

INSTITUTO FEDERAL

Rio Grande do Sul

Campus Porto Alegre

Linguagem de Programação I

Prof. Fabio Okuyama

Curso Superior em Tecnologia em Sistemas para Internet

Matrizes e Vetores

Como Armazenar as notas de 50 alunos?

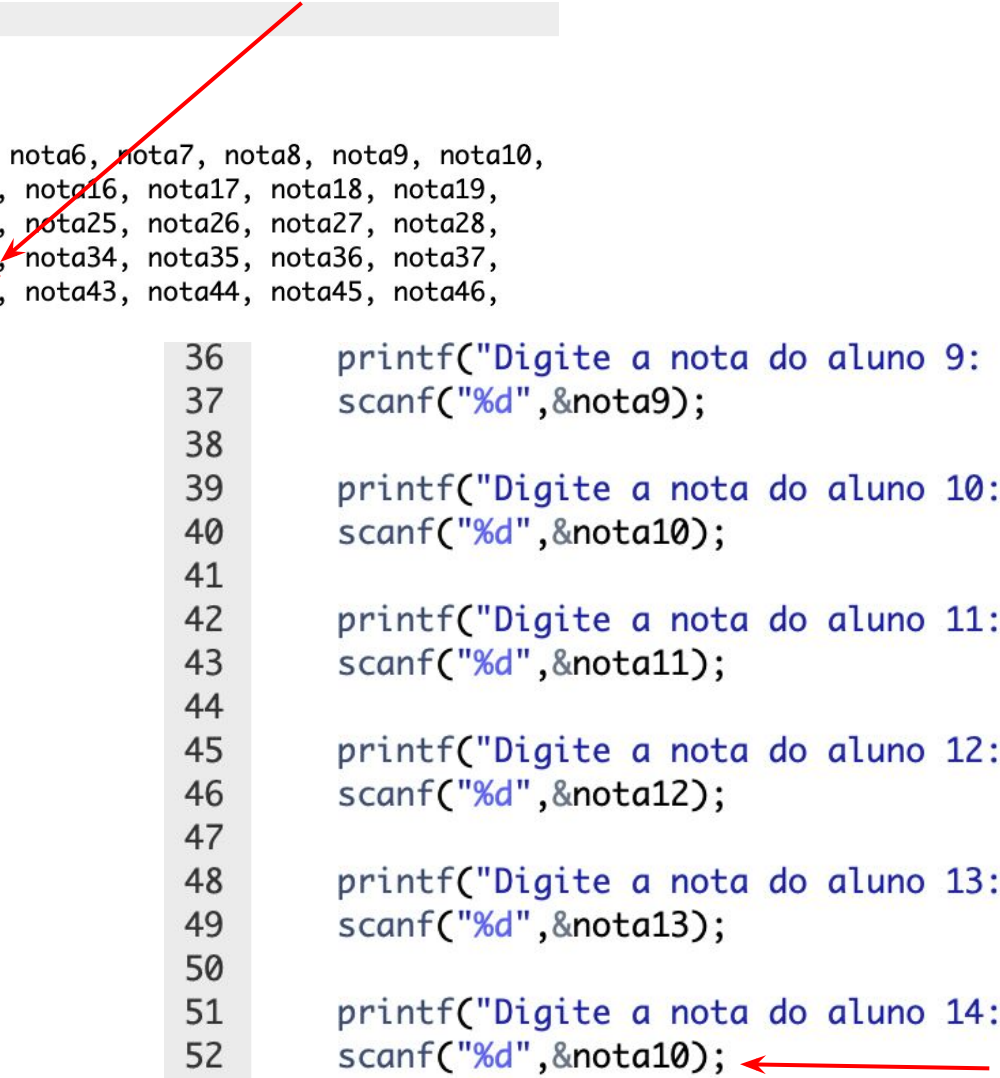
(sem usar vetores!)

```
1 |
2 #include <stdio.h>
3
4 int main() {
5     int nota1, nota2, nota3, nota4, nota5, nota6, nota7, nota8, nota9, nota10,
6     nota11, nota12, nota13, nota14, nota15, nota16, nota17, nota18, nota19,
7     nota20, nota21, nota22, nota23, nota24, nota25, nota26, nota27, nota28,
8     nota29, nota30, nota31, nota33, nota33, nota34, nota35, nota36, nota37,
9     nota38, nota39, nota40, nota41, nota42, nota43, nota44, nota45, nota46,
10    nota47, nota48, nota49, nota50;
11
12    printf("Digite a nota do aluno 1: ");
13    scanf("%d",&nota1);
14
15    printf("Digite a nota do aluno 2: ");
16    scanf("%d",&nota2);
17
18    printf("Digite a nota do aluno 3: ");
19    scanf("%d",&nota3);
20
21    printf("Digite a nota do aluno 4: ");
22    scanf("%d",&nota4);
23
24    printf("Digite a nota do aluno 5: ");
25    scanf("%d",&nota5);
26
27    printf("Digite a nota do aluno 6: ");
28    scanf("%d",&nota6);
29
30    printf("Digite a nota do aluno 7: ");
31    scanf("%d",&nota7);
32
33    printf("Digite a nota do aluno 8: ");
34    scanf("%d",&nota8);
35
36    printf("Digite a nota do aluno 9: ");
37    scanf("%d",&nota9);
38
39    printf("Digite a nota do aluno 10: ");
40    scanf("%d",&nota10);
41
42    printf("Digite a nota do aluno 11: ");
43    scanf("%d",&nota11);
44
45    printf("Digite a nota do aluno 12: ");
46    scanf("%d",&nota12);
47
48    printf("Digite a nota do aluno 13: ");
49    scanf("%d",&nota13);
50
51    printf("Digite a nota do aluno 14: ");
52    scanf("%d",&nota10);
53
54    printf("Digite a nota do aluno 15: ");
55    scanf("%d",&nota15);
56
```

