

Processando a Informação: um livro prático de programação independente de linguagem

Rogério Perino de Oliveira Neves

Francisco de Assis Zampirolli

EDUFABC

editora.ufabc.edu.br

Notas de Aulas inspiradas no livro

Utilizando a(s) Linguagem(ns) de Programação:

C

Exemplos adaptados para Correção Automática no Moodle+VPL

Francisco de Assis Zampirolli

3 de setembro de 2022

Sumário

0.1	Processando a Informação: Cap. 6: Matrizes - Prática 3	2
0.1.1	Exercícios	2

0.1 Processando a Informação: Cap. 6: Matrizes - Prática 3



Este caderno (Notebook) é parte complementar *online* do livro **Processando a Informação: um livro prático de programação independente de linguagem**, que deve ser consultado no caso de dúvidas sobre os temas apresentados.

Este conteúdo pode ser copiado e alterado livremente e foi inspirado nesse livro.

0.1.1 Exercícios

Fonte: [ref1](#); [ref2](#)

1. Uma matriz quadrada $N \times N$ lógica representa as posições minadas de um jogo. Quando uma posição possui o valor VERDADEIRO significa que há uma mina ali. Escreva um programa que informa se é possível percorrer o tabuleiro de um lado ao outro em linha reta (atravessando uma linha inteira ou coluna inteira) ou diagonal sem passar por uma mina sequer.

[]: `# escreva o seu código`

2. Uma imagem em níveis de cinza pode ser representada por uma matriz. Leia uma imagem A de números inteiros. Leia também um limiar inteiro. Fazer um método para calcular e retornar uma imagem binária de saída B com as mesmas dimensões de A, considerando para cada pixel $A(i,j) > \text{limiar}$, $B(i,j)$ deve receber o valor 1, caso contrário recebe o valor 0. Imprimir as imagens de entrada e saída.

[]: *# escreva o seu código*

3. Leia uma matriz A de inteiros. Criar uma matriz B considerando que cada elemento de B é o resultado do elemento de A multiplicado pela média dos elementos da linha deste elemento. Arredondar cada elemento com o comando `round`.

[]: *# escreva o seu código*

4. Faça um programa para gerar automaticamente números entre 0 e 99 de uma cartela de bingo. Sabendo que cada cartela deverá conter 5 linhas de 5 números, gere estes dados de modo a não ter números repetidos dentro das cartelas. O programa deve exibir na tela a cartela gerada.

[]: *# escreva o seu código*

5. Leia uma matriz 5 x 10 que se refere respostas de 10 questões de múltipla escolha, referentes a 50 candidatos de um processo seletivo. Leia também um vetor de 10 posições contendo o gabarito de respostas que podem ser a, b, c, d ou e. Seu programa deverá comparar as respostas de cada candidato com o gabarito e emitir um vetor denominado resultado, contendo a pontuação correspondente a cada candidato. Imprimir também a lista de aprovados e em seguida a lista de reprovados, considerando média 5.

[]: *# escreva o seu código*

6. Leia uma matriz A de inteiros. Qual é o maior produto de quatro números adjacentes em qualquer direção (cima, baixo, esquerda, direita, ou nas diagonais) na matriz de 20x20 e em qual(is) coordenada(s) ocorreu(ram)? Ou seja, em cada posição [i,j] de A calcular o máximo valor entre as multiplicações dos elementos abaixo:

vizinhança		
[i-1,j-1]	[i-1,j]	[i-1,j+1]
[i ,j-1]	[i ,j]	[i ,j+1]
[i+1,j-1]	[i+1,j]	[i+1,j+1]

[]: *# escreva o seu código*

7. Uma matriz de caracteres 3x3 foi utilizada para armazenar uma partida de jogo da velha. Os caracteres 'O' e 'X' foram utilizados para armazenar a jogada de cada participante. Informe na tela se o vencedor foi o jogador 'O', o jogador 'X' ou se

o resultado foi empate. IMPORTANTE: não serão informadas partidas com dois vencedores, apenas partidas válidas e todas as 9 casas estarão preenchidas com 'O' ou 'X'.

[]: *# escreva o seu código*

-
8. Faça um programa para determinar a próxima jogada em um Jogo da Velha. Assumir que o tabuleiro é representado por uma matriz de 3 x 3, onde cada posição representa uma das casas do tabuleiro. A matriz pode conter os seguintes valores -1, 0, 1 representando respectivamente uma casa contendo uma peça minha (-1), uma casa vazia do tabuleiro (0), e uma casa contendo uma peça do meu oponente (1).

[]: *# escreva o seu código*