

QUARTA LISTA DE EXERCÍCIOS

1. Faça um programa que leia um valor  $n$  e depois leia uma sequência de  $n$  números inteiros e os armazene em um vetor. Posteriormente, seu programa deve determinar o maior e menor elemento deste vetor.
2. Escreva um programa que inverta um vetor  $v$ , isto é, troca o valor  $v[1]$  com  $v[n]$ ,  $v[2]$  com  $v[n-1]$ , etc.
3. Escreva um programa que leia duas palavras do teclado e determina se a segunda é um *anagrama* da primeira. Uma palavra é um anagrama de outra se todas as letras de uma ocorrem na outra, *em mesmo número, independente da posição*. Exemplos: ROMA, MORA, ORAM, AMOR, RAMO são anagramas entre si.
4. Escreva um programa que leia uma palavra do teclado e determine se esta é um palíndromo.
5. Escreva um programa que leia uma sequência de  $n$  inteiros e determine quantas sequências isoladas de números iguais existem. Exemplo: para  $n = 13$ ,

3 4 4 1 2 5 5 5 2 2 6 2 2

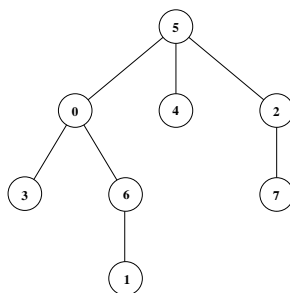
há 4 grupos de 1 elemento (3, 1, 2 e 6), 3 grupos de 2 elementos (4, 2 e 2) e 1 grupo de 3 elementos (5). O valor  $n$  deve ser um dado de entrada.

6. Dadas duas sequências de  $n$  e  $m$  elementos binários (0 ou 1), onde  $n \leq m$ , faça um programa que verifique quantas vezes a primeira sequência ocorre na segunda. Ex: se a primeira for 101 e a segunda 1101010011010, então o resultado deve ser 3.
7. (*Busca sequencial*) Faça um programa que leia uma sequência de números positivos terminada com o número 0 e os armazene em um vetor. Depois disso, seu programa deve solicitar um número ao usuário e pesquisar se o número digitado ocorre no vetor. Seu programa deve implementar o algoritmo de busca sequencial.
8. (*Busca binária*) O programa deste exercício é essencialmente o mesmo que o anterior. Entretanto, neste exercício você deve implementar o algoritmo de busca binária (portanto, é necessário que você ordene o seu vetor antes de fazer a busca).
9. (*Busca sequencial versus busca binária*) Considere os dois últimos programas. Suponha que você queira pesquisar a ocorrência do número  $x$ . O objetivo deste exercício é contar o número de *testes/comparações* de  $x$  com os elementos do vetor que cada um dos programas precisa fazer para tomar uma decisão. Para isso, em cada um dos programas, acrescente um contador para determinar este número.

Compare o desempenho de ambos os programas, considerando o número de comparações.

10. Escreva um programa que leia do teclado uma frase e a coloque em um vetor de caracteres. Entre as palavras que compõem a frase pode haver um ou mais brancos ou os sinais de pontuação , e .. O programa deve escrever na tela as palavras que compõem a frase, uma por linha.

11. Em uma empresa onde trabalham  $n$  pessoas, cada pessoa recebe um número de identificação único, que varia de 0 a  $n - 1$ . Além disso, esta empresa encontra-se organizada em níveis hierárquicos, de forma que cada pessoa possui exatamente um chefe, exceto o presidente da empresa, que se encontra no topo da organização e que, portanto, não possui chefe. Por exemplo, a figura abaixo representa a hierarquia de uma empresa com 8 pessoas, cada pessoa sendo representada por um círculo com seu número de identificação dentro. As retas ligam cada pessoa ao seu chefe no nível superior. Por exemplo, a pessoa 3 tem como chefe a pessoa 0. Por sua vez, a pessoa 0 tem como chefe a pessoa 5. A pessoa 5 não possui chefe e é, portanto, o presidente.



As informações de chefia podem ser resumidas em tabelas, como exemplificado abaixo. Note que como o presidente não possui chefe, o “chefe do presidente” está representado como -1.