Como Armazenar as notas de 50 alunos?

(sem usar vetores!)

```
1 |
2 #include <stdio.h>
3
4 int main() {
5     int nota1, nota2, nota3, nota4, nota5, nota6, nota7, nota8, nota9, nota10,
6     nota11, nota12, nota13, nota14, nota15, nota16, nota17, nota18, nota19,
7     nota20, nota21, nota22, nota23, nota24, nota25, nota26, nota27, nota28,
8     nota29, nota30, nota31, nota33, nota33, nota34, nota35, nota36, nota37,
9     nota38, nota39, nota40, nota41, nota42, nota43, nota44, nota45, nota46,
10    nota47, nota48, nota49, nota50;
11
12    printf("Digite a nota do aluno 1: ");
13    scanf("%d",&nota1);
14
15    printf("Digite a nota do aluno 2: ");
16    scanf("%d",&nota2);
17
18    printf("Digite a nota do aluno 3: ");
19    scanf("%d",&nota3);
20
21    printf("Digite a nota do aluno 4: ");
22    scanf("%d",&nota4);
23
24    printf("Digite a nota do aluno 5: ");
25    scanf("%d",&nota5);
26
27    printf("Digite a nota do aluno 6: ");
28    scanf("%d",&nota6);
29
30    printf("Digite a nota do aluno 7: ");
31    scanf("%d",&nota7);
32
33    printf("Digite a nota do aluno 8: ");
34    scanf("%d",&nota8);
35
36    printf("Digite a nota do aluno 9: ");
37    scanf("%d",&nota9);
38
39    printf("Digite a nota do aluno 10: ");
40    scanf("%d",&nota10);
41
42    printf("Digite a nota do aluno 11: ");
43    scanf("%d",&nota11);
44
45    printf("Digite a nota do aluno 12: ");
46    scanf("%d",&nota12);
47
48    printf("Digite a nota do aluno 13: ");
49    scanf("%d",&nota13);
50
51    printf("Digite a nota do aluno 14: ");
52    scanf("%d",&nota10);
53
54    printf("Digite a nota do aluno 15: ");
55    scanf("%d",&nota15);
56
```



Como Armazenar as notas de 50 alunos?

