



Lista de Exercícios 08 (Pesquisa)

1. Em que tipo de lista a busca binária é mais eficiente? a) Listas desordenadas
b) Listas ordenadas
c) Listas de tamanho fixo
d) Listas encadeadas
2. Qual é a complexidade de tempo da busca binária? a) $O(n)$
b) $O(\log)$
n) c) $O(1)$
d) $O(n^2)$
3. Qual é a principal desvantagem da busca binária? a) É difícil de implementar em Python.
b) Só funciona com números inteiros.
c) A lista precisa estar desordenada para funcionar corretamente.
d) A lista precisa estar ordenada para funcionar corretamente.
4. Ao utilizar a pesquisa binária, qual condição leva ao término da recursão na busca? A) Quando o índice de início é menor que o índice final.
B) Quando o índice de meio é menor que o índice final.
C) Quando o índice de início é maior que o índice final.
D) Quando o índice do elemento buscado é encontrado.
5. Dada uma lista desordenada `[10, 22, 5, 1, 7]`, qual o número de comparações feitas na pesquisa sequencial para encontrar o elemento `5`?
6. Após quantas iterações a pesquisa binária encontra o elemento `8` em `[2, 4, 6, 8, 10]`?
7. Em uma lista ordenada de 1000 elementos, a pesquisa binária levará em média quantas comparações no pior caso?
8. Em uma pesquisa sequencial, quantas iterações no máximo são necessárias para encontrar um elemento em uma lista de 100 elementos?
9. Suponha que você tenha uma lista de strings ordenadas alfabeticamente. Como você adaptaria a busca binária para encontrar uma determinada string nessa lista? Forneça um exemplo em Python.
10. Liste duas vantagens e duas desvantagens da busca sequencial em relação à busca binária. Explique por que essas características são importantes.
11. Apresente uma versão recursiva da busca binária em Python. Explique como a recursão funciona nesse caso.
12. Crie uma função de pesquisa binária que retorne todos os índices de um número em uma lista ordenada que pode conter duplicatas.
13. Implemente a busca binária usando recursividade sem o operador slice. Você precisará passar o índice dos valores iniciais e finais na sublistas.
14. Gere uma lista de números aleatórios, ordene-os e verifique o desempenho dos algoritmos de busca sequencial binária com slide, binária sem slide.