

Bacharelado em Sistemas de Informação

Algoritmos - Aula 12

Prof. Dr. Sérgio Luis Antonello

Prof. Me. Antero Sewaybricker Todesco

Bibliografia básica desta aula

Medina & Fertig (2006). Capítulo 07

Forbellone & Eberspacher (2000). Capítulo 04

Ascencio & Campos (2002). Capítulo 05

26/05/2020

Plano de Ensino: agenda

15	19/05/2020	Variáveis indexadas: declaração e manipulação de vetor.
16	26/05/2020	Variáveis indexadas: declaração e manipulação de matriz.
17	02/06/2020	Semana Científica do Curso.
18	09/06/2020	AVALIAÇÃO: Prova 2.
19	16/06/2020	AVALIAÇÃO: SUB Devolutiva da Prova 2. TRABALHO: Entrega do trabalho A2.
20	23/06/2020	Devolutiva da SUB.
21	30/06/2020	Fechamento do semestre.

OBS: Trabalho em grupo

- Tem alguém sem grupo?
- Deadline 16/06/2020 (19h)

Plano de ensino: avaliação

$$\text{Nota Final} = (A1 + (2 \times A2)) / 3$$

Composição da nota A2

- 70% por uma prova individual (P2) +
- 10% Atividades práticas (tarefas e atividades) +
- 20% Atividade prática em grupo (trabalho A2)

Plano de ensino: conteúdo

1. Unidade I – Introdução a algoritmos (objetivos a, b, c)
 - 1.1. Conceitos de abstração de dados
 - 1.2. Lógica de programação
 - 1.3. Algoritmos
 - 1.4. Formas de representação de algoritmos: pseudocódigo e fluxograma.
 - 1.5. Teste de mesa
 - 1.6. Tipos de dados
 - 1.7. Constantes e variáveis
 - 1.8. Atribuição
 - 1.9. Operadores e precedência
 - 1.10. Expressões aritméticas, relacionais e lógicas.
2. Unidade II – Estruturas básicas de controle (objetivos c, d, e)
 - 2.1. Blocos de comando
 - 2.2. Estruturas de decisão
 - 2.3. Estruturas de repetição
 - 2.4. Aninhamento
3. Unidade III – Modularização (objetivos c, d, e)
 - 3.1. Dividir para conquistar
 - 3.2. Procedimentos e funções
 - 3.3. Escopo de variáveis
 - 3.4. Parâmetros e argumentos
 - 3.5. Passagem de parâmetros por valor e por referência
4. Unidade IV – Estruturas de dados homogêneas (objetivos d, e)
 - 4.1. Vetor
 - 4.2. Matriz

Sumário

Primeiro momento: revisão

- ✓ Vetor
- ✓ Correção de exercícios

Segundo momento

- ✓ Variáveis indexadas
 - ✓ Vetor
 - ✓ Matriz

Terceiro momento: síntese

1. Primeiro momento: vetor

- ❑ O que é um vetor?
- ❑ Como é feito o acesso de um elemento do vetor?
- ❑ Para acessar todos os elementos de um vetor, que estrutura usamos?

1. Primeiro momento: vetor

- ❑ O que é um vetor?
- ❑ Como é feito o acesso de um elemento do vetor?
- ❑ Para acessar todos os elementos de um vetor, que estrutura usamos?

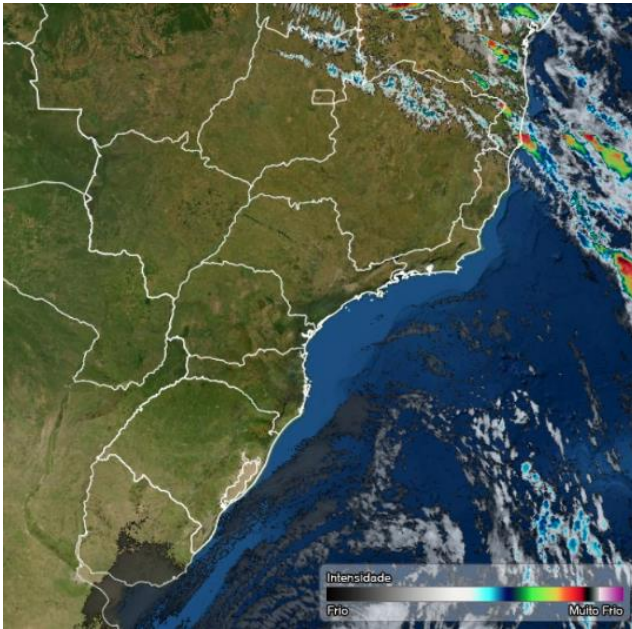
Nome da variável	Nota				
Índice (posição)	1	2	3	4	5
Conteúdo	8.5	2.0	9.5	10.0	6.5

```
para ind de 1 ate 5 passo 1 faca  
    leia (Nota[ind])  
fimpara
```

