## VETORES E MATRIZES

# **VETORES**

#### **RELEMBRANDO**

- O que são variáveis?
  - o Locais da memória que armazenam um valor para ser acessado posteriormente.
- E o que acontece quando precisamos de mais locais de memória para serem acessados?
  - o Declaramos mais variáveis
- Isso é prático?Dinâmico?
  - o Não...

## SITUAÇÃO PROBLEMA

- Crie um programa que receba um conjunto de 5 notas de um aluno, e ao final compute a média do aluno, e informe se o mesmo foi aprovado, ficou em exame ou reprovou em uma disciplina. Posteriormente a isso, considere uma turma com 15 alunos, compute a média de cada aluno, e a mediana da turma, bem como o diário de classe, contendo as notas de cada aluno.
  - o Cerca de 75 variáveis;

• Uma outra forma de resolver isso é utilizar vetores e matrizes

#### ESTRUTURAS DE DADOS

- Estruturas de dados são formas de representar dados do mundo real no mundo computacional
- A grosso modo, uma variável é uma estrutura de dado.
- Assim, vetores e matrizes também são estruturas de dados, que armazenam dados na memória do computador, mas de uma forma sequencial.

#### **VETORES**

- Também chamados de **arrays**, representam valores deforma unidimensional, isto é, somente em uma dimensão
  - o Organiza os dados em forma de linha ou coluna
  - o Cada elemento é armazenado em uma posição do vetor
  - o A posição geralmente começa em 0.
- A depender da linguagem, é permitido apenas um tipo de dado por vetor
  - o No caso do PHP, não importa muito. Cada posição é tratada de forma independente.

#### **VETORES**

- Vetores podem ser declarados de duas formas no PHP
  - o Utilizando os colchetes após o nome de uma variável quando for utilizar
    - \$vetor[0] = 0 /\*criou um vetor\*/
    - \$vetor = array(); /\*função nativa do PHP que cria um vetor\*/
- E agora, como manipular os elementos do vetor

#### PERCORRENDO O VETOR

- Para que possamos manipular um vetor, devemos poder acessar os elementos, e como fazemos isso?
  - o De forma dinâmica, preferencialmente?
- Utilizamos laços de repetição
  - o while, do ... while, for.
  - o Utiliza-se mais o for, pois geralmente sabe-se o tamanho dos vetores

#### ATRIBUINDO VALORES

- Assim, de forma dinâmica podemos acessar os elementos do vetor
  - o Utilizamos um indicador/variável de controle para indexar o vetor
- Podemos então atribuir valores.
  - Os valores são atribuídos aos elementos, não ao vetor em si.
  - o Atribui-se da mesma forma que a uma variável
  - o Pode ser atribuído qualquer valor (números, texto, booleanos)

#### ACESSANDO VALORES

- Para acessar os valores, o procedimento é o mesmo
  - o Utiliza-se um laço de repetição para percorrer o vetor
  - o Utiliza-se uma variável de controle para modificar o índice de acesso
  - o Tem-se acesso ao elemento.
- E se eu acessar um índice que não existe
  - o Meu vetor tem 10 posições, e eu quero acessar a posição 11.
  - o O que acontece??

#### **ACESSANDO VALORES**

- Depende.
  - o O programa pode ser encerrado de forma sumária
  - o Pode ser feito apenas um aviso
- Como prevenir isso...
  - o Podemos utilizar constantes, definindo os tamanhos.
  - o Ou então
  - o Utilizar a instrução foreach

#### **FOREACH**

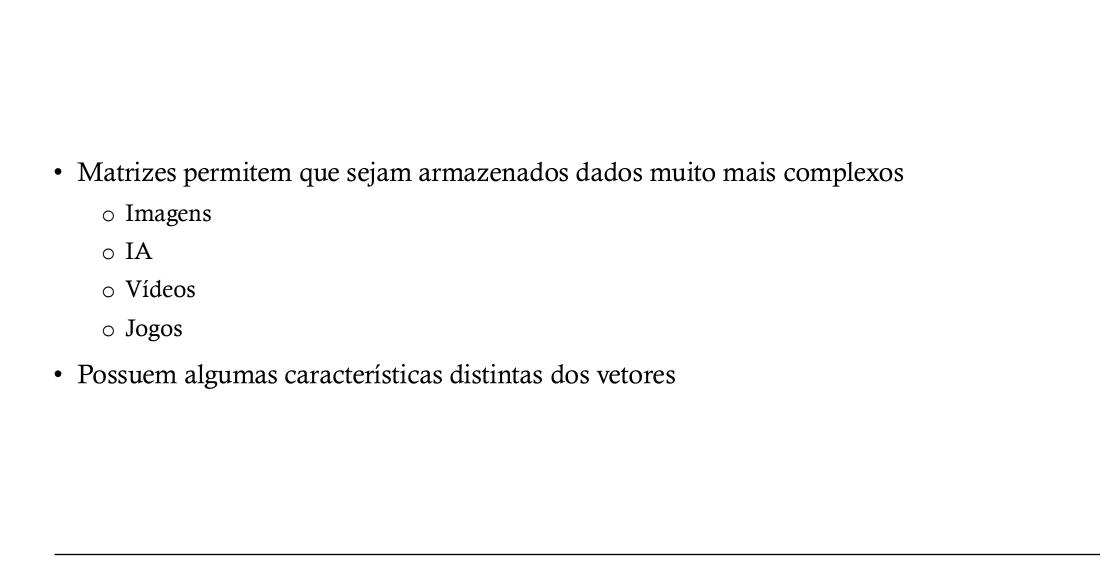
- Uma opção para realizar a leitura e manipulação de vetores
- Permite executar instruções para cada elemento do vetor, evitando índices inválidos
- Possui duas sintaxes
  - Foreach (\$vetor as \$valor) /\*Cria uma variável para uso interno, representando o elemento da posição\*/
  - Foreach (\$vetor as \$chave => \$valor) /\*Cria um par (chave,valor) com o índice e o valor do elemento para uso interno\*/

