**Aula 03 – Busca Sequencial em Vetores Não Ordenados**

**Atividade com IA – Chat GPT**

**Quais os passos para encontrar um elemento em uma lista (vetor) não ordenado?**

R: Um **vetor não ordenado** é uma estrutura de dados (lista/array) em que **os elementos estão armazenados sem seguir uma ordem específica** — ou seja**, não estão organizados de forma crescente nem decrescente.**

**O que caracteriza um vetor não ordenado?**

Características principais de um vetor não ordenado:

1. A ordem dos elementos é aleatória (ou baseada na ordem de inserção).
2. Não há garantia de que um valor maior ou menor estará em uma posição específica.
3. Busca eficiente por ordem (como a binária) não é possível — por isso, usamos busca sequencial.
4. Pode ser mais rápido para inserir elementos, já que não há necessidade de manter a ordem.

**Quais são os passos necessários para encontrar um valor dentro desse vetor?**

A busca começa no primeiro elemento.

Ela compara cada elemento com o valor buscado.

Quando encontra o elemento igual ao valor procurado, ela imediatamente interrompe o processo e retorna à posição desse elemento.

Ou seja, você sabe exatamente onde está o valor dentro do vetor.

**O que acontece se o elemento não estiver presente?**

**R:** Quando o elemento **não está presente** no vetor, a busca sequencial percorre **todo o vetor** e, ao final, não encontra nenhum elemento igual ao valor buscado.

### O que acontece na prática?

* A busca começa do início e compara cada elemento com o valor buscado.
* Passa por **todos os elementos** sem encontrar nenhuma correspondência.
* Ao chegar no **último elemento** e ainda não encontrar, a busca termina.
* Retorna um valor especial, geralmente -1, para indicar que o elemento **não está no vetor**.

**Qual é a eficiência (tempo de execução) desse processo de busca?**

**R:** Tempo de execução depende de quantos elementos o algoritmo precisa analisar para encontrar o valor. Se o vetor for muito grande, e você precisar fazer muitas buscas, a busca sequencial pode ser lenta. Para vetores **ordenados**, a busca binária, que é O(log n), costuma ser mais eficiente.