

Disciplina

R para Ciência de Dados

Aula 3

VETORES E MATRIZES

Alan Rodrigo Panosso

Departamento de Ciências Exatas

alan.panosso@unesp.br



Estrutura de Dados

O que é uma
ESTRUTURA DE DADOS?

A declaração de variáveis, uma a uma, é suficiente para a codificação algorítmica da solução de uma ampla gama de problemas até agora estudados.

Entretanto, esse tipo de declaração é ***insuficiente*** para resolver um grande número de problemas computacionais.

Imagine, por exemplo, como faríamos para construir um algoritmo, que lesse os nomes de 36 alunos e imprimisse a respeito desses alunos.

início

caractere: nome1, nome2, ..., nome36;

leia (nome1, nome2,..., nome36);

Escreva(nome1, nome2, ..., nome36);

fim.

Observe que isso não seria uma tarefa simples, pois teríamos que definir 36 variáveis do tipo **caractere**.

Nomes

Allana Lojó Pizápio
Ana Carolina Oliveira de Andrade
Ana Leticia Antonio Vital
Ana Paula dos Santos Pereira
Aymar Orlandi Neto
Bianca Oliveira Rocha
Cíntia Lionela Ambrosio de Menezes
Danilo Silva Teixeira
Ewerton Kenji Sakita
Isabela Alonso Buzetti
Jaquelyne Poliszuk de Azevedo
Jaurés Barbosa Guisard
Jean da Silva Gomes Júnior
Juan Vítor Ruiz
Juliana Prado Gusson
Julliene Stephanie Guaraldi Monteiro da Silva
Laís Pedroso Borges
Leticia Yumi Ichikawa
Lucas Spada
Maira da Silva Rodrigues
Márcia Evelyn Alves
Marcos Vinicius Lopes Queiroz
Maria Luiza Diniz dos Reis
Mariana Retuci Pontes
Marina Guimarães Gemini
Mauricio Lucas dos Santos do Amaral
Mayke Leão Goulart
Patrícia Fernanda Rosalem
Paulo Henrique Furlan
Rafael de Farias Barbosa
Rafael de Oliveira
Renata Cannizza Bernardes da Rosa
Roberta de Oliveira Barbosa
Thaís Bravo Picão
Thaís Dias
Thaís Pigozzi Codo Amaral

A quantidade de tipos de dados primitivos (**caractere**, **real**, **inteiro**, **booleano**) não é suficiente para representar toda e qualquer informação que possa surgir.

Assim, em muitas situações, esses recursos de representação são escassos, o que poderá ser suprimido se existisse mais tipos de dados ou, ainda melhor, se esses tipos pudessem ser **construídos**, à medida que fossem necessários.

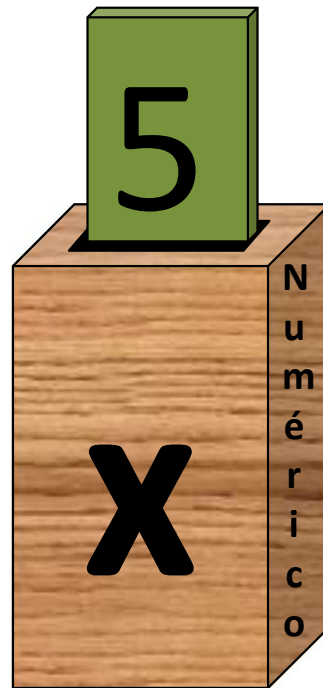
Construiremos novos tipos de dados a partir da composição de tipos primitivos.

Esses novos tipos tem o formato denominado **estrutura de dados** que define como os tipos primitivos são organizados.

ESTRUTURA DE DADOS
=
CONJUNTO DE DADOS

Estrutura de dados

Uma **variável** pode ser vista como uma caixa com um rótulo ou nome colado a ela, que num dado instante guarda um determinado objeto, essa caixa pode ter seu conteúdo alterado diversas vezes. Contudo, o conteúdo deve ser sempre do mesmo tipo.