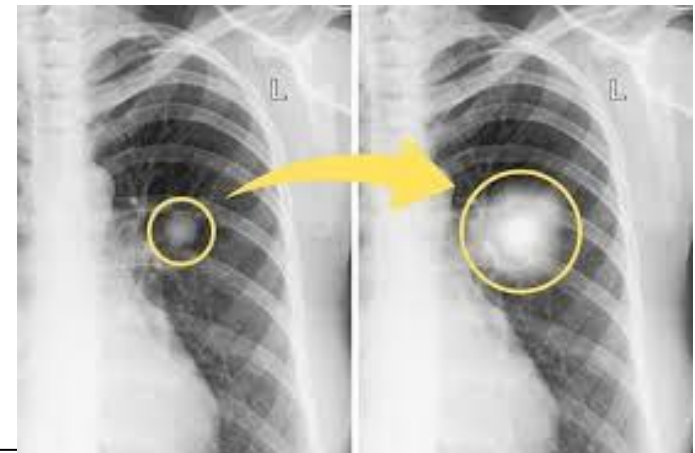
1. Primeiro momento

Correção de exercícios

2. Segundo momento: Motivação



100, 2, 1, 3, 100



41

42

43

44

45

46

47

48

49

50

2. Segundo momento

❑ Variáveis indexadas

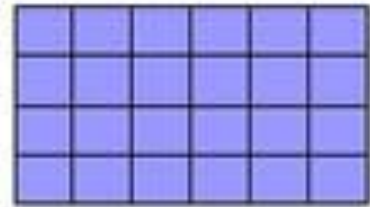
❑ Vetor

❑ Matriz

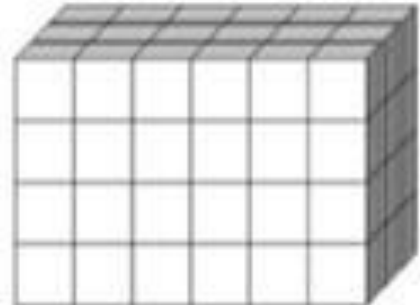
■ Vetor



■ Matriz 2D



■ Matriz 3D



3. Variável indexada: matriz

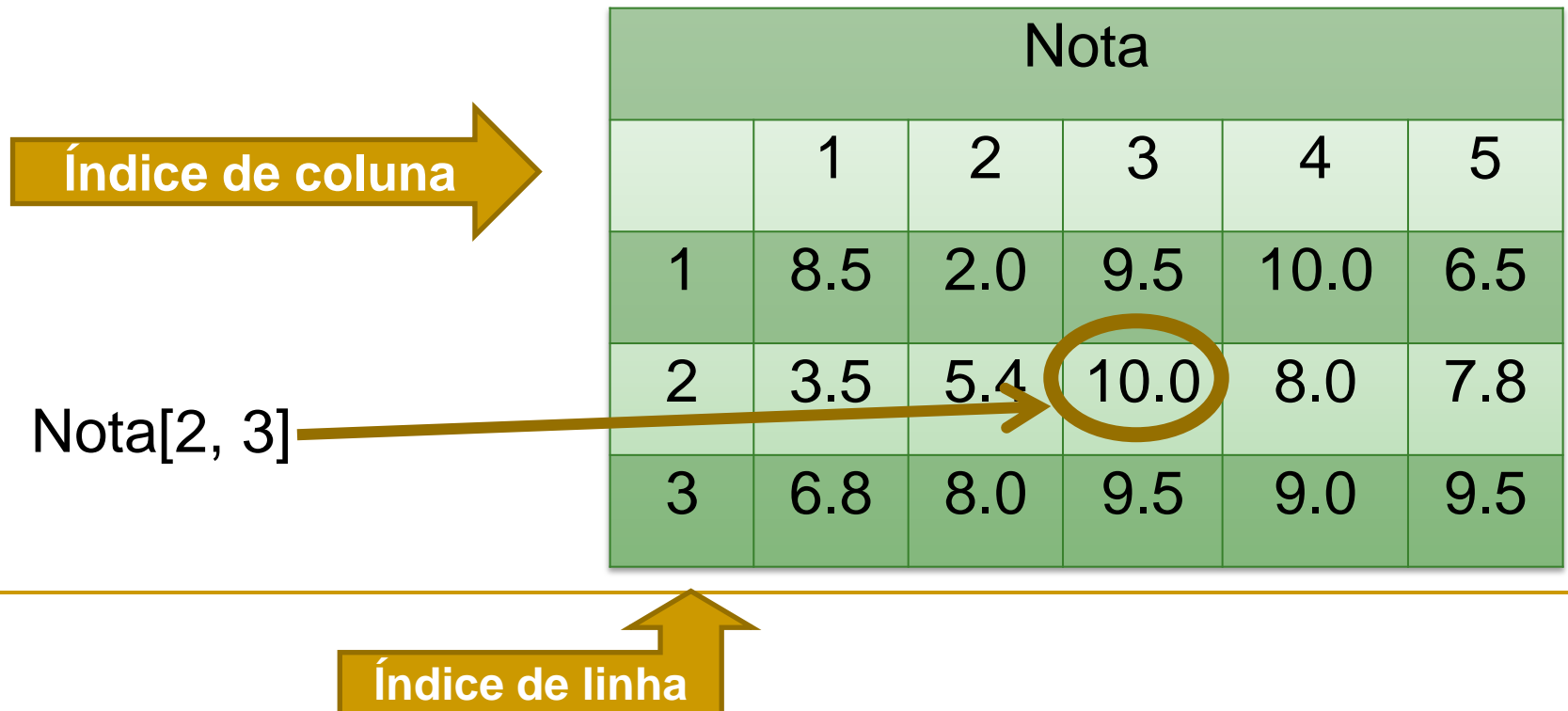
- ❑ Anteriormente foi trabalhada variável indexada de **uma dimensão**, conhecida por **vetor**.
 - ❑ Porém, uma variável indexada pode ter **um número maior de dimensões**.
 - ❑ Neste caso chama-se **Matriz**.
-

3. Variável indexada: matriz

- ❑ Se constitui de um conjunto de espaços de memória.
 - ❑ Cada espaço deste conjunto possibilita armazenar um valor de cada vez.
 - ❑ Todos os valores armazenados em uma matriz devem ser de mesmo tipo.
 - ❑ Assim como no vetor, os elementos da matriz ficam “lado a lado” na memória.
-

3. Matriz: declaração

- ❑ A declaração deve conter o nome da variável, o tipo de dados e a quantidade de valores a serem armazenados em cada linha e em cada coluna.
- ❑ Nota: Vetor[1..3, 1..5] de Real



