Quiz de revisão

Aula 19 – Matrizes e Vetores Multidimensionais

Declarando um Vetor Multidimensional

$$\texttt{<tipo>}\ \texttt{nome}\ [< dim_1 >]\ [< dim_2 >] \ldots [< dim_N >]$$

- Este vetor possui $dim_1 \times dim_2 \times \cdots \times \mathrm{dim}_N$ variáveis do tipo <tipo>
- Cada dimensão é numerada de 0 a $dim_i - 1$

Vetores multi-dimensionais e funções

- Pode-se criar uma função deixando de indicar a primeira dimensão:

```
void mostra_matriz(int mat[][10], int n_linhas) {
  ...
}
```

- Ou pode-se criar uma função indicando todas as dimensões:

```
void mostra_matriz(int mat[5][10], int n_linhas) {
  ...
}
```

- Mas não pode-se deixar de indicar outras dimensões (exceto a primeira):

```
void mostra_matriz(int mat[5][], int n_linhas) {
  //ESTE NÃO FUNCIONA
  ...
}
```

Inicialização de Matrizes

```
int main(){
  int i,j,k;
  int v1[5] = {1,2,3,4,5};
  int v2[2][3] = { {1,2,3}, {4,5,6}};
  int v3[2][3][4] = {
    { {1, 2, 3, 4}, {5, 6, 7, 8}, {9, 10, 11, 12} },
    { {0, 0, 0, 0}, {5, 6, 7, 8}, {0, 0, 0, 0} }
  };

  .
  .
  .
  .
}
```

Linearização de Índices

- Considere o exemplo:
  int mat[12]; // ao invés de int mat[3][4]
- Fazemos a divisão por linhas como segue:
  - Primeira linha: **mat[0]** até **mat[3]**
  - Segunda linha: **mat[4]** até **mat[7]**
  - Terceira linha: **mat[8]** até **mat[11]**
- Para acessar uma posição $[i][j]$ usamos:
  - **mat[i*4 + j];**
    onde $0 \leq i \leq 2$ e $0 \leq j \leq 3$.

Linearização de Índices

- De forma geral, seja matriz **mat[n*m]**, representando **mat[n][m]**.
- Para acessar a posição correspondente à $[i][j]$ usamos:
  - **mat[i*m + j];**
    onde $0 \leq i \leq n-1$ e $0 \leq j \leq m-1$.
- Note que i pula de blocos de tamanho m, e j indexa a posição dentro de um bloco.

Linearização de Índices

- Podemos estender para mais dimensões. Seja matriz **mat[n*m*q]**, representando **mat[n][m][q]**.
  - As posições de 0 até $(m * q) - 1$ são da primeira matriz.
  - As posições de $(m * q)$ até $(2 * m * q) - 1$ são da segunda matriz.
  - Etc...
- De forma geral, seja matriz **mat[n*m*q]**, representando **mat[n][m][q]**.
- Para acessar a posição correspondente à $[i][j][k]$ usamos:
  - **mat[i*m*q + j*q + k];**

Exercícios

- Escreva um programa que lê todos os elementos de uma matriz 4×4 e mostra a matriz e a sua transposta na tela.

$$\begin{array}{cc} \text{Matriz} & \text{Transposta} \\ \begin{bmatrix} 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 2 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 \end{bmatrix} \end{array}$$

Exemplo

Em C temos:

```
void multiplica(double m1[MAX][MAX], double m2[MAX][MAX], double m3[MAX][MAX],
                                                                        int n){
  int i, j, k;

  for(i=0; i<n; i++){
    for(j=0; j<n; j++){
      m3[i][j] = 0;
      for(k=0; k<n; k++){
         m3[i][j] = m3[i][j] + (m1[i][k] * m2[k][j]);
      }
    }
  }

}
```