Identificador da variável: X

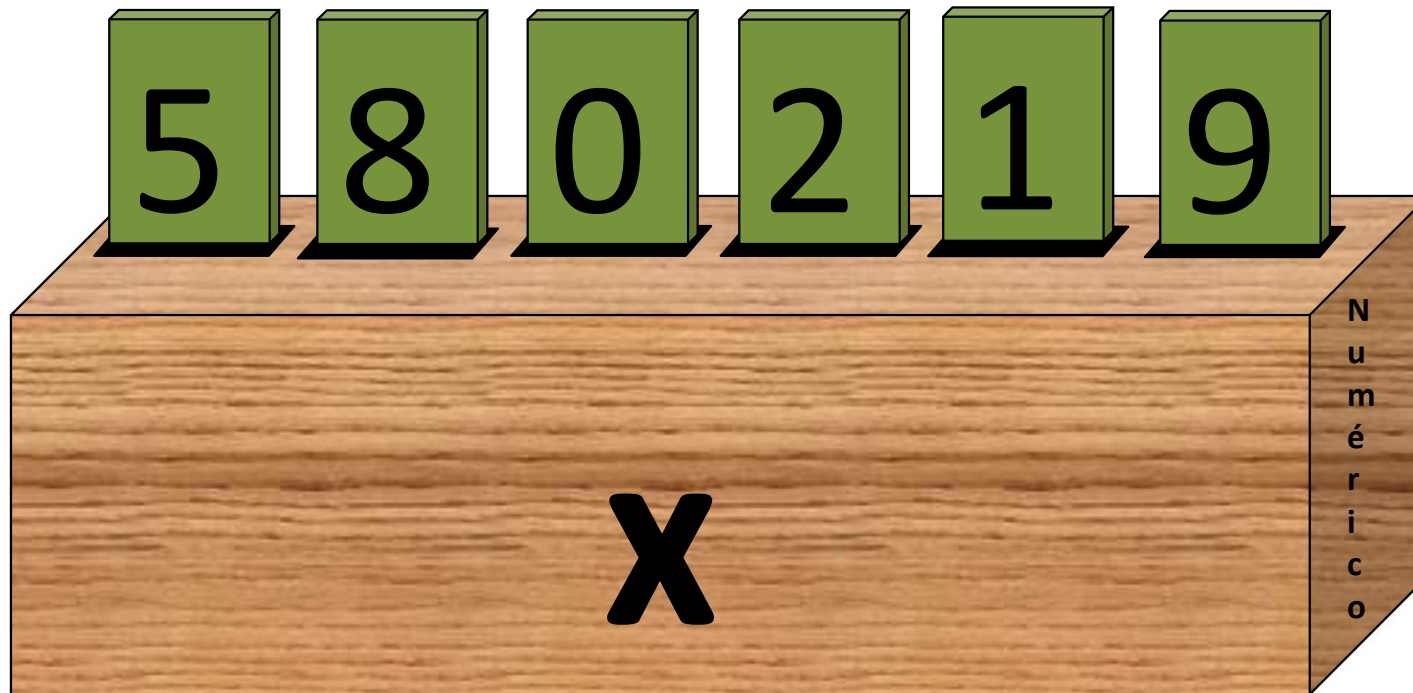
Tipo: Numérico

Conteúdo: 5

Armazena um valor

Estrutura de dados

De forma análoga, no conceito de **estrutura de dados** uma caixa poderá comportar não apenas uma e somente uma informação, a caixa comportará um conjunto de dados, desde que previamente organizada, ou seja, dividida em compartimentos.



Identificador da Estrutura: X

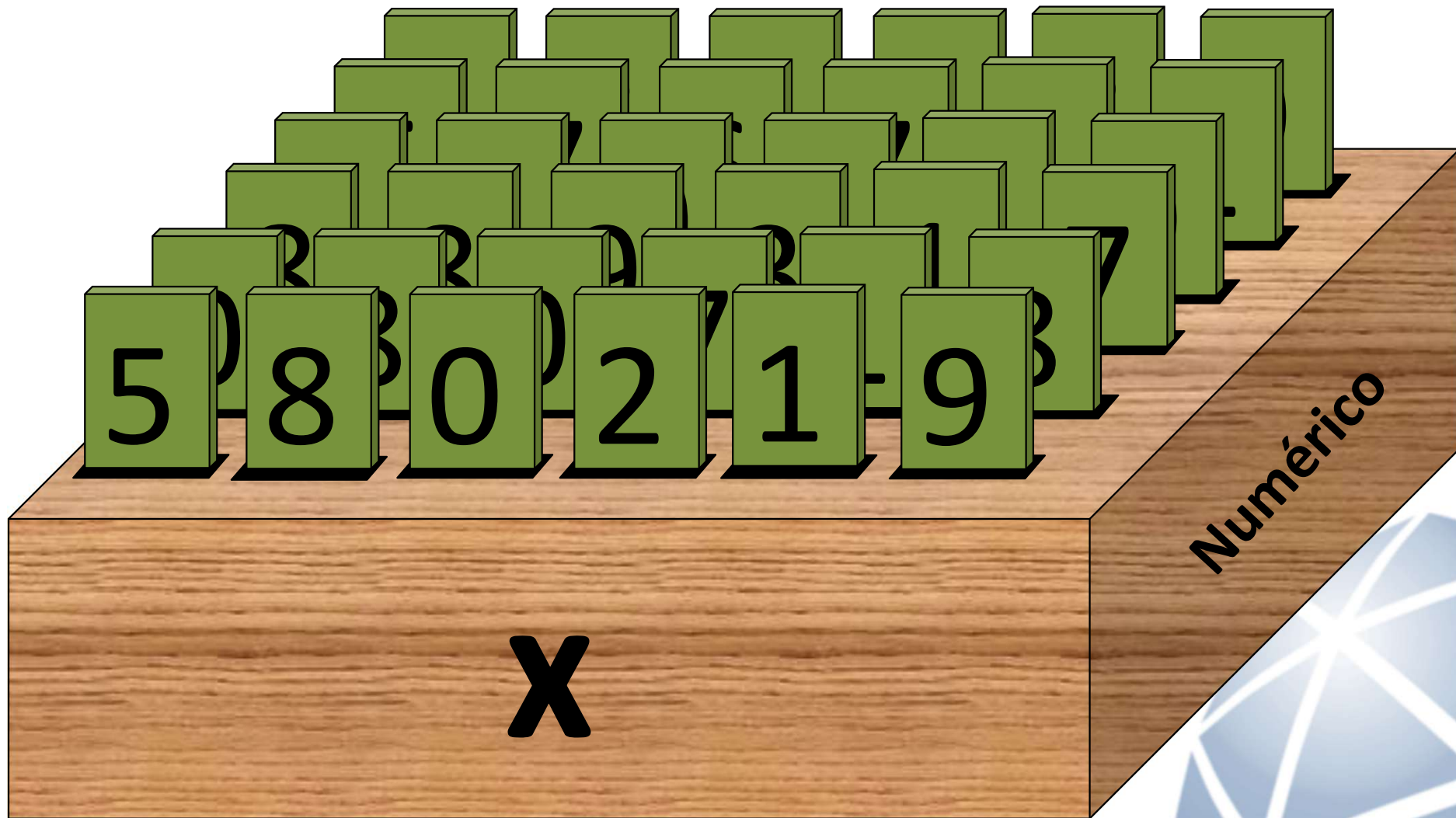
Tipo: Numérico

Conteúdo: {5, 8, 0, 2, 1, 9}

Armazena 6 valores

Estrutura de dados

Os compartimentos, podem estar arrançados, assim.