2 notas para cada aluno (aluno de 1 a 50)

1 - Prova A

2 - Prova B

```
1
2 #include <stdio.h>
3
4 int main() {
5     int notaA1, notaB1, notaA2, notaB2, notaA3, notaB3, notaA4, notaB4,
6     notaA5, notaB5, notaA6, notaB6, notaA7, notaB7, notaA8, notaB8, notaA9,
7     notaB9, notaA10, notaB10, notaA11, notaB11, notaA12, notaB12, notaA13,
8     notaB13, notaA14, notaB14, notaA15, notaB15, notaA16, notaB16, notaA17,
9     notaB17, notaA18, notaB18, notaA19, notaB19, notaA20, notaB20, notaA21,
10    notaB21, notaA22, notaB22, notaA23, notaB23, notaA24, notaB24, notaA25,
11    notaB25, notaA26, notaB26, notaA27, notaB27, notaA28, notaB28, notaA29,
12    notaB29, notaA30, notaB30, notaA31, notaB31, notaA32, notaB32, notaA33,
13    notaB33, notaA34, notaB34, notaA35, notaB35, notaA36, notaB36, notaA37,
14    notaB37, notaA38, notaB38, notaA39, notaB39, notaA40, notaB40, notaA41,
15    notaB41, notaA42, notaB42, notaA43, notaB43, notaA44, notaB44, notaA45,
16    notaB45, notaA46, notaB46, notaA47, notaB47, notaA48, notaB48, notaA49,
17    notaB49, notaA50, notaB50;
18 }
```


O que é uma matriz?



- A matriz é um tipo de dado usado para representar uma certa quantidade de variáveis que são referenciados pelo mesmo nome.
- Cada variável pode ser acessada individualmente pelo índice.
- Consiste em locações contíguas de memória.

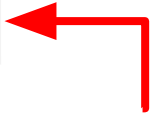
Matriz unidimensional: Vetores

índices



0 1 2 3 4 8 6 7 8

10	34	12	66	45	70	51	18	6
----	----	----	----	----	----	----	----	---



Conteúdo da Matriz

Para declarar uma matriz unidimensional, use a sintaxe:

Tipo nome [**tamanho**];

Declaração da matriz unidimensional do exemplo acima:

int idade[**9**];

Exemplo 1 - Utilizando um Vetor

CRIAR UM PROGRAMA QUE ARMAZENE EM UM VETOR NÚMEROS DE 0 A 18, DE 2 EM 2.

```
#include <stdio.h>
```

```
int main(void) {
```

```
    int meuVetor[10];
```

```
    int posicao;
```

```
    for(posicao = 0; posicao < 10; posicao++) {
```

```
        meuVetor[posicao] = posicao*2;
```

```
        printf("%d\n", meuVetor[posicao]);
```

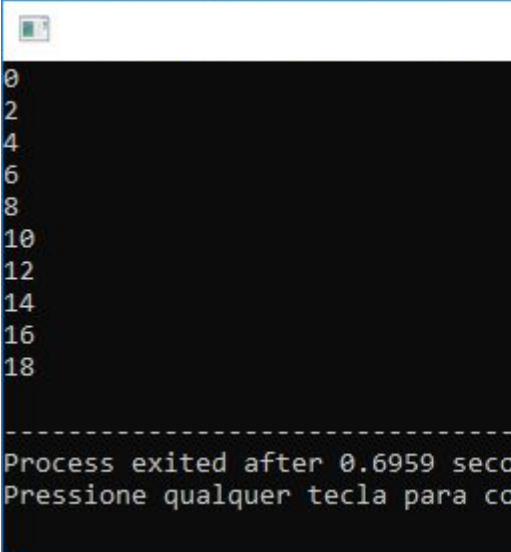
```
    }
```

```
    return 0;
```

```
}
```

