

Aula 30: Matrizes: linhas, colunas e percursos

Lógica de Programação – Superior · Bloco 3

José Dihego · IFBA

O que você precisa saber hoje

Objetivo: Declarar, ler e percorrer matrizes; trocar o papel de linha e coluna.

- `int m[L][C];` e acesso `m[i][j]` (linha `i`, coluna `j`)
- Dois laços: externo nas linhas, interno nas colunas
- `m[i][j]` percorre a linha `i`; `m[j][i]` percorre a coluna `i`
- Rotação horária: `nova[i][j] = m[n - 1 - j][i]`

```
for (int i = 0; i < n; i++) {  
    int soma_linha = 0, soma_coluna = 0;  
    for (int j = 0; j < n; j++) {  
        soma_linha += m[i][j];  
        soma_coluna += m[j][i];  
    }  
}
```

Erros que custam pontos

Errado	Certo	Por quê
<code>scanf("%d", m[i][j]);</code>	<code>scanf("%d", &m[i][j]);</code>	o & continua necessário
<code>for (j = 0; j < n; i++)</code>	<code>for (j = 0; j < n; j++)</code>	incremento da variável errada
<code>int m[10][10]; /* sem limite */</code>	<code>#define N 10</code>	use uma constante para recompilar fácil

```
int m[3][3] = {{0, 1, 0}, {1, 0, 0}, {0, 0, 1}};  
for (int i = 0; i < 3; i++) {  
    int ul = 0, uc = 0;  
    for (int j = 0; j < 3; j++) { ul += m[i][j];  
        uc += m[j][i]; }  
}
```

i	linha i	uns linha	uns coluna
0	0 1 0	1	1
1	1 0 0	1	1
2	0 0 1	1	1

Avaliação 3 – 2014.2, Questão I

avaliacao_3_ano_2014_2.pdf

A matriz é de permutação? (na prova, $N = 10$)

Estratégia:

1. Permutação: só 0 e 1, e exatamente um 1 por linha e por coluna
2. Para cada i , somar a linha i e a coluna i no mesmo laço
3. Qualquer soma diferente de 1 (ou valor fora de 0/1) reprova

Enunciado completo (com figuras e exemplos) no PDF da prova.

Solução (1/2)

```
1  #include <stdio.h>           // printf e scanf
2  #define MAX 100             // maior dimensão aceita
3
4  int main(void) {             // início do programa
5      int n, m[MAX][MAX];      // dimensão e matriz
6      scanf("%d", &n);         // lê N (10 na prova; ler N deixa testar com
                                // matrizes pequenas)
7      for (int i = 0; i < n; i++) // cada linha
8          for (int j = 0; j < n; j++) scanf("%d", &m[i][j]); // cada coluna
9      int ok = 1;              // supõe que é de permutação
10     for (int i = 0; i < n; i++) { // para cada índice i: linha i e coluna i
11         int uns_linha = 0, uns_coluna = 0; // quantos 1 na linha i e na coluna i
```

Solução (2/2)

```
12     for (int j = 0; j < n; j++) {           // percorre a linha e a coluna ao mesmo tempo
13         if (m[i][j] != 0 && m[i][j] != 1) ok = 0; // só pode ter 0 e 1
14         uns_linha += m[i][j];               // m[i][j]: linha i
15         uns_coluna += m[j][i];              // m[j][i]: coluna i
16     }
17     if (uns_linha != 1 || uns_coluna != 1) ok = 0; // exatamente um 1 por linha e por
    coluna
18 }
19 printf(ok ? "É uma matriz de permutação\n" : "Não é uma matriz de permutação\n"); //
    operador ternário escolhe a mensagem
20 return 0;                                // fim sem erro
21 }
```

Avaliação 3 – 2026.1, Questão I

avaliacao_3_ano_2026_1.pdf

Rotação de 90 graus no sentido horário (na prova, N = 50)

Dicas:

- Desenhe uma matriz 3×3 e a girada: de onde vem cada elemento?
- Linha i da nova = coluna i da original, lida de baixo para cima
- Imprima direto, sem segunda matriz, ou use uma auxiliar

Para conferir: entrada 2 1 2 3 4 $\rightarrow$ a saída deve conter 3 1

A solução comentada está na página do curso: <https://josedihego.net/logica/#aula-30>