Armazena $6 \times 6 = 36$ valores

Estrutura de dados

Variável = Um elemento

Estrutura de dados = Um conjunto

Quando uma **estrutura de dados** é composta de variáveis **com o mesmo tipo primitivo**, temos um conjunto **homogêneo** de dados.



Estrutura de dados

Se um conjunto homogêneo de dados é composto por variáveis do mesmo tipo, um **conjunto heterogêneo de dados** é composto por elementos que **não são do mesmo tipo primitivo**.



Alunos

Rosa

Ana

Pedro

Joana

José

Ana

João

Conjunto de Alunos

=

Estrutura de dados

homogênea

=

Estrutura de dados do
tipo CHARACTER

Conjunto de Idades

=

**Estrutura de dados
homogênea**

=

**Estrutura de dados do
tipo INTEIRO**

Idades

23

38

47

19

26

52

29

O que é estrutura de dados
UNIDIMENSIONAL?

**ESTRUTURA DE DADOS
UNIDIMENSIONAL**

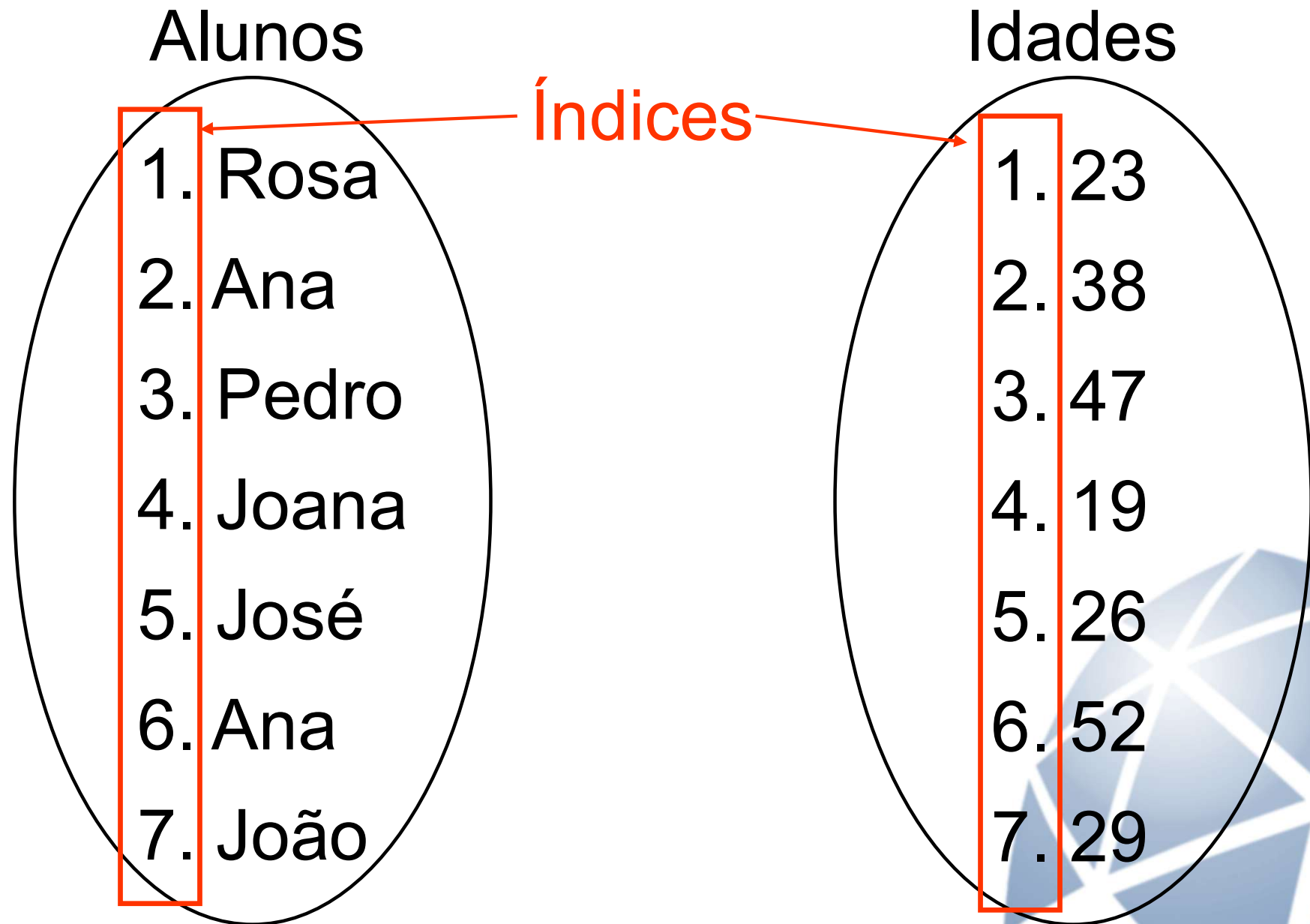
=

CONJUNTO DE DADOS

onde cada elemento é

identificado por um

único índice



ESTRUTURA DE DADOS UNIDIMENSIONAL

=

VETOR em **Computação**

Qual o nome e a idade
do Aluno 5?

Alunos

1.	Rosa
2.	Ana
3.	Pedro
4.	Joana
5.	José
6.	Ana
7.	João

José, 26!

Idades

1.	23
2.	38
3.	47
4.	19
5.	26
6.	52
7.	29

Como isso funciona
no Computador?

