Análise de Tempo de Execução de Algoritmos de Busca

Para fazer a análise de tempo dos algoritmos anexados, vamos avaliar o comportamento de cada um em termos de número de passos básicos, em relação ao tamanho da entrada n.

# 1) Busca Linear

Pior Caso: O algoritmo percorre todos os elementos de A. Se x não pertence a A, ele precisa realizar n comparações. Logo, o tempo de execução é O(n).  
Melhor Caso: Se x = A[1], apenas uma comparação é necessária. Portanto, o tempo de execução é O(1).  
Caso Médio: A busca encontra x em média após n/2 comparações. Assim, o tempo esperado é O(n).

# 2) Busca Linear em Ordem

Pior Caso: Caso x seja maior que todos os elementos, o algoritmo fará n comparações. Logo, o tempo é O(n).  
Melhor Caso: Se x = A[1], o algoritmo retorna após uma comparação, então O(1).  
Caso Médio: Se os elementos estiverem em ordem crescente e x está presente, em média são feitas n/2 comparações. Portanto, o tempo é O(n).

# 3) Busca Binária

Pior Caso: O algoritmo divide o intervalo em duas partes a cada passo, o que resulta em um número de comparações proporcional a log\_2(n). Assim, o tempo de execução é O(log n).  
Melhor Caso: Se x estiver no meio da lista já na primeira comparação, o tempo será O(1).  
Caso Médio: Em média, a busca encontrará x após O(log n) comparações.

# Tabela Comparativa

|  |  |  |  |
| --- | --- | --- | --- |
| Situação | Busca Linear | Busca Linear em Ordem | Busca Binária |
| x ∈ A | O(n) | O(n) | O(log n) |
| x = A[1] | O(1) | O(1) | O(1) |
| x = A[n] | O(n) | O(n) | O(log n) |
| x ∉ A | O(n) | O(n) | O(log n) |