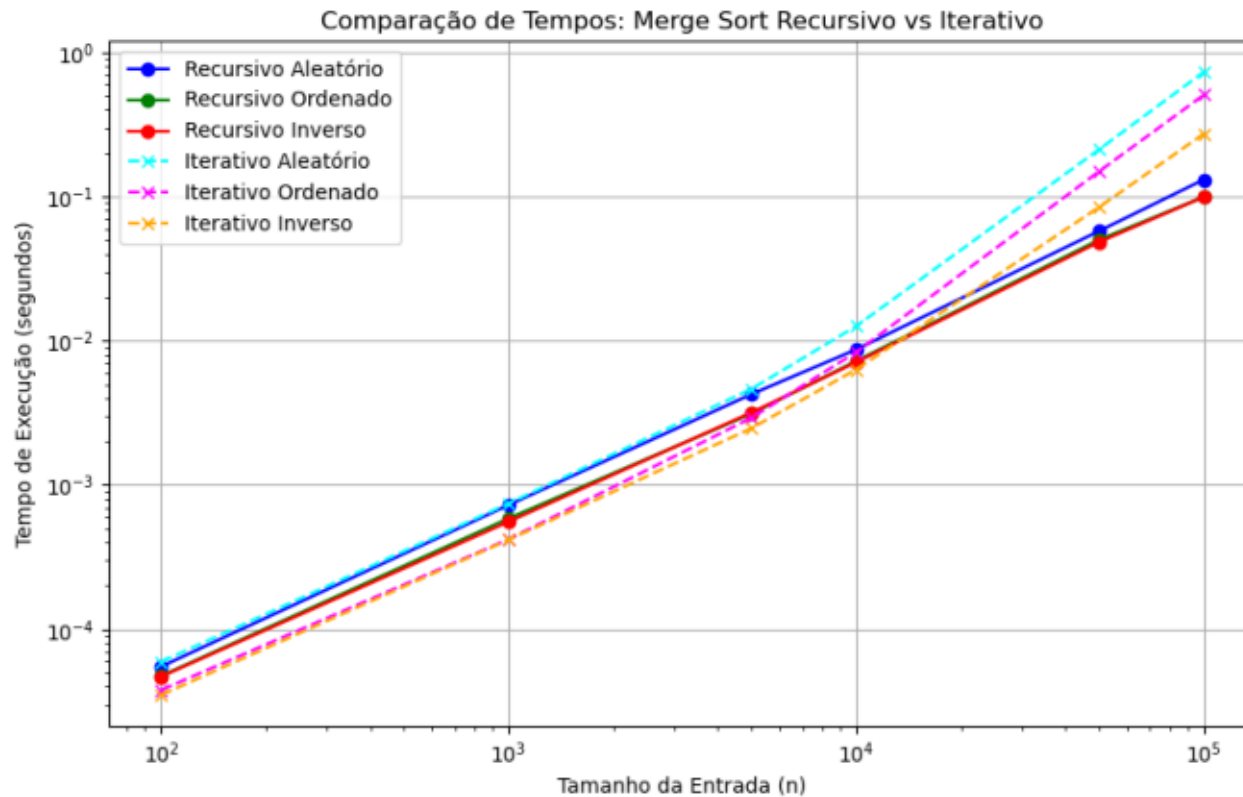


Comparação de Tempos: Merge Sort Recursivo vs Iterativo



Após implementar e testar o algoritmo Merge Sort nas versões recursiva e iterativa, foram observadas as seguintes considerações:

1. Clareza do Código:

- A implementação recursiva é geralmente mais clara e concisa, refletindo a lógica do algoritmo de maneira mais direta. A recursão é mais fácil de entender para aqueles que estão familiarizados com a divisão e conquista.
- A versão iterativa, embora funcional, tende a ser um pouco mais complexa e difícil de seguir, especialmente devido à necessidade de gerenciar manualmente a mesclagem das sublistas.

2. Desempenho:

- Em termos de tempo de execução, ambas as implementações têm complexidade teórica $O(n \log n)$. Contudo, a implementação recursiva pode ser mais lenta em alguns casos devido ao overhead das chamadas de função e à profundidade da pilha, especialmente para n menores.
- A implementação iterativa, por outro lado, tende a ser mais eficiente em termos de uso de memória, pois não utiliza a pilha de chamadas recursivas, o que pode levar a estouro de pilha em listas muito grandes, especialmente para n menores.
- Por outro lado, para n a partir de valores 10^4 , a versão recursiva é mais eficiente devido a otimizações como realizar a mesclagem em sublistas que são geralmente menores e, portanto, mais eficientes. Isso ocorre porque em listas maiores, as vantagens de dividir a lista em sublistas e resolver cada parte de forma independente tornam-se mais evidentes na versão recursiva.

3. Uso de Memória:

- A implementação recursiva pode consumir mais memória, especialmente para entradas grandes, devido à pilha de chamadas. Cada chamada recursiva consome espaço na pilha, o que pode levar a problemas em entradas grandes.
- A versão iterativa, em contrapartida, gerencia a mesclagem dentro de um único bloco de memória, o que a torna mais robusta para listas grandes.

Essas observações sublinham a importância de considerar tanto a clareza do código quanto o desempenho e o uso de memória ao escolher entre abordagens recursivas e iterativas em algoritmos de ordenação.

Resultados de Tempo de Execução - Recursivo

Tamanho da Entrada (n)	Tempo Recursivo Aleatório (s)	Tempo Recursivo Ordenado (s)	Tempo Recursivo Inverso (s)
100.0	5.507469177246094e-05	4.744529724121094e-05	4.696846008300781e-05
1000.0	0.0007276535034179688	0.0005872249603271484	0.0005574226379394531
5000.0	0.004269838333129883	0.0031273365020751953	0.0031867027282714844
10000.0	0.008709192276000977	0.007249593734741211	0.007110595703125
50000.0	0.05776357650756836	0.05020761489868164	0.04845285415649414
100000.0	0.13126707077026367	0.09947371482849121	0.09949350357055664

Resultados de Tempo de Execução - Iterativo

Tamanho da Entrada (n)	Tempo Iterativo Aleatório (s)	Tempo Iterativo Ordenado (s)	Tempo Iterativo Inverso (s)
100.0	5.936622619628906e-05	3.790855407714844e-05	3.504753112792969e-05
1000.0	0.0007462501525878906	0.0004210472106933594	0.0004177093505859375
5000.0	0.004641532897949219	0.0029299259185791016	0.0024797916412353516
10000.0	0.012639760971069336	0.008360147476196289	0.006334781646728516
50000.0	0.21261191368103027	0.14975833892822266	0.08457231521606445
100000.0	0.7313745021820068	0.5069615840911865	0.2727048397064209