Aluno

1. Rosa
2. Ana
3. Pedro
4. Joana
5. José
6. Ana
7. João

Memória RAM

...	
[1]	Rosa
[2]	Ana
[3]	Pedro
[4]	Joana
[5]	José
[6]	Ana
[7]	João
...	

Índices Unidimensionais

Alunos

Estrutura de Dados

Alunos[5] = José

Vetores

Idades

1. 23

2. 38

3. 47

4. 19

5. 26

6. 52

7. 29

Índices
Unidimensionais

Memória
RAM

[1]

23

[2]

38

[3]

47

[4]

19

[5]

26

[6]

52

[7]

29

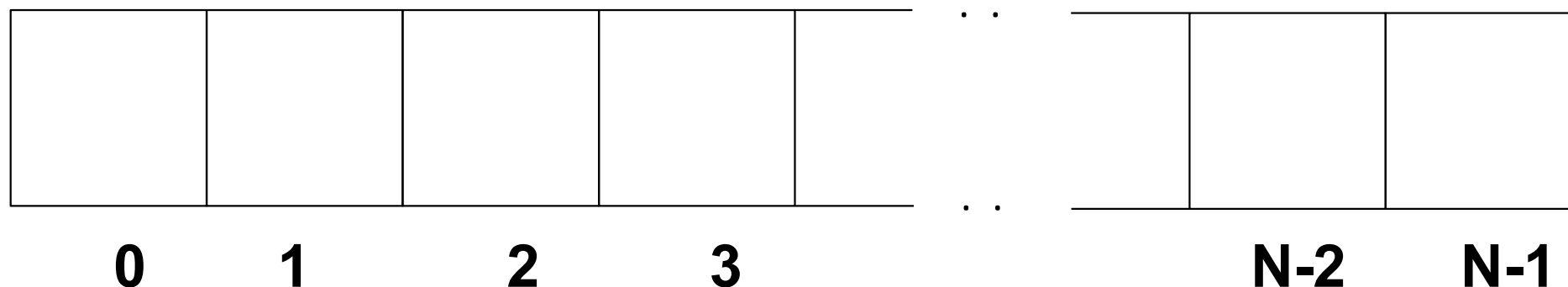
...

Estrutura de Dados
Idades

Idades[5] = 26

Vetores

Vetor genérico:



A manipulação dos valores de um Vetor é realizada pela determinação inicial do **identificador do vetor** e depois, especificar qual a **posição** que o valor se encontra. **Onde N é o número de elementos do vetor.**

Identificador [**posição**]

Assim, para mostrar um dado em particular dentro do vetor VCLASSE, declarado anteriormente.

idades[**3**]

Vetores

Após isolar um único elemento do vetor, poderemos manipulá-lo por meio de qualquer operação de entrada, saída e atribuição:

Exemplo: Dado o vetor VCLASSE:

8,5	7,3	9,6	6,4	...		7,3	8,9
0	1	2	3	...		8	9

VCLASSE [**3**] \leftarrow 2,8;

8,5	7,3	9,6	2,8	...		7,3	8,9
0	1	2	3	...		8	9

Vetores

8,5	7,3	9,6	6,4	7,4	5,8	4,0	3,5	7,3	8,9
0	1	2	3	4	5	6	7	8	9

Escreva(VCLASSE [4])

7,4

VCLASSE

Escreva(VCLASSE [0] + VCLASSE [2])

8,5 + 9,6 = 18,1

Se $D = 7$, qual será a saída para:

Escreva(VCLASSE [D-6] * VCLASSE [D+1])

7,3 * 7,3 = 53,29

Escreva(VCLASSE [D+2] - VCLASSE [2³])

8,9 - 7,3 = 1,6

Escreva(VCLASSE [VCLASSE [1] - VCLASSE [D+1]])

8,5

Vetores

Construa um programa que leia as médias de cinco alunos e armazene-as em um vetor, entretanto, o programa deve inicialmente perguntar o número de elementos que serão armazenados no vetor.

	A	B
1	Nomes	Média
2	DIEGO GABRIEL TORRES DINI	6,05
3	DIOGO TIAGO DA SILVA	2,07
4	ELIZEU DE SOUZA LIMA	8,00
5	EPITÁCIO JOSÉ DE SOUZA	4,01
6	HERNANDES ANDRADE QUEIROZ	9,05

Algoritmo

início

inteiro: i, n;

leia(n);

Real: medias[n];

Para i = 0 até 4 passo 1 *// leitura das notas*

 leia (notas[i]); *// entrada de dados*

Próximo i;

fim

Vetores

Implemente o algoritmo anterior para que ele imprima as notas com a posição do vetor entre colchetes, como no exemplo abaixo. Para deixar o programa mais dinâmico, insira as medias utilizando as chaves:

Algoritmo

início

```
Real: medias[5];  
leia(medias); // {6.05, 2.07, 8.00, 4.01, 9.01}  
// impressão das notas  
Escreva("Vetor de médias");  
Para i = 0 até 4 passo 1  
    Escreva("[", i, "]", notas[i], "Nova Linha");  
Próximo i;
```

Fim

Vetores

Modifique o programa anterior, de modo que ele calcule e mostre a soma das médias, utilize a estratégia de acumular operações de adição em uma variável destinada à soma (utilize a estrutura de repetição for).

```
real: soma;  
inteiro: i;  
leia (notas);  
soma ← 0;  
Para i=0 até 4 Passo 1  
    soma ← soma + medias[ i ];  
Próximo i
```

$$Soma = \sum_{i=1}^n x_i$$

Em seguida, calcule a média geral desses valores

$$média = \frac{\sum_{i=1}^n x_i}{n}$$

Primeira Iteração

soma \leftarrow 0;
Para i=0 **até** 4 **Passo** 1
 soma \leftarrow soma + medias[i];
Próximo i

i=0; soma=0
soma = soma + medias[0]
soma = 0 + 6,05

medias

6,05	2,07	8,00	4,01	9,05
0	1	2	3	4

soma

6,05

Segunda Iteração

soma \leftarrow 0;