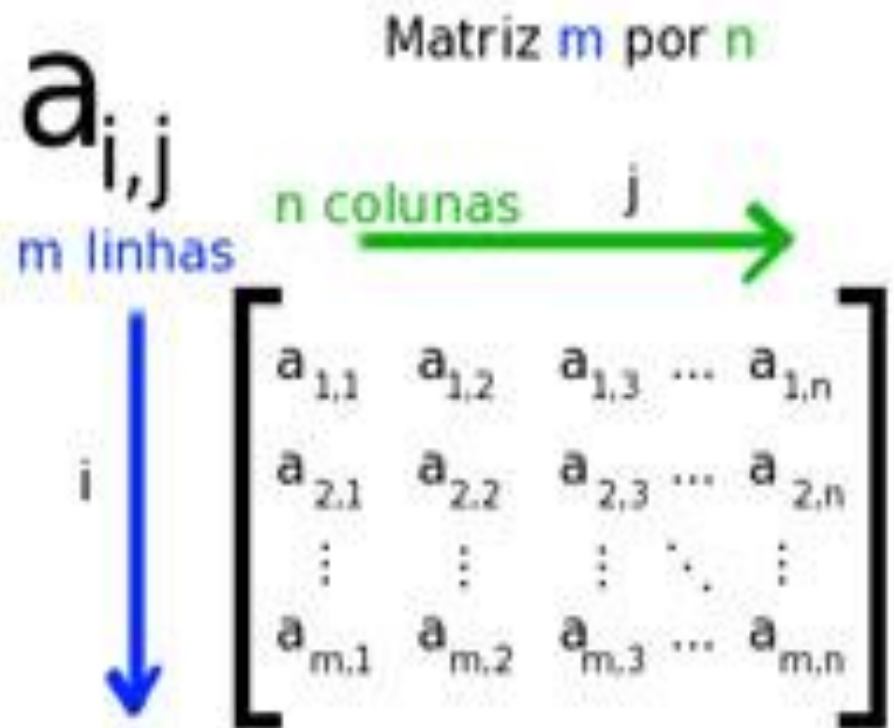
The diagram illustrates a matrix structure. A horizontal arrow labeled "Índice de coluna" points to the columns of the matrix. A vertical arrow labeled "Índice de linha" points to the rows. The matrix itself is a table with 3 rows and 5 columns. The first row is the header, with the title "Nota" centered above the column indices 1 through 5. The subsequent rows contain numerical values. The value 10.0 at the intersection of row 2 and column 3 is highlighted with a brown circle. An arrow points from the text "Nota[2, 3]" to this specific cell.

Nota					
	1	2	3	4	5
1	8.5	2.0	9.5	10.0	6.5
2	3.5	5.4	10.0	8.0	7.8
3	6.8	8.0	9.5	9.0	9.5

3. Matriz: manipulação

- ❑ A atribuição de uma determinada nota no respectivo espaço de memória deve ser realizada com uso de dois índices (linha e coluna), como exemplo:

- ✓ `Nota[1,1] <- 8.5`
- ✓ `Nota[1,3] <- 2.0`
- ✓ `Nota[2,5] <- 7.8`
- ✓ `Leia (Nota[3,1])`
- ✓ `Leia (Nota[3,4])`

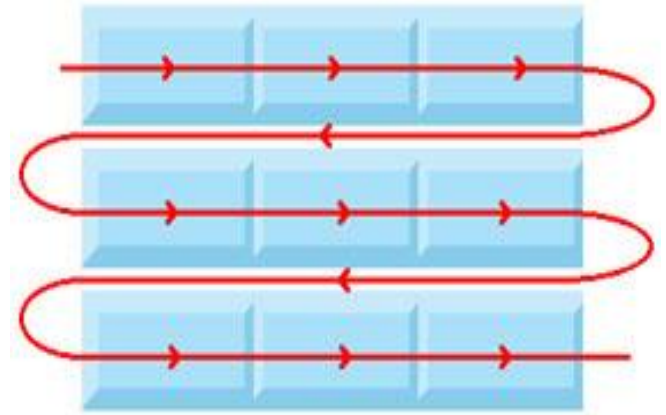


3. Matriz: manipulação

- ❑ No caso de **manipulação de todos os elementos** de uma matriz, ou mesmo de todos os elementos de uma linha ou coluna da matriz, podem-se utilizar **índices variáveis** no lugar de valores numéricos constantes. Para tanto é essencial o uso de **estruturas de repetição**.

Exemplo:

```
Para lin de 1 Ate 3 Faca  
  Para col de 1 Ate 5 Faca  
    Leia(Nota[lin, col])  
  FimPara  
FimPara
```



4. Matriz: manipulação

- ❑ Como seria um código para percorrer a matriz no sentido coluna?
- ❑ Exemplo:

```
Para col de 1 Ate 5 Faca  
  Para lin de 1 Ate 3 Faca  
    Leia(Nota[lin, col])  
  FimPara  
FimPara
```



Vamos Programar!



5. Exercícios com matriz



- 1) Escrever um algoritmo para armazenar valores inteiros em um array 4 X 3. Calcular e exibir a média de todos os valores armazenados na matriz.
- 2) Desenvolver um algoritmo que possibilite carregar, com números inteiros, os dados em uma matriz de 3 linhas e 4 colunas. Calcule e apresente a soma dos elementos da linha 3.
- 3) Escrever um algoritmo para armazenar valores inteiros em uma matriz 4 X 3. Após todos os dados armazenados, calcular e exibir a média dos valores armazenados em cada coluna, uma por uma.