

made for free at coggle.it

BUSCA BINÁRIA: algoritmo eficiente de busca que requer uma lista ordenada. Ele reduz o espaço de busca pela metade a cada iteração.

Começando no meio da lista ordenada, o algoritmo compara o elemento central com o valor desejado. Com base nessa comparação, metade da lista é descartada e a busca continua na metade restante.

Extremamente eficiente para listas ordenadas e é amplamente utilizada em algoritmos de busca em estruturas de dados como árvores binárias de busca.

BUSCA SEQUENCIAL: algoritmo simples de busca que percorre uma lista elemento por elemento até encontrar o valor desejado.

Começando do **primeiro elemento**, cada elemento da lista é comparado com o valor procurado até que seja encontrado ou até o final da lista ser alcançado.

Busca Sequencial e Busca Binária

Eficiência e complexidade: complexidade de tempo de O(log n), onde n é o tamanho da lista. Isso significa que a eficiência aumenta exponencialmente com o tamanho da lista.

Eficiência e complexidade: complexidade de tempo de O(n), onde n é o tamanho da lista. Isso significa que o tempo de execução aumenta linearmente com o tamanho da lista.

É eficaz para **listas não ordenadas ou quando não há garantia sobre a ordenação dos elementos**. No entanto, pode ser ineficiente para grandes conjuntos de dados.