Para $i=0$ até 4 Passo 1

soma \leftarrow soma + medias[i];

Próximo i

$i=1$;

soma = soma + medias[1]

soma = 6,05+2,07

medias

6,05	2,07	8,00	4,01	9,05
0	1	2	3	4

soma

8,12

Terceira Iteração

soma \leftarrow 0;

Para i=0 até 4 Passo 1

soma \leftarrow soma + medias[i];

Próximo i

i=2;

soma = soma + medias[2]

soma = 8,12+8,00

medias

6,05	2,07	8,00	4,01	9,05
0	1	2	3	4

soma

16,12

Quarta Iteração

soma \leftarrow 0;

Para $i=0$ até 4 Passo 1

soma \leftarrow soma + medias[i];

Próximo i

$i=3$;

soma = soma + medias[3]

soma = 16,12+4,01

medias

6,05	2,07	8,00	4,01	9,05
0	1	2	3	4

soma

20,13

Quinta Iteração

soma \leftarrow 0;

Para $i=0$ até 4 Passo 1

soma \leftarrow soma + medias[i];

Próximo i

$i=4$;

soma = soma + medias[4]

soma = 20,13+9,05

medias

6,05	2,07	8,00	4,01	9,05
0	1	2	3	4

soma

29,18

O que é estrutura de dados
MULTIDIMENSIONAL?