Pessoa	0	1	2	3	4	5	6	7
Chefe	5	6	5	0	5	-1	0	2

Com as informações de chefia, podemos determinar o nível de cada pessoa, onde o presidente está no nível 1, as pessoas logo abaixo (0, 4 e 2) estão no nível 2, as pessoas abaixo delas (3, 6 e 7) estão no nível 3 e assim em diante.

De forma resumida, temos:

Pessoa	0	1	2	3	4	5	6	7
Chefe	2	4	2	3	2	1	3	3

Sua tarefa é determinar o nível de cada pessoa na empresa.

Deve então ser lido um número  $n$  indicando quantas pessoas há em uma empresa, seguido de  $n$  valores inteiros positivos, indicando os chefes de cada pessoa, em ordem crescente de números de identificação, como na segunda linha da primeira tabela. Na saída, seu programa deve calcular e exibir, em ordem de número de identificação, o nível de cada pessoa, como na segunda linha da segunda tabela.

12. Ainda considerando o estrutura hierárquica do exercício anterior, considere que, para certas operações, é preciso que um documento de autorização seja assinado pelo presidente. Entretanto, por questões burocráticas, sempre que uma pessoa precisa de uma autorização desse tipo, ela deve fazer a solicitação primeiro para o seu próprio chefe, que irá repassá-la para o nível de cima, se houver, até que em algum momento a requisição chegue ao presidente.

Por exemplo, no caso do exemplo do exercício anterior, se a pessoa 1 necessitar de uma autorização, ela deve pedir ao seu chefe, a pessoa 6. Em seguida, a pessoa 6 irá repassar a solicitação para o seu próprio chefe, a pessoa 0. A pessoa 0 enviará a solicitação para seu chefe, a pessoa 5. Como a pessoa 5 é o próprio presidente, ela poderá assinar a autorização.

Em cada passo desse processo, perde-se um pouco de tempo, o qual depende da rapidez e eficiência de cada pessoa em passar a requisição para seu chefe ou, no caso do presidente, em assinar a autorização.

No caso do exemplo do exercício anterior, considere que o tempo que o documento fica com cada pessoa está representado na tabela abaixo.

Pessoa	0	1	2	3	4	5	6	7
Chefe	5	4	10	8	15	4	9	12

Desta forma, podemos então determinar o tempo total necessário até que a autorização solicitada por uma dada pessoa seja assinada. Por exemplo, se a pessoa 0 inicia a solicitação, ela leva um tempo igual a 5 para repassá-la para seu chefe, o presidente 5, que por sua vez, leva um tempo igual a 4 para assiná-la. Portanto, solicitações iniciadas pela pessoa 0, levam um tempo total de  $5 + 4 = 9$ . Já no caso em que a solicitação é iniciada pela pessoa 7, a mesma leva um tempo igual a 12 para repassá-la para seu chefe, a pessoa 2. A pessoa 2 leva um tempo igual a 10 para repassar o documento ao presidente 5, que leva mais um tempo igual a 4 para assinar a autorização. Dessa forma, o tempo total para que uma autorização requisitada pela pessoa 7 seja assinada é de  $12 + 10 + 4 = 26$ .

Determinando este tempo total para todas as pessoas (cada pessoa é vista como um solicitante), teríamos:

Pessoa	0	1	2	3	4	5	6	7
Chefe	9	22	14	17	19	4	18	26

Sua tarefa é determinar este tempo total, por solicitante.

13. Escreva um programa que, dada uma matriz de  $m$  linhas e  $n$  colunas, verifique se esta é simétrica.
14. Uma matriz de elementos inteiros é um *quadrado mágico* se a soma dos elementos de cada linha, a soma dos elementos de cada coluna e a soma dos elementos das diagonais principal e secundária são todas iguais. Exemplo: a matriz abaixo é um quadrado mágico.

$$A = \begin{pmatrix} 8 & 0 & 7 \\ 4 & 5 & 6 \\ 3 & 10 & 2 \end{pmatrix}$$

Escreva um programa que verifique se uma matriz de  $n$  linhas e  $n$  colunas é um quadrado mágico.