//outra opção seria usar a linha:

```
//printf("Posição %d: %d\n", posicao, meuVetor[posicao]);
```



```
0
2
4
6
8
10
12
14
16
18
-----
Process exited after 0.6959 seconds
Pressione qualquer tecla para continuar
```


Exemplo 2 - Calcular a média da Turma

```
#include <stdio.h>

int main(void) {
    int notas[50], posicao, soma;

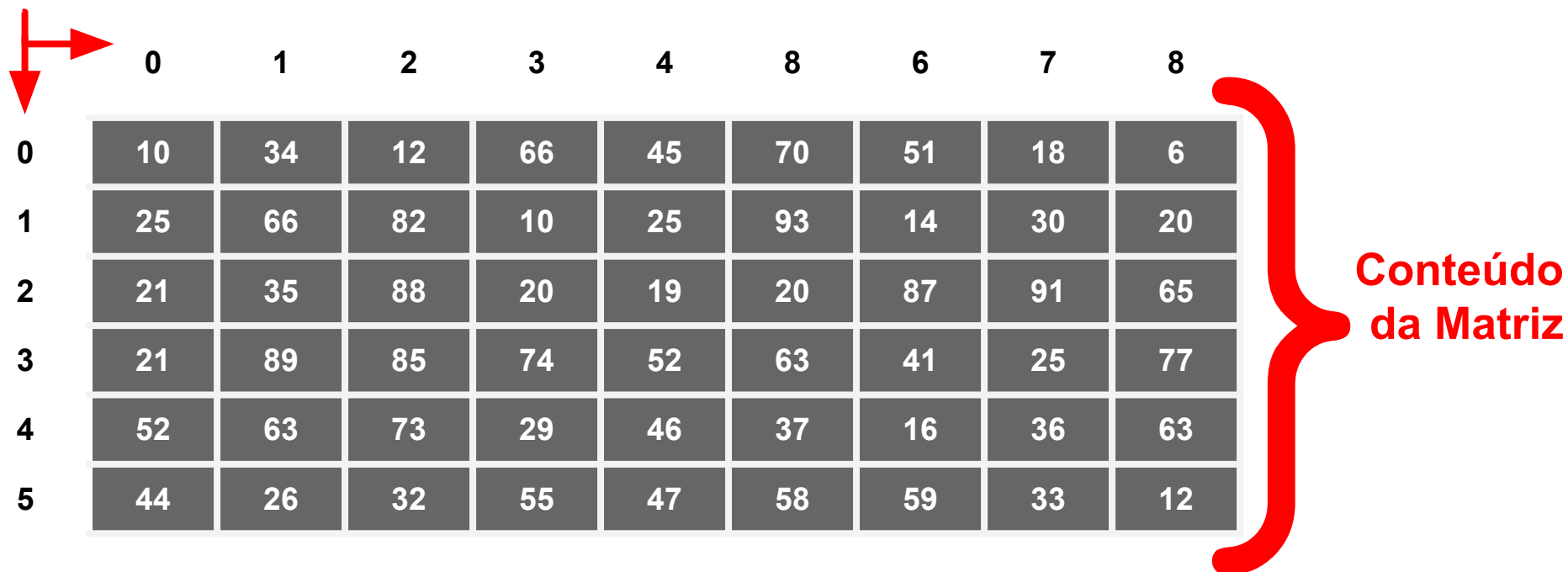
    // leitura dos alunos diretamente na posição do vetor
    for(posicao = 0; posicao < 50; posicao++){
        printf("\nDigite a Nota do aluno %d: ", posicao);
        scanf("%d", &notas[posicao]);
    }

    soma = 0;
    // percorre o vetor somando o valor das posições
    for(posicao = 0; posicao < 50; posicao++){
        soma = soma + notas[posicao];
    }

    printf("\nMédia da Turma: %d", soma / 50);
}
```

