

Lista de Exercícios

1. Para se determinar o número de lâmpadas necessárias para cada cômodo de uma residência, existem normas que fornecem o mínimo de potência de iluminação exigida por metro quadrado (m^2) conforme a utilização deste cômodo. Suponha que só serão usadas lâmpadas de 60W.

Seja a seguinte tabela:

Utilização	Classe	Potência/ m^2
quarto	1	15
sala de TV	1	15
salas	2	18
cozinha	2	18
varandas	2	18
escritório	3	20
banheiro	3	20

- (a) Faça um módulo (função ou um procedimento) que receba a classe de iluminação de um cômodo e suas duas dimensões e devolve o número de lâmpadas necessárias para o cômodo.
 - (b) Faça um algoritmo que leia um número indeterminado de informações, contendo cada uma o nome do cômodo da residência, sua classe de iluminação e as suas duas dimensões e, com base no módulo anterior, imprima a área de cada cômodo, sua potência de iluminação e o número total de lâmpadas necessárias. Além disso, seu algoritmo deve calcular o total de lâmpadas necessárias e a potência total para a residência.
2. A comissão organizadora de um *rallye* automobilístico decidiu apurar os resultados da competição através de um processamento eletrônico. Um dos programas necessários para a classificação das equipes é o que emite uma listagem geral do desempenho das equipes, atribuindo pontos segundo determinadas normas.
 - (a) Escreva um módulo (função ou procedimento) que calcula os pontos de cada equipe em cada uma das etapas do *rallye*, seguindo o seguinte critério. Seja Δ o valor absoluto da diferença entre o tempo-padrão e o tempo despendido pela equipe numa etapa (fornecidos como parâmetros):

$\Delta < 3$ minutos	atribuir 100 pontos à etapa
$3 \leq \Delta \leq 5$ minutos	atribuir 80 pontos à etapa
$\Delta > 5$	atribuir $80 - \frac{\Delta-5}{5}$ pontos à etapa

- (b) Faça um algoritmo que leia os tempos-padrão (em minutos decimais) para as três etapas da competição, leia um número indeterminado de informações, contendo para cada equipe o seu número de inscrição e os tempos (em minutos decimais) despendidos para cumprir as três etapas e, utilizando o módulo anterior, calcule os pontos obtidos por cada equipe em cada etapa, a soma total de pontos por equipe, e a equipe vencedora.
3. (a) Faça uma função *quantosdias* que recebe o dia, o mês e o ano de uma data e retorna um valor que contém o número de dias do ano até a data fornecida.
- (b) Faça um algoritmo que recebe n pares de datas, onde cada par deve ser fornecido no formato *dia1, mês1, ano1, dia2, mês2, ano2*, verifique se as datas estão corretas e mostre a diferença, em dias, entre essas duas datas. Utilize a função anterior para o cálculo do total de dias de cada data.
4. (a) Escreva uma função que recebe dois números inteiros positivos e determina o produto dos mesmos, utilizando o seguinte método de multiplicação:
- dividir, sucessivamente, o primeiro número por 2, até que se obtenha 1 como quociente;
 - paralelamente, dobrar, sucessivamente, o segundo número;
 - somar os números da segunda coluna que tenham um número ímpar na primeira coluna. O total obtido é o produto procurado.
- Exemplo: 9×6
- $$\begin{array}{r}
 9 \quad 6 \quad \rightarrow \quad 6 \\
 4 \quad 12 \\
 2 \quad 24 \\
 1 \quad 48 \quad \rightarrow \quad 48 \\
 \hline
 54
 \end{array}$$
- (b) Escreva um programa que leia n pares de números e calcule os respectivos produtos, utilizando a função anterior.
5. Um número a é dito ser *permutação* de um número b se os dígitos de a formam uma permutação dos dígitos de b .

Exemplo:

5412434 é uma permutação de 4321445, mas não é uma permutação de 4312455.

Observação: considere que o dígito 0 (zero) não aparece nos números.

- (a) Faça uma função *contadígitos* que, dados um inteiro n e um inteiro d , $0 < d \leq 9$, devolve quantas vezes o dígito d aparece em n ;
- (b) Utilizando a função do item anterior, faça um algoritmo que leia dois números a e b e responda se a é permutação de b .

6. Um número b é dito ser *sufixo* de um número a se o número formado pelos últimos dígitos de a são iguais a b .

Exemplo:

a	b		
567890	890	→	sufixo
1234	1234	→	sufixo
2457	245	→	não é sufixo
457	2457	→	não é sufixo

- (a) Construa uma função *sufixo* que dados dois números inteiros a e b verifica se b é um sufixo de a .
- (b) Utilizando a função do item anterior, escreva um algoritmo que leia dois números inteiros a e b e verifica se o menor deles é subsequência do outro.

Exemplo:

a	b		
567890	678	→	b é subsequência de a
1234	2212345	→	a é subsequência de b
235	236	→	Um não é subsequência do outro

7. Uma seqüência de n números inteiros não nulos é dita m -alternante se é constituída por m segmentos: o primeiro com um elemento, o segundo com dois elementos e assim por diante até a m -ésima, com M elementos. Além disso, os elementos de um mesmo segmento devem ser todos pares ou todos ímpares e para cada segmento, se seus elementos forem todos pares (ímpares), os elementos do segmento seguinte devem ser todos ímpares (pares).

