

ATENÇÃO	a) Todas as matrizes utilizadas nos exercícios têm que ser declaradas na função <i>main</i> ; b) Não é permitido utilizar variáveis globais. O #define pode ser utilizado.
----------------	--

1) Faça um algoritmo para ler pelo teclado uma matriz 5×4 com valores reais e imprimir o maior valor presente na matriz e a sua posição (linha e coluna). Crie e utilize duas **funções**: uma para a leitura da matriz e outra para imprimir as informações.

2) Faça um algoritmo para ler pelo teclado uma matriz 7×9 com valores reais e imprimir quantos valores maiores do que 20 e menores ou iguais a 50 ela possui. Crie e utilize duas **funções**: uma para a leitura da matriz e outra para fazer o cálculo. A impressão do resultado deve ser na função *main*.

3) Faça um algoritmo para ler pelo teclado uma matriz 5×5 com valores inteiros e também um número inteiro x . Ao final, o algoritmo tem que imprimir se o número x pertence ou não à matriz. O número x tem que ser lido na função *main*. Crie e utilize duas **funções**: uma para a leitura da matriz e outra para realizar a busca. A impressão da informação (se x está presente ou não na matriz) deve ser na função *main*.

4) Faça um algoritmo para ler pelo teclado duas matrizes de números reais 5×3 e imprimir a soma dessas matrizes. Crie e utilize três **funções**: uma para a leitura das matrizes, outra para calcular a soma e uma terceira para imprimir a matriz com o resultado da soma.

5) Faça um algoritmo para imprimir o resultado da multiplicação de um número lido pelo teclado (lido na função *main*) por uma matriz de inteiros 5×15 gerada aleatoriamente com números de 0 até 29. Crie e utilize três **funções**: uma para a geração da matriz, outra para calcular a multiplicação e uma terceira para imprimir a matriz com o resultado da multiplicação.

6) Faça um algoritmo para verificar se uma matriz quadrada 10×10 gerada aleatoriamente com números de 1 até 50 é **simétrica**. Crie e utilize duas **funções**: uma para a geração da matriz e outra para a verificação. De acordo com o retorno da função de verificação, deve-se imprimir na função *main*: “MATRIZ SIMETRICA” ou “MATRIZ NAO SIMETRICA”.

7) Faça um algoritmo para calcular e imprimir a matriz **transposta** de uma matriz 4×9 gerada aleatoriamente com números de 1 até 25. Crie e utilize três **funções**: uma para a geração da matriz, outra para calcular a transposta e uma terceira para imprimir a matriz transposta.

8) Faça um algoritmo para verificar se uma dada matriz quadrada 15×15 gerada aleatoriamente com números de 0 até 19 é uma **matriz identidade**. Crie e utilize duas **funções**: uma para a geração da matriz e outra para a verificação. De acordo com o retorno da função de verificação, deve-se imprimir na função *main*: “MATRIZ IDENTIDADE” ou “MATRIZ NAO E IDENTIDADE”.

9) Faça um algoritmo que, dadas duas matrizes A e B de tamanho 10×10 geradas aleatoriamente com números de 1 até 10, verifique se B é a **inversa** de A, isto é, se B é igual a A^{-1} . Se B for a inversa, a multiplicação $A \times B$ resulta em uma matriz identidade. Crie e utilize três **funções**: uma para verificar se a matriz resultante de $A \times B$ é uma matriz identidade, outra para a geração das matrizes e uma terceira para realizar a multiplicação delas. De acordo com o retorno da função de verificação, deve-se imprimir na função *main*: “B E INVERSA DE A” ou “B NAO E INVERSA DE A”.

10) Faça um algoritmo que verifique se uma dada matriz quadrada 8×8 gerada aleatoriamente com números de 0 até 14 é uma matriz **triangular inferior**. Em uma matriz triangular inferior, todos os elementos acima da diagonal principal são iguais a 0. Os demais elementos podem assumir quaisquer valores. Crie e utilize duas **funções**: uma para a geração da matriz e outra para a verificação. De acordo com o retorno da função de verificação, deve-se imprimir na função *main*: “MATRIZ TRIANGULAR INFERIOR” ou “MATRIZ NAO TRIANGULAR INFERIOR”.