

Suponha que estamos comparando implementações de ordenação por inserção e ordenação por intercalação na mesma máquina. Para entradas de tamanho n , a ordenação por inserção é executada em $8n^2$ passos, enquanto a ordenação por intercalação é executada em $64n \lg n$ passos. Para quais valores de n a ordenação por inserção supera a ordenação por intercalação?

Para determinar os valores de n para os quais a ordenação por inserção é mais eficiente, precisamos resolver a seguinte inigualdade:

$$8n^2 < 64n \lg n$$

Podemos simplificar essa inequidade dividindo ambos os lados por $8n$ (considerando que $n > 0$):

$$n < 8 \lg n$$

Agora, para encontrar o valor exato de n , podemos usar métodos numéricos ou aproximar a solução. Uma forma simples de estimar é testar valores inteiros e observar quando a inequação passa de verdadeiro para falso.

Vamos testar alguns valores:

- Para $n = 16$: $16 < 8 \lg 16 \implies 16 < 8 \times 4 \implies 16 < 32$ (Verdadeiro)
- Para $n = 32$: $32 < 8 \lg 32 \implies 32 < 8 \times 5 \implies 32 < 40$ (Verdadeiro)
- Para $n = 64$: $64 < 8 \lg 64 \implies 64 < 8 \times 6 \implies 64 < 48$ (Falso)

Portanto, a ordenação por inserção é mais eficiente que a ordenação por intercalação para $n = 16$ e $n = 32$, mas não para $n = 64$.