15. Dado um elemento  $mat[i][j]$  de uma matriz  $mat$ , dizemos que os elementos *adjacentes* a  $mat[i][j]$  são  $mat[i-1][j-1]$ ,  $mat[i-1][j]$ ,  $mat[i-1][j+1]$ ,  $mat[i][j-1]$ ,  $mat[i][j+1]$ ,  $mat[i+1][j-1]$ ,  $mat[i+1][j]$  e  $mat[i+1][j+1]$ . Note que  $mat[1][1]$  tem somente três elementos adjacentes, o mesmo acontecendo com  $mat[1][n]$ ,  $mat[n][1]$  e  $mat[n][n]$ . Analogamente, os outros elementos da primeira e última linha e coluna de  $mat$  tem somente 5 elementos adjacentes. Todos os outros elementos de  $mat$  tem 8 elementos adjacentes. Escreva um programa que leia uma matriz  $mat$  de números inteiros, de  $m$  linhas e  $n$  colunas e produza uma matriz  $mat1$ , também de  $m$  linhas e  $n$  colunas, tal que  $mat1[i][j]$  contenha a média dos elementos adjacentes a  $mat[i][j]$ .

16. Dada a matriz

$$mat = \begin{pmatrix} 1 & 0 & 2 \\ 2 & 2 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

(a) Execute o comando (faça sem o computador):

```
for (i=0;i<3;i++)
  for (j=1;j<3;j++)
    mat1[i][j] = mat[mat[i][j]][mat[j][i]];
```

Supondo que todas as variáveis tenham sido declaradas corretamente, responda às perguntas:

(b) Qual o valor da variável **mat1** resultante?

(c) Substitua no comando a variável **mat1** por **mat**. Qual o valor de **mat** resultante?

17. Escreva um programa que, dada uma matriz de  $m$  linhas e  $n$  colunas, verifique se esta é simétrica.

18. A matriz abaixo representa o Triângulo de Pascal de ordem 6:

$$\begin{array}{cccccc} 1 & & & & & \\ 1 & 1 & & & & \\ 1 & 2 & 1 & & & \\ 1 & 3 & 3 & 1 & & \\ 1 & 4 & 6 & 4 & 1 & \\ 1 & 5 & 10 & 10 & 5 & 1 \end{array}$$

Os elementos extremos de cada linha são iguais a 1. Os outros elementos são obtidos somando-se os dois elementos que aparecem imediatamente acima e à esquerda na linha anterior. Assim,  $10 = 4 + 6$ .

Escreva três versões de um programa que, dado  $n$ , construa e imprima o Triângulo de Pascal de ordem  $n$ , utilizando:

- (a) Uma matriz
- (b) Dois vetores
- (c) Apenas um vetor

19. Uma matriz de permutações é uma matriz quadrada cujos elementos são zeros ou uns tal que em cada linha e em cada coluna exista *exatamente* um elemento igual a 1. A matriz abaixo representa uma matriz de permutações de ordem 3:

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Escreva um programa que determine se uma matriz lida do teclado é uma matriz de permutações.

20. Uma matriz triangular inferior preenchida com valores inteiros não negativos é denominada de *mapa triangular*. Um mapa triangular é utilizado em uma certa brincadeira, na qual os participantes caminham ao longo das casas, recolhendo prêmios ao longo do trajeto. O valor do prêmio coletado corresponde ao número no interior de cada posição do mapa.

No início do jogo, o participante se encontra na posição  $(0, 0)$  e a cada rodada deve se locomover para uma casa na linha imediatamente abaixo da sua posição atual. Entretanto, somente dois movimentos são permitidos. Um deles consiste em o jogador se mover para a posição de coluna igual a atual, na linha de baixo. O outro corresponde a se mover para a posição uma coluna à direita da posição atual na linha de baixo. Isto é, um participante que se encontre na posição  $(i, j)$  pode optar por se mover para a posição  $(i + 1, j)$  ou para a posição  $(i + 1, j + 1)$ .

A brincadeira se encerra quando o participante atinge qualquer posição na última linha.

Sua tarefa é elaborar um programa que leia um mapa triangular e retorne a quantidade máxima de prêmios que um participante pode coletar. Além disso, o seu programa deve informar o caminho, partindo da posição inicial, que leva à coleta máxima de prêmios.