^2)$	$\Theta(n^2)$	$O(n)$
Bubble Otimizado	$O(n^2)$	$O(n^2)$	$\Theta(n \log n)$
Insertion	$O(n^2)$	$O(n^2)$	$O(n)$
Insertion Otimizado	$O(n(\log(n))^2)$	depende da sequência do gap	$O(n \log n)$
Selection	$O(n^2)$	$O(n^2)$	$O(n^2)$
Merge	$\Theta(n \log n)$	$O(n \log n)$	$O(n \log n)$ ou $O(n)$
Quick	$O(n^2)$	$O(n \log n)$	$O(n \log n)$
Cocktail	$O(n^2)$	$O(n^2)$	$O(n)$

Foi realizado a ordenação três vezes para cada tamanho de vetor, já que os tempos gerado em cada execução possui variações, as Tabelas 3 - 8 trazem a média dessas execuções em ordem crescente, sendo assim o primeiro algoritmo apresentado realiza a ordenação de forma mais rápido e o último de forma mais lenta, para cada tamanho de vetor.

Com isso, podemos observar que em todos os casos o algoritmo **Merge Sort** realiza a ordenação com menor tempo, o que já era esperado, pois sua complexidade é $O(n \log n)$.

Assim como é possível notar que o algoritmo **Bubble Sort** realiza a ordenação com maior tempo, pois sua complexidade é **O(n²)**. Sendo assim, o algoritmo Merge Sort é cerca de 34,5% mais rápido que o **Bubble Sort**.

O algoritmo possui complexidade **O(n²)** em seu pior caso, com isso, apesar de ser **O(n log n)** seu pior caso o deixou com o segundo melhor. **Bubble Sort Otimizado** e **Insertion Sort Otimizado** possuem um tempo de ordenação muito próximo. **Selection Sort**, entretanto, já se distancia muito maior dos outros, pois sua complexidade é **O(n²)**. **Cocktail Sort** é também um algoritmo que consome um grande tempo com sua complexidade **O(n²)**. O algoritmo **Insertion Sort** possui maior variabilidade em tempo.

Tabela 3. Ranking da média de tempo de execução para 1000 números				
Algoritmo	1000			MÉDIA (nano)
Merge	23224	49444	23165	23224
Quick	487719	503994	490030	490030
Insertion Otimizado	701461	726358	709417	709417
Bubble Otimizado	871761	844992	917051	871761
Selection	3848275	320759 3	322100 6	3221006
Cocktail	5501056	543626 4	535795 4	5436264
Insertion	6711388	938562 7	671977 1	6719771
Bubble	8202019	818923 5	801130 2	8189235

Tabela 4. Ranking da média de tempo de execução para 5000 números				
Algoritmo	5000			MÉDIA (nano)
Merge	117592	238926	23165	117592
Quick	569527	565740	566656	566656
Bubble Otimizado	2380572	2446243	2398909	2398909
Insertion Otimizado	2773355	2712730	2668383	2712730
Selection	7923687	6769432	10482961	7923687
Insertion	12831944	14954191	15229575	14954191
Cocktail	19292160	19145164	20740470	19292160
Bubble	30059240	27888238	19584128	27888238

Tabela 5. Ranking da média de tempo de execução para 10000 números				
Algoritmo	10000			MÉDIA (nano)
Merge	237145	474563	227816	237145
Quick	1198334	1125014	1100171	1125014
Insertion Otimizado	1766741	1869913	1775687	1775687
Bubble Otimizado	1890768	1710957	1786885	1786885
Selection	17801645	18044791	18064693	18044791
Insertion	27148833	27542446	27396998	27396998
Cocktail	72020734	71023715	72141786	72020734
Bubble	93165454	89401258	94558719	93165454

Tabela 6. Ranking da média de tempo de execução para 20000 números				
Algoritmo	20000			MÉDIA (nano)
Merge	265923	458580	251978	265923
Quick	2144339	2224713	2380094	2224713
Insertion Otimizado	2319554	2274562	2410058	2319554
Bubble Otimizado	4348780	5383719	4744775	4744775
Selection	70055727	70180445	69687837	70055727
Cocktail	319075585	317059479	316426461	317059479
Insertion	110981205	111840974	111475793	111475793
Bubble	461518082	462048440	470718116	462048440

Tabela 7. Ranking da média de tempo de execução para 50000 números				
Algoritmo	50000			MÉDIA (nano)
Merge	626044	610665	633571	626044
Quick	4406855	4672961	4455363	4455363
Insertion Otimizado	5398232	5349708	5386631	5386631
Bubble Otimizado	7363311	4542431	6970267	6970267
Selection	429974615	431927100	430059181	430059181

Insertion	694447403	701116679	690727284	694447403
Cocktail	2054763900	2066820575	2057338971	2057338971
Bubble	3383870971	3458934313	3361139969	3383870971

Tabela 8. Ranking da média de tempo de execução para 100000 números				
Algoritmo	10000			MÉDIA(nano)
Merge	1292081	1269798	1287259	1287259
Bubble Otimizado	8798677	8463001	8406874	8463001
Quick	8927997	8907651	9361256	8927997
Insertion Otimizado	11863847	11853901	11879363	11863847
Selection	1713850227	1732261610	1722303646	1722303646
Insertion	2823349275	2837802554	2834583000	2834583000
Cocktail	8345523115	8337379287	8276504715	8337379287
Bubble	14230502466	14117534035	14114983699	14117534035

Tabela 9. Ranking da quantidade de comparações para 5000 números				
Algoritmo	5000			MÉDIA
Merge	63965	63983	63928	63965
Quick	67089	69019	67063	67089
Insertion Otimizado	73167	69339	72975	72975
Bubble Otimizado	171090	175090	172089	172089
Insertion	6378629	6555710	6492605	6492605
Cocktail	9720440	9880101	9800080	9800080
Bubble	12997000	12997000	12997000	12997000
Selection	12997000	12997000	12997000	12997000