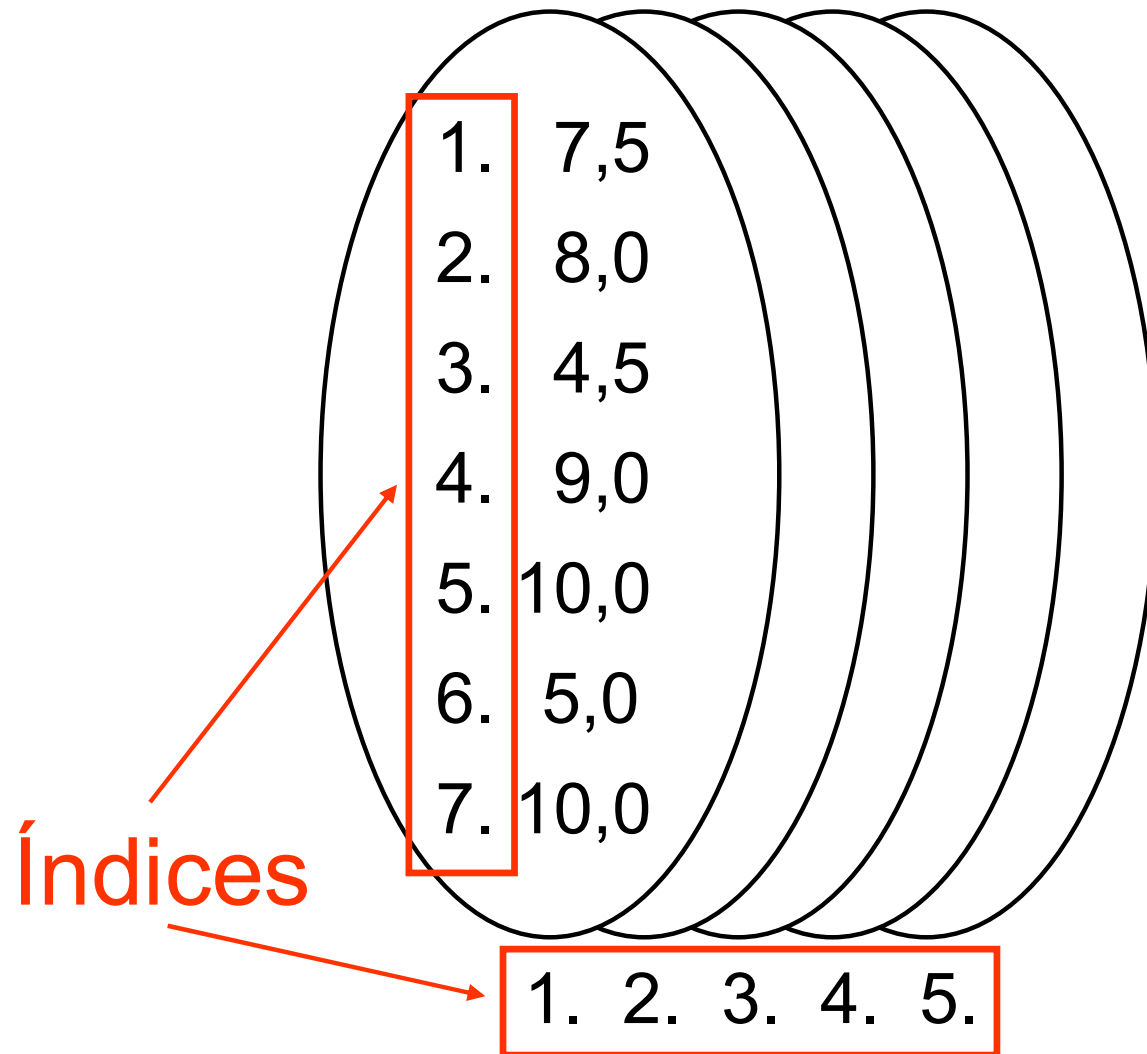
**ESTRUTURA DE DADOS
MULTIDIMENSIONAL**

=

CONJUNTO DE DADOS
onde cada elemento é
identificado por **MAIS**
de um índice

Estrutura de dados Multidimensional

Notas de 7 **alunos** por 5 **disciplinas**



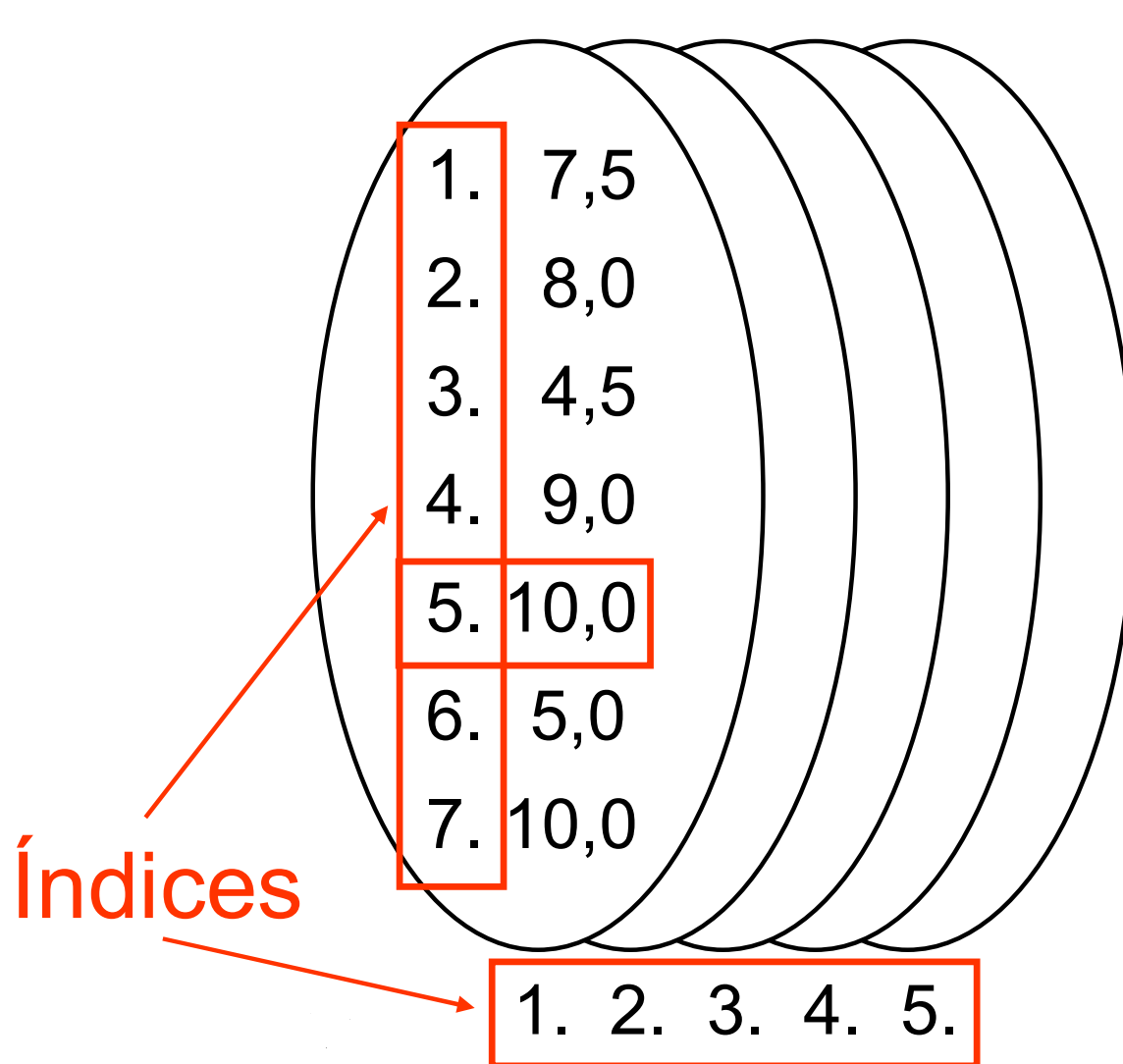
ESTRUTURA DE DADOS MULTIDIMENSIONAL

=

MATRIZ em **Computação**

Qual a nota do aluno 5
na disciplina 1?

