

Matriz

Introdução à Ciência da Computação I
Simone Senger Souza

Introdução

- Como armazenar um **nome** em um programa?
 - Uso de um vetor de caracteres (string)
- E um conjunto de nomes?
 - Suponha que seja necessário armazenar todos os nomes da turma e depois ordená-los segundo algum critério
 - Necessário uma **estrutura composta bidimensional**

ESTRUTURAS COMPOSTAS

- Pode-se organizar os dados dos tipos simples em tipos mais complexos formando-se as ESTRUTURAS COMPOSTAS
- Exemplo:
 - variáveis compostas unidimensionais (VETOR)
 - variáveis compostas bidimensionais (MATRIZ)

ESTRUTURAS COMPOSTAS

- Pode-se organizar os dados dos tipos simples em tipos mais complexos formando-se as ESTRUTURAS COMPOSTAS
- Exemplo:
 - variáveis compostas unidimensionais (VETOR)
 - variáveis compostas bidimensionais (MATRIZ)

Matriz de Nomes

nomes =

A	N	A	\0		
J	O	A	0	\0	
B	R	U	N	0	\0

DUAS DIMENSÕES: QUANTIDADE DE
NOMES x OS NOMES(QTDE DE LETRAS)

MATRIZ

$$A = \begin{array}{|cccc|} \hline 10 & 8 & 5 & 1 \\ \hline 5 & 7 & 7 & 7 \\ \hline 8 & 0 & 0 & 10 \\ \hline \end{array}$$

- Para fazer referência ou selecionar um determinado elemento da matriz usa-se dois índices: um representa a linha e outro a coluna da matriz

MATRIZ

A[0,0]	10	8	5	1
A =	5	7	7	7
	8	0	0	10

linha

coluna

A[1][2]

- Genericamente, um elemento da matriz é representado por:



MATRIZ

- Cada **variável indexada bidimensional** é associada à uma posição de memória, como acontece com variáveis simples.
- Exemplo:



Declaração de Variável Indexada Bidimensional

- Na linguagem C:

```
int a[20][10];
```

- Com valor constante:

```
const int MAX = 10  
int a[MAX][MAX];
```

Declaração de Variável Indexada Bidimensional - Exemplos

- Exemplo:
 - Declaração de uma matriz S com no máximo 80 nomes e no máximo 30 letras cada nome.

```
char S[80][30];
```

Exercício

- Escreva a matriz resultante:

$A[0][0] = 15;$

para i de 0 até 4 faça

para j de 1 até 4 faça

$A[i][j] = A[i][j-1] / 2 ;$

fim-para;

$A[i+1][0] = A[i][0] + 2;$

fim-para;



Exemplos de Programas com Variáveis Compostas Bidimensionais

MATRIZ - Exemplo 1

- Ler uma tabela de 10 linhas e 3 colunas armazenando-as em uma matriz de inteiros
TAB

MATRIZ - Exemplo 1

```
int main(){  
int TAB[10][3];  
int i,j;  
for(i=0; i<10; i++){  
  for(j=0; i<3; j++){  
    scanf("%d", &TAB[i][j]);  
  }  
}
```

i	j	
0	0	ler TAB[0][0]
	1	ler TAB[0][1]
	2	ler TAB[0][2]
	3	
1	0	ler TAB[1][0]
	1	ler TAB[1][1]
	2	ler TAB[1][2]
	3	

etc.

MATRIZ - Exemplo 2

- Ler uma tabela de M linhas e N colunas armazenando-as em uma matriz A . Exibir a matriz A .

MATRIZ - Exemplo 2

```
#define MAX 100
```

Número de
linhas

```
int main(){
```

```
int TAB[MAX][MAX];
```

Número de
colunas

```
int i,j,m,n;
```

```
printf("Linhas="); scanf("%d",&m);
```

```
printf("Colunas="); scanf("%d",&n);
```

```
for(i=0; i<m; i++){  
    for(j=0; i<n; j++){  
        printf("TAB[%d][%d]=",i,j);  
        scanf("%d", &TAB[i][j]);  
    }  
}
```

Leitura da
Tabela

```
for(i=0; i<m; i++){  
    for(j=0; i<n; j++){  
        printf("TAB[%d][%d]=%d",i,j,TAB[i][j]);  
    }  
}
```

Escrita da
Tabela

MATRIZ - Exemplo 3

- Ler uma tabela de M linhas e N colunas armazenando-as em uma matriz de inteiros **A**
- Calcular a soma de todos os elementos da matriz **A**

MATRIZ - Exemplo 3

```
#define MAX 100
int main(){
int TAB[MAX][MAX], soma=0;
int i,j,m,n;
printf("Linhas="); scanf("%d",&m);
printf("Colunas="); scanf("%d",&n);
for(i=0; i<m; i++){
  for(j=0; i<n; j++){
    printf("TAB[%d][%d]=",i,j);
    scanf("%d", &TAB[i][j]);
  }
}
```

```
for(i=0; i<m; i++){
  for(j=0; i<n; j++){
    soma += TAB[i][j]);
  }
}
```

Soma dos
elementos

Escrita da
soma
dos elementos

```
printf("Soma=%d",soma);
return 0;
```

```
}
```

Exercício - solução

$A[0][0] = 15;$

para i de 0 até 4 faça

para j de 1 até 4 faça

$A[i][j] = A[i][j-1] / 2 ;$

fim-para;

$A[i+1][0] = A[i][0] + 2;$

fim-para;

$A =$

15	7	3	1
17	8	4	2
19	9	4	2
21	10	5	2

Exercícios

1. Faça um algoritmo que lê uma matriz de inteiros 6×3 , calcula e mostra o maior e o menor elemento da matriz e suas posições (linhas e colunas)
2. Faça um algoritmo que lê uma matriz de inteiros de n linhas e m colunas. O algoritmo deve somar cada uma das linhas da matriz e guardar o resultado da soma de cada linha em um vetor. A seguir mostrar o vetor resultante.
3. Considerando o algoritmo 2, mude para que no vetor seja armazenada a soma de cada coluna da matriz. A seguir, multiplique cada elemento da matriz pela soma da coluna e mostre a matriz resultante.