### **EXERCÍCIOS**

- Acesse o link
- Lista de Exercícios 05 -.pdf
- Escolha 5 exercícios sobre vetores e os implemente

# MATRIZES

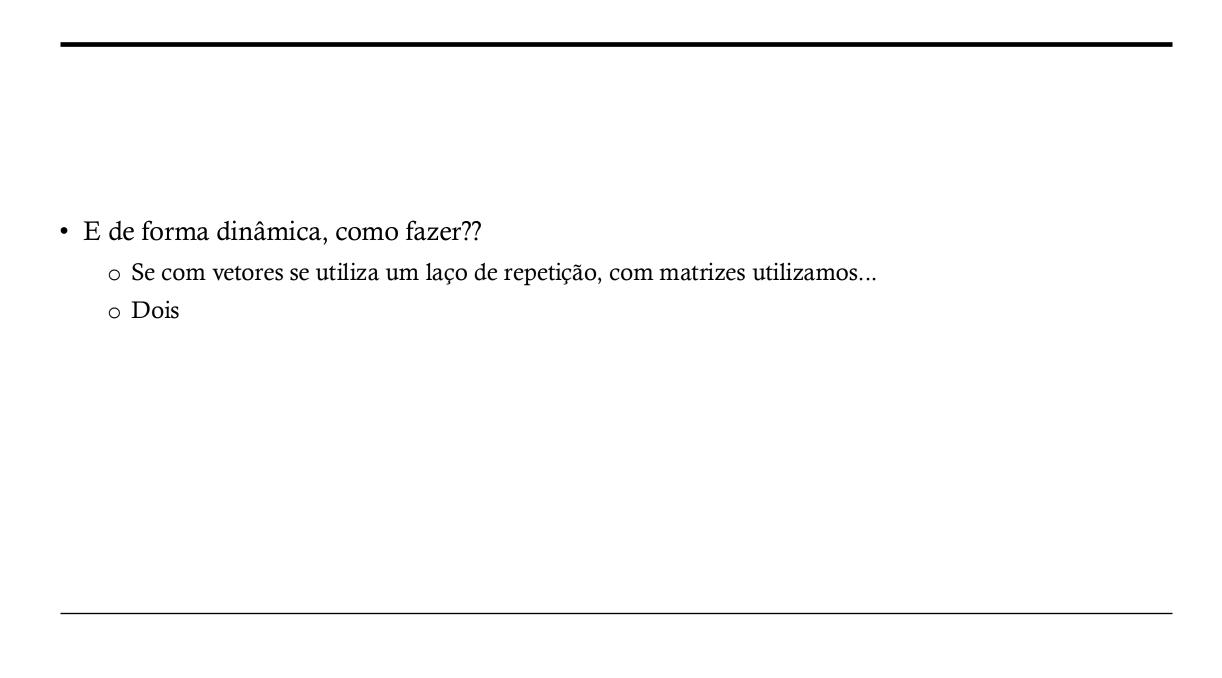
• De forma similar aos vetores, matrizes também são uma forma de estrutura de dados
<ul> <li>Porém representam dos dados de forma bidimensional</li> <li>Linhas e colunas</li> </ul>
• Qual a vantagem sobre os vetores?



- Dimensionalidade
  - o Não está limitada a apenas duas dimensões
- Ordem
  - o Tamanho
  - $\circ$  2x2, 3x4, 5x7
- Tipos
  - o Quadrada, identidade, simétrica..

- As matrizes são usualmente bidimensionais, então, como eu acesso os dados?
  - o O acesso pode ser direto, similar ao vetor (vetor[1])
  - o matriz[0][1] Acessa a posição 1 da matriz
- E para declarar uma matriz?
  - o Em PHP, de forma similar ao vetor, utilizamos

```
$matriz[0][0] = "valor";
```



#### **EXEMPLO**

Crie um sistema simples de notas escolares usando matrizes. O sistema armazenará 4 notas de 3 alunos, calculará a média de cada aluno e exibirá os resultados.

#### EXEMPLO 2

Crie um programa que armazene as temperaturas registradas em uma cidade durante uma semana (7 dias), em três períodos do dia (manhã, tarde e noite). O programa deve:

Ler as temperaturas e armazená-las em uma matriz Calcular a média de temperatura para cada dia da semana.

Determinar o dia mais quente da semana (considerando a maior média diária).

Exibir a matriz completa, as médias diárias e o dia mais quente.

### **EXERCÍCIOS**

- Acesse o link
- Lista de Exercícios 05 -.pdf
- Escolha 5 exercícios sobre matrizes e os implemente.

#### TRABALHO FINAL

- Em grupos de no máximo 3 integrantes, escolha um dos exercícios da lista abaixo.
- Implementem o solicitado, utilizando os conhecimentos adquiridos ao longo do curso
- Utilizem um sistema de controle de versão para rastrear mudanças e colaborar no desenvolvimento

#### TRABALHO FINAL - ASSUNTOS

- Link para os assuntos do trabalho
  - o Avaliação Final.pdf
- Os assuntos já escolhidos estarão disponíveis na planilha abaixo
  - o Assuntos escolhidos.xlsx

#### TRABALHO FINAL - ENTREGAS

- As entregas do trabalho serão em duas etapas.
- A primeira deve ser via software de controle de versão.
  - o Recomendo github
  - o Data limite: 09/06/2025
- A segunda será um pitch, similar ao modelo do projeto integrador e do Hackaton
  - o Data: 11/06/2025
  - o Tempo de apresentação 5 minutos