

Aula Prática 4

Prazo final: 22/04/2018

19 de abril de 2018

Instruções gerais

Use um `scanf` para cada entrada, ou seja, se você tem que ler nove dígitos do teclado, use a função `scanf` nove vezes, uma para cada valor a ser lido.

Não utilize a função `system("pause")` nos arquivos a serem submetidos.

Não utilize mensagens para a entrada dos dados (ex.: "Entre com o numero 1: "), utilize a função `printf` apenas para imprimir as mensagens descritas nas atividades.

1 Matriz simetrica (matsimetrica.c)

Uma matriz simétrica A é uma matriz quadrada tal que $a_{ij} = a_{ji}, \forall i, j$, em outras palavras uma matriz quadrada é simétrica se $A = A^t$. Na tabela 4.2, abaixo, você pode ver um exemplo de uma matriz simétrica:

Tabela 1.1: Exemplo de matriz simétrica.

1	3	-1	2
3	2	0	6
-1	0	-5	1
2	6	1	2

Você deve fazer um programa que leia uma matriz 4x4 e identifique se a matriz lida é simétrica ou não.

- **Entrada:** 16 valores inteiros que representam os valores da matriz A . A leitura deve preencher linha por linha, ou seja, considerando o exemplo acima a entrada seria nessa ordem: 1, 3, -1, 2, 3, 2, 0, 6, -1, 0, -5, 1, 2, 6, 1, 2.
- **Saída:** Uma das seguintes mensagens:
"Matriz simetrica"
"Nao e uma matriz simetrica"

2 Multiplicacao de matrizes (matmul.c)

Faça um programa que leia uma matriz 3 por 3, A , e calcule o resultado da seguinte expressão: $B = A * A^t$, onde A^t representa a matriz transposta de A .

- **Entrada:** 9 valores inteiros. A leitura deve preencher linha por linha (veja o exercício anterior).
- **Saída:** Utilize o bloco de código a seguir para imprimir a matriz B:
Atenção: Note que na função `printf` do código abaixo, ela está usando a variável chamada `matrizB`, supondo que essa seja o nome da variável; Se esse não for o caso, mudar essa linha e incluir o valor da variável correta a ser impressa.

```
int i, j;
for (i=0; i<3; i++){
    for (j=0; j<3; j++){
        printf("%d\t", matrizB[i][j]);
    }
    printf("\n");
}
```

3 Linhas e colunas nulas (nulos.c)

Faça um programa que leia uma matriz 4 por 4, e imprima na tela quantas linhas, e quantas colunas nulas existem na matriz.

Por exemplo, a matriz abaixo:

Tabela 3.1: Matriz exemplo.

1	0	-1	2
3	0	0	6
0	0	0	0
0	0	0	0

Possui 2 linhas nulas, e 1 coluna nula.

- **Entrada:** 16 valores inteiros. A leitura deve preencher linha por linha (veja o primeiro exercício).
- **Saída:** As seguintes mensagens dentro da função `printf`, nessa ordem:
`"Total de colunas nulas: %d\n"`
`"Total de linhas nulas: %d\n"`

4 Movimentação de valores (movimentacao.c)

Faça um programa que leia uma matriz 4 por 4, e após isso, faça uma movimentação de linha e coluna, ou seja, a primeira linha passa a ser a segunda linha, a segunda linha passa a ser a terceira linha, a terceira linha passa a ser a quarta linha, e a quarta linha passa a ser a primeira linha. A mesma movimentação deve ser feita para as colunas. O exemplo abaixo ilustra uma matriz exemplo, e o resultado esperado:
 Por exemplo, a matriz abaixo:

Tabela 4.1: Matriz exemplo, antes da movimentação.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Tabela 4.2: Matriz exemplo, após a movimentação.

16	13	14	15
4	1	2	3
8	5	6	7
12	9	10	11

- **Entrada:** 16 valores inteiros. A leitura deve preencher linha por linha (veja o primeiro exercício).
- **Saída:** Utilize o bloco de código a seguir para imprimir a matriz após a movimentação:

Atenção: Note que na função `printf` do código abaixo, ela está usando a variável chamada `matriz`, supondo que essa seja o nome da variável; Se esse não for o caso, mudar essa linha e incluir o valor da variável correta a ser impressa.

```
int i, j;
for (i=0; i<4; i++){
    for (j=0; j<4; j++){
        printf("%d\t", matriz[i][j]);
    }
    printf("\n");
}
```