Estrutura de dados Multidimensional

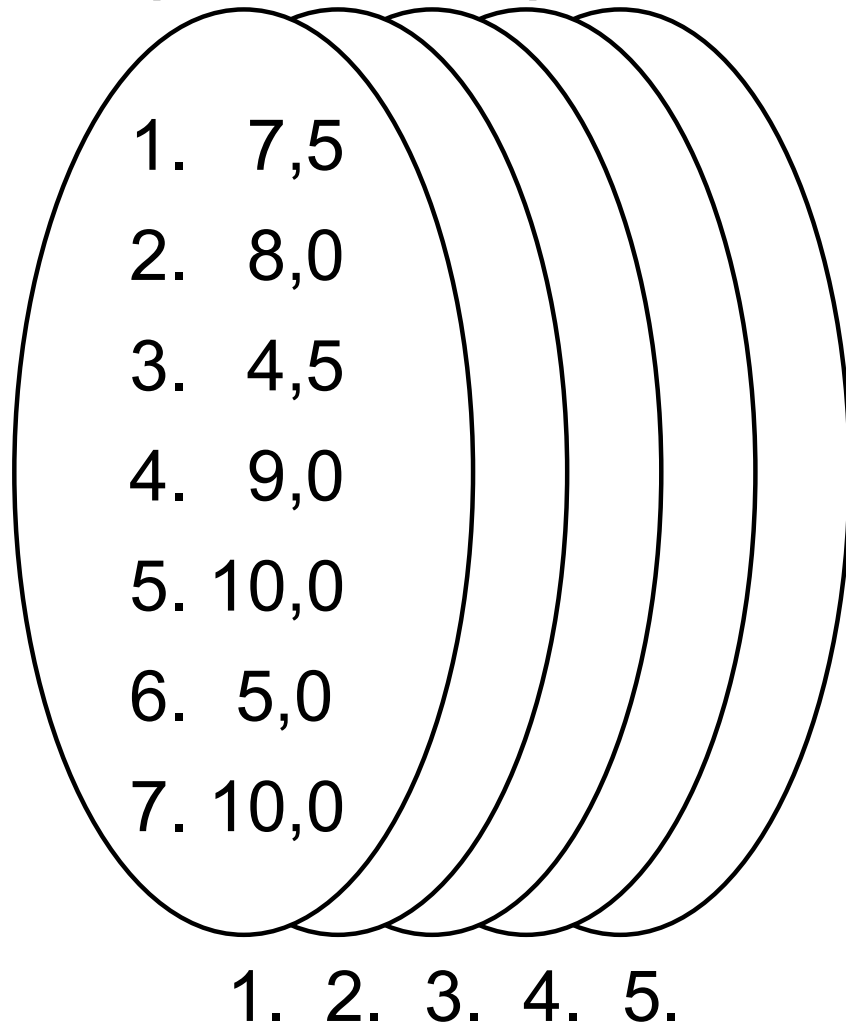


aluno 5
disciplina 1

10,0!

Como isso funciona
no Computador?

Notas de 7 alunos
por 5 disciplinas



Memória
RAM

Estrutura de Dados Notas

Índices Multidimensionais

	[1]	[2]	[3]	[4]	[5]
[1]	7,5	8,0	5,0	8,5	9,0
[2]	8,0	7,0	6,5	8,5	10,0
[3]	4,5	8,0	7,0	7,0	8,5
[4]	9,0	6,5	8,5	9,0	9,0
[5]	10,0	9,0	8,5	9,0	10,0
[6]	5,0	7,5	6,5	8,0	9,0
[7]	10,0	9,0	8,5	9,5	8,0

notas[5,1] = 10,0

Como se faz isso
no Algoritmo?

- **Declaração de Matrizes**

- Semântica:

- São estruturas de dados homogêneas multidimensionais que permitem agrupar **diversas informações** dentro de uma variável.
- Estas correspondem a um grupo de posições contínuas na memória que possuem o mesmo nome e o mesmo tipo de dado e são acessadas por **MAIS DE UM índice**.
- Seu tamanho é definido por **constantes inteiras e positivas** e a definição do seu nome segue as mesmas regras aplicadas para identificadores.

Note

A matriz notas ocupa 35 posições de memória (notas de 7 alunos X 5 disciplinas = criar 35 variáveis).

Ressalta-se que a matriz nota não armazena nome de aluno ou disciplina.

Somente Notas!

E se aumentar de 7
para 700 alunos e de
5 para 50 disciplinas?

É se acrescentar uma
nova dimensão?

Notas por
7 Alunos,
5 Disciplinas
e 10 Cursos?

Note

Cada dimensão da matriz é manipulada por um laço exclusivo. Por isso, ao se adicionar uma nova dimensão, deve-se criar uma nova variável e um novo laço para manipulá-la.

Note II

Análogo aos vetores, **não é possível operar com todos os elementos da matriz de uma só vez.**

Por isso, o correto é operar com cada um de seus elementos **isoladamente.**

Lembre-se: um laço para cada índice!