Matriz multidimensional

índices



	0	1	2	3	4	8	6	7	8
0	10	34	12	66	45	70	51	18	6
1	25	66	82	10	25	93	14	30	20
2	21	35	88	20	19	20	87	91	65
3	21	89	85	74	52	63	41	25	77
4	52	63	73	29	46	37	16	36	63
5	44	26	32	55	47	58	59	33	12

Para declarar uma matriz unidimensional, use a sintaxe:

Tipo nome [**tamanho 1**][**tamanho 2**] .. [**tamanho**];

Declaração da matriz multidimensional do exemplo acima:

```
int idade[6][9];
```

Exemplo - Cálculo da média dos Alunos

```
1  #include <stdio.h>
2  int main() {
3      int notas[50][3], linha, coluna, soma;
4
5      // leitura dos alunos diretamente nas posições do vetor
6      for(linha = 0; linha < 50; linha++){
7          for(coluna = 0; coluna < 3; coluna++){
8              printf("\nDigite Nota %d aluno %d: ", coluna, linha);
9              scanf("%d", &notas[linha][coluna]);
10         }
11     }
12     for(linha = 0; linha < 50; linha++){
13         soma = 0; // a soma deve ser zerada a cada troca de linha;
14         for(coluna = 0; coluna < 3; coluna++){
15             soma = soma + notas[linha][coluna];
16         }
17         printf("\nMédia do aluno %d: %d", linha, soma / 3);
18     }
19 }
```

ATENÇÃO

- Em C, os índices iniciam em **ZERO**.
- No índice, deve obrigatoriamente ser um inteiro
- Limites das matrizes não são verificados pelo compilador
- A responsabilidade no uso é do programador

```
#include <stdio.h>
int main(void) {
    int erro[10], posicao;
    for(posicao = 0; posicao < 1000; posicao++){
        erro[posicao] = posicao;
        printf("%d\n", erro[posicao]);
    }
}
```



Atribuição de valores na declaração

```
//Cria e inicializa os valores
```

```
int x[10] = {5, 120, 28, 38, 489, 596, 68, 78, 86, 95};
```

```
//Cria vetor de 3 posições e inicializa
```

```
char a[] = {'a', 'b', 'c'};
```

Importante: esse tipo de atribuição é válido apenas na declaração da variável!

Matrizes e Funções: Matrizes Globais

```
1  #include <stdio.h>
2  int notas[3]; // vetor declarado globalmente
3  int media();
4  int main() {
5      int posicao;
6      // leitura dos alunos diretamente na posição do vetor
7      for(posicao = 0; posicao < 3; posicao++){
8          printf("\nDigite a Nota %d: ", posicao);
9          scanf("%d", &notas[posicao]);
10     }
11     printf("\nMédia do aluno: %d", media());
12     return 0;
13 }
14 int media() {
15     int posicao, soma = 0;
16     // percorre o vetor somando o valor das posições
17     for(posicao = 0; posicao < 3; posicao++){
18         soma = soma + notas[posicao];
19     }
20     return soma / 3;
21 }
```

Matrizes e Funções: Matrizes por parâmetros

```
1  #include <stdio.h>
2  int media(int notas[3]);
3  int main(void) {
4      int notas[3]; // vetor declarado localmente
5      int posicao;
6      // leitura dos alunos diretamente na posição do vetor
7      for(posicao = 0; posicao < 3; posicao++){
8          printf("\nDigite a Nota %d: ", posicao);
9          scanf("%d", &notas[posicao]);
10     }
11     printf("\nMédia do aluno: %d", media(notas));
12     return 0;
13 }
14 int media(int notas[3]) {
15     int posicao, soma = 0;
16     // percorre o vetor somando o valor das posições
17     for(posicao = 0; posicao < 3; posicao++){
18         soma = soma + notas[posicao];
19     }
20     return soma / 3;
21 }
```