Por exemplo:

A seqüência com $n = 10$ elementos: 8 3 7 2 10 4 5 13 4 11 é 4-alternante.

A seqüência com $n = 3$ elementos: 7 2 8 é 2-alternante.

A seqüência com $n = 8$ elementos: 1 12 4 2 13 5 12 6 não é alternante, pois o último segmento não tem tamanho 4.

- (a) Escreva uma função *bloco* que recebe como parâmetro um inteiro n e lê n inteiros do teclado, devolvendo um dos seguintes valores:
- 0, se os n números lidos forem pares;
 - 1, se os n números lidos forem ímpares;
 - 1, se entre os n números lidos há números com paridades diferentes.
- (b) Utilizando a função do item anterior, escreva um algoritmo que, dados um inteiro n ($n \geq 1$) e uma seqüência de n números inteiros, verifica se ela é m -alternante. O algoritmo deve imprimir o valor de m ou dar uma resposta dizendo que a seqüência não é alternante.

8. Considere as seguintes fórmulas de recorrência:

$$\left\{ \begin{array}{l} F_1 = 2; \\ F_2 = 1; \\ F_i = 2 * F_{i-1} + G_{i-2} \quad i \geq 3. \end{array} \right. \quad \left\{ \begin{array}{l} G_1 = 1; \\ G_2 = 2; \\ G_i = G_{i-1} + 3 * F_{i-2} \quad i \geq 3. \end{array} \right.$$

Podemos então montar a seguinte tabela:

i	1	2	3	4	5	...
F_i	2	1	3	8	24	...
G_i	1	2	8	11	20	...

Este exercício está dividido em três partes.

- (a) Só para ver se você entendeu as fórmulas, qual é o valor de F_6 e G_6 ?
- (b) Faça um procedimento de nome *valor* que recebe um inteiro $K \geq 1$ e devolve F_k e G_k .

Exemplo:

Para $k = 2$, o procedimento deve retornar os valores 1 e 2. Para $k = 3$, a função deve devolver os valores 3 e 8. Para $k = 4$, a função deve devolver os valores 8 e 11.

Observação: não utilize vetores neste exercício.

- (c) Faça um algoritmo que lê um inteiro $n > 2$ e imprime os valores $F_{n-2} + G_{n+200}$ e $F_{n+200} + G_{n-2}$. Seu algoritmo deve obrigatoriamente utilizar o procedimento do item anterior.

9. Um conjunto pode ser representado por um vetor da seguinte forma: $V[0]$ é o tamanho do conjunto; $V[1], V[2], \dots$ são os elementos do conjunto (sem repetições).

- (a) Faça uma função *intersecção* que dados dois conjuntos de números inteiros A e B , constrói um terceiro conjunto C que é a intersecção de A e B . Lembre-se de que em $C[0]$ a sua função deve colocar o tamanho da intersecção.

- (b) Faça um algoritmo que leia um inteiro $n \geq 2$ e uma seqüência de n conjuntos de números inteiros (cada um com no máximo 100 elementos) e construa e imprima o vetor INTER que representa a intersecção dos n conjuntos.

NOTE que NÃO é preciso ler todos os conjuntos de uma só vez. Você pode ler os dois primeiros conjuntos e calcular a primeira intersecção. Depois, leia o próximo conjunto e calcule uma nova intersecção entre esse conjunto lido e o conjunto da intersecção anterior, e assim por diante.

10. (a) Escreva uma função que recebe como parâmetros um vetor real A com n elementos e um vetor B com m elementos, ambos representando conjuntos, e verifica se A está contido em B ($A \subset B$).
- (b) Utilizando a função anterior verifique se dois conjuntos são iguais ($A = B$ se e somente se $A \subset B$ e $B \subset A$).
11. (a) Escreva uma função que troca o conteúdo de duas variáveis.
- (b) Escreva uma função que recebe dois inteiros, i e j , uma matriz real $A_{m \times n}$ e troca a linha i pela linha j . Utilize o procedimento do item anterior.
12. Dizemos que uma matriz $A_{n \times n}$ é um *quadrado latino* de ordem n se em cada linha e em cada coluna aparecem todos os inteiros $1, 2, 3, \dots, n$ (ou seja, cada linha e coluna é permutação dos inteiros $1, 2, \dots, n$).

Exemplo:

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 4 & 1 & 2 & 3 \\ 3 & 4 & 1 & 2 \end{pmatrix}$$

A matriz acima é um quadrado latino de ordem 4.

- (a) Escreva uma função que recebe como parâmetro um vetor inteiro V com n elementos e verifica se em V ocorrem todos os inteiros de 1 a n .
- (b) Escreva uma função que recebe como parâmetros uma matriz inteira $A_{n \times n}$ e um índice j e verifica se na coluna j de A ocorrem todos os inteiros de 1 a n .
- (c) Utilizando as funções acima, verifique se uma dada matriz inteira $A_{n \times n}$ é um quadrado latino de ordem n .