**O que são Ponteiros**

* Ponteiros são variáveis que **armazenam o endereço de memória** de outra variável.
* Exemplo:
* int x = 10; // variável normal
* int \*p = &x; // ponteiro que armazena o endereço de x
* Analogia: memória = fileira de caixas; ponteiro = anotação com o número da caixa.

**Por que usar ponteiros?**

* Manipular dados de forma eficiente.
* Necessários para:
  + Arrays
  + Strings
  + Alocação dinâmica de memória
  + Passagem por referência em funções

**Passagem de Argumentos**

Existem duas formas:

**1. Passagem por Valor**

* É criada **uma cópia** da variável.
* Alterações feitas na função **não afetam** a original.
* Exemplo:
* int CuboPorValor(int n) {
* return n\*n\*n;
* }

**2. Passagem por Referência**

* É passado o **endereço da variável** para a função.
* Feito com **ponteiros**.
* Alterações na função **afetam a variável original**.
* Exemplo:
* void cuboPorReferencia(int \*nPtr) {
* \*nPtr = (\*nPtr) \* (\*nPtr) \* (\*nPtr);
* }

**Diferença**

* **Por valor** → cópia → não altera a original.
* **Por referência** → endereço → altera a original.

**✅ Exemplo prático com vetor**

* Quando trabalhamos com **vetores**, eles são sempre passados por referência (na prática, o nome do vetor é um ponteiro).
* Se quisermos dobrar cada elemento:
* void dobrarValores(int vetor[], int tamanho) {
* for (int i = 0; i < tamanho; i++) {
* vetor[i] = vetor[i] \* 2; // altera o vetor original
* }
* }
* Aqui, usamos **passagem por referência implicitamente**, porque vetor é um ponteiro para o primeiro elemento.