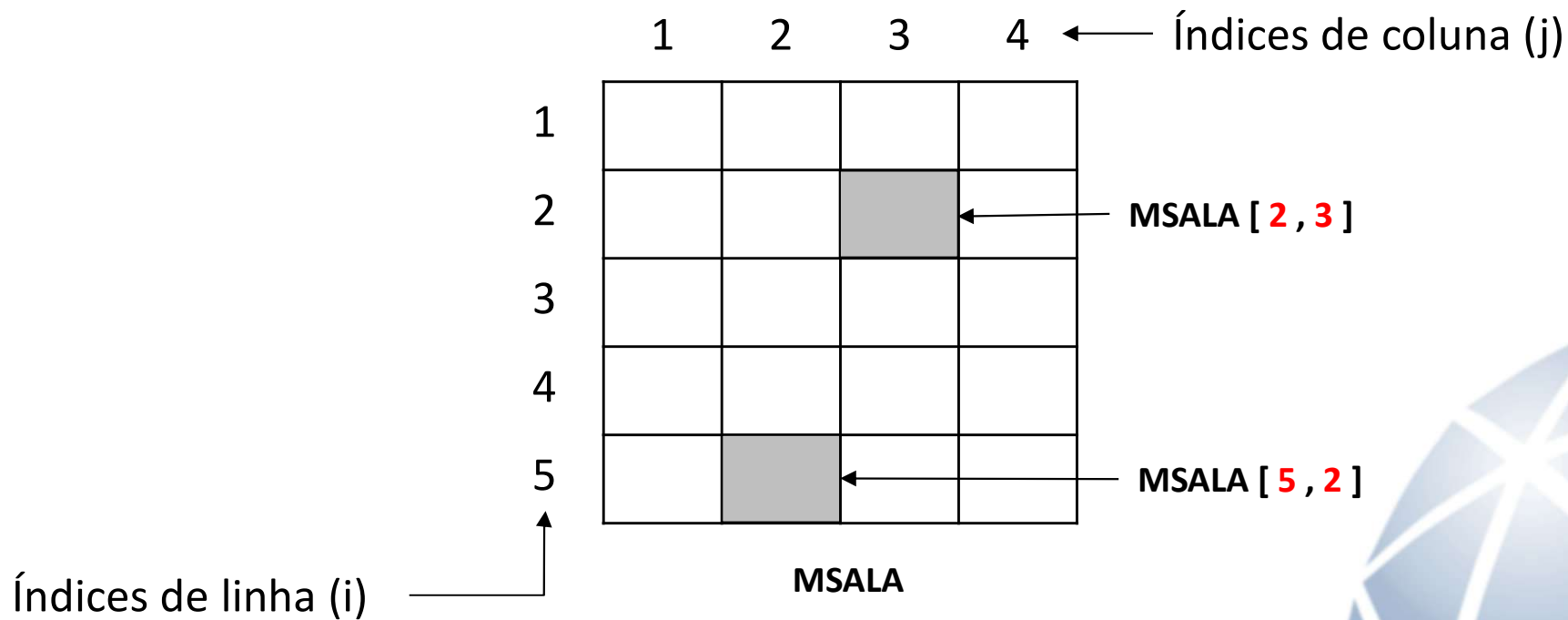
Note III

Por isso, semelhante aos vetores, o acesso a cada elemento de uma matriz é feito pela manipulação dos seus índices entre **[colchetes]**!

Estrutura de dados Multidimensional

Exemplos:

Real: `MSALA[4 , 5]` // declaração de MSALA como matriz do tipo com 2 dimensões (*i* linhas = 5 e *j* colunas = 4).



Estrutura de dados Multidimensional

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

início

Caractere SALA: MSALA[5, 4];

inteiro: i, j;

caractere: saida;

para i de 1 até 5 passo 1 faça

para j de 1 até 4 passo 1 faça

escreva(← "(" e i e "," e j e ")")

Próximo j;

escreva("Nova Linha");

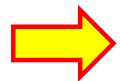
Próximo i;

Fim

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

início



Caractere SALA: MSALA[5, 4];

inteiro: i, j;

caractere: saida;

para i de 1 até 5 passo 1 faça

para j de 1 até 4 passo 1 faça

escreva(← "(" e i e "," e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

	1	2	3	4
1				
2				
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

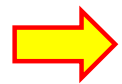
Algoritmo

início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saida;



para i de 1 até 5 passo 1 faça

para j de 1 até 4 passo 1 faça

escreva(← "(" e i e "," e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

	1	2	3	4
i = 1				
2				
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

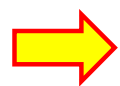
início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saída;

para i de 1 até 5 passo 1 faça



para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

	SALA			
	1	2	3	4
i = 1	(1,1)			
2				
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

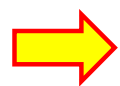
início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saida;

para i de 1 até 5 passo 1 faça



para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

i =

	1	2	3	4
1	(1,1)	(1,2)		
2				
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

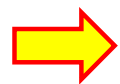
início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saída;

para i de 1 até 5 passo 1 faça



para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

i =

	1	2	3	4
1	(1,1)	(1,2)	(1,3)	
2				
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

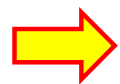
início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saida;

para i de 1 até 5 passo 1 faça



para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

	1	2	3	4
i = 1	(1,1)	(1,2)	(1,3)	(1,4)
2				
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saida;

para i de 1 até 5 passo 1 faça

para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;



Fim

	1	2	3	4
i = 1	(1,1)	(1,2)	(1,3)	(1,4)
2				
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

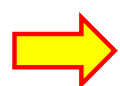
início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saída;

para i de 1 até 5 passo 1 faça



para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

i =

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

	SALA			
	1	2	3	4
1	(1,1)	(1,2)	(1,3)	(1,4)
2	(2,1)			
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

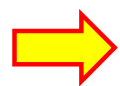
início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saída;

para i de 1 até 5 passo 1 faça



para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

i =

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

	1	2	3	4
1	(1,1)	(1,2)	(1,3)	(1,4)
2	(2,1)	(2,2)		
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

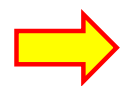
início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saída;

para i de 1 até 5 passo 1 faça



para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

i =

	1	2	3	4
1	(1,1)	(1,2)	(1,3)	(1,4)
2	(2,1)	(2,2)	(2,3)	
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

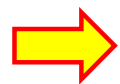
início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saída;

para i de 1 até 5 passo 1 faça



para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

i =

	1	2	3	4
1	(1,1)	(1,2)	(1,3)	(1,4)
2	(2,1)	(2,2)	(2,3)	(2,4)
3				
4				
5				

Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saida;

para i de 1 até 5 passo 1 faça

para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;



Fim

i =

	1	2	3	4
1	(1,1)	(1,2)	(1,3)	(1,4)
2	(2,1)	(2,2)	(2,3)	(2,4)
3				
4				
5				

Então, o processo continua
sucessivamente até...



Exemplo 1: Construa um algoritmo para criar e imprimir a matriz MSALA com a identificação de cada célula da matriz.

Algoritmo

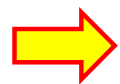
início

Caractere **SALA**: MSALA[5, 4];

inteiro: i, j;

caractere: saída;

para i de 1 até 5 passo 1 faça



para j de 1 até 4 passo 1 faça

escreva(← "(" e i e ", " e j e ")")

Próximo j;

escreva("Nova Linha");

Próximo i;

Fim

	1	2	3	4
1	(1,1)	(1,2)	(1,3)	(1,4)
2	(2,1)	(2,2)	(2,3)	(2,4)
3	(3,1)	(3,2)	(3,3)	(3,4)
4	(4,1)	(4,2)	(4,3)	(4,4)
i = 5	(5,1)	(5,2)	(5,3)	(5,4)

RESOLVER A

LISTA 05