UNIVERSIDADE SÃO FRANCISCO ALGORITMOS E LINGUAGEM DE PROGRAMAÇÃO

LISTA DE EXERCÍCIOS 02

Algoritmos aplicados a uma linguagem. Estruturas de Decisão. Laços. Arrays Uni e Multidimensionais

- 1. Implementar um algoritmo capaz de encontrar o maior dentre 3 números inteiros quaisquer. Suponha todos serem distintos.
- 2. Implementar um algoritmo que leia 3 números quaisquer e os imprima em ordem crescente.
- 3. Implementar um algoritmo que leia um conjunto de 50 elementos inteiros e os imprima em ordem contrária da que foi lida (DICA: use um vetor).

421	37	82	 23	
1	2	3	50	

Ex. conjunto lido :

impressão : 23, ..., 82, 37, 421.

4. Implementar um algoritmo que calcule e escreva o somatório dos valores armazenados numa variável unidimensional A de 100 elementos numéricos a serem lidos do teclado.

Somatória = 32 + 17 + 10.7 + ... + 15.8 = 75.5 =
$$\sum_{i=1}^{100} a_i$$

- 5. Implementar um algoritmo capaz de ler uma variável de 100 elementos numéricos e verificar se existem elementos iguais a 30. Se sim, escrever as posições em que estão armazenados.
- 6. Dada uma seqüência numérica de N elementos armazenados num vetor, determinar o índice (posição) do maior elemento do conjunto. Suponha que os elementos sejam distintos.
- 7. Dado um vetor com N elementos numéricos reais positivos obter a maior diferença entre dois elementos consecutivos neste vetor.
- 8. Implementar um algoritmo para transpor matrizes NxM. Transpor uma matriz significa transformar suas linhas em colunas e vice-versa. Ex.:

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}^t = \begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix}$$

Implementar um algoritmo para somar matrizes.
Obs.: as matrizes obrigatoriamante têm a mesma dimensão. Ex.:

$$\begin{bmatrix} a_{11} & \dots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \dots & a_{nm} \end{bmatrix}_{NxM} + \begin{bmatrix} b_{11} & \dots & b_{1m} \\ \vdots & & \vdots \\ b_{n1} & \dots & b_{nm} \end{bmatrix}_{NxM} = \begin{bmatrix} (a_{11} + b_{11}) & \dots & (a_{1m} + b_{1m}) \\ \vdots & & \vdots \\ (a_{n1} + b_{n1}) & \dots & (a_{nm} + b_{nm}) \end{bmatrix}_{NxM}$$

- Fazer um algoritmo que leia uma matriz inteira de ordem 4 e verifique se a soma dos elementos das diagonais s\u00e3o iguais.
- 11. Dados dois conjuntos de números reais <u>ordenados</u> com 100 elementos do tipo inteiro, montar um terceiro conjunto contendo os elementos da intersecção dos conjuntos originais.
- 12. Tremblay, Jean Paul; "Ciência dos Computadores"

(OBS.: faca uma versão dos algoritmos em Pascal, testando as respostas)

- a) pg 132 ex 1, 2, 5, 6, 8
- b) pg 208 ex 7, 9

-*- DESAFIOS -*-

1. Tremblay, Jean Paul; "Ciência dos Computadores"

(OBS.: faça uma versão dos algoritmos em Pascal, testando as respostas)

- a) pg 166 ex 4
- b) pg 216 implementar o algoritmo de diagramação.

LISTA DE EXERCÍCIOS - parte 2 Aplicação em problemas reais

A) Exercícios usando de estruturas de repetição e decisão

- Num frigorífico existem 90 bois. Cada boi traz preso no seu pescoço um cartão contendo um número de identificação e seu peso. Implementar um algoritmo que escreva o número e o peso do boi mais gordo e do boi mais magro (não é necessário armazenar os dados de todos os bois).
- 2. Foi feita uma pesquisa de audiência de TV em várias casas de uma certa cidade, num determinado dia. Para cada casa visitada, o entrevistador (munido de um "notebook") escolhia num menu qual o canal que estava sendo assistido (Cultura, SBT, Globo, Record, MTV, TVA, Manchete, Bandeirantes) e o número de pessoas que estavam assistindo TV. Se o TV estivesse desligado, nada era anotado, ou seja, esta casa não entrava na pesquisa. Implementar uma versão do algoritmo que o entrevistador está usando no seu "notebook", e que:
 - a)leia um número indeterminado de dados, terminando quando o entrevistador escolher a opção Fim
 - b) calcule e escreva a percentagem de audiência para cada emissora.
 - c) identifique a maior e a menor audiência da pesquisa.

DICA: implemente um "menu" de opções em sua tela.

- 3. Deseja-se fazer um levantamento a respeito da ausência de alunos à segunda prova de I.C.C. para cada uma das 14 turmas existentes. Para cada turma é fornecido um conjunto de valores, sendo que os dois primeiros valores do conjunto correspondem à identificação da turma (A, B, C, ...) e ao número de alunos matriculados, e os demais valores deste conjunto correspondem ao número de matrícula do aluno e à letra A ou P, para o caso de o aluno estar ausente ou presente, respectivamente. Implementar um algoritmo que:
 - a. para cada turma, calcule a porcentagem de ausência e escreva a identificação da turma e a porcentagem calculada;
 - b. determine e escreva quantas turmas tiveram porcentagem de ausência superior a 5%.
- 4. Foi feita uma pesquisa para determinar o índice de mortalidade infantil em um certo período. Implementar um algoritmo que:
 - a) leia inicialmente o número de crianças nascidas no período:
 - b) leia, em seguida, um número indeterminado de linhas contendo, cada uma, o sexo de uma criança morta (masc, fem) e o número de meses de vida da criança. A última linha (que identificará o final da entrada de dados) conterá a palavra 'VAZIO' no lugar do sexo.
 - c) determine e imprima:
 - c1) a porcentagem de crianças mortas no período;

- c2) a porcentagem de crianças do sexo masc. mortas no período:
- c3) a porcentagem de crianças que viveram 24 meses ou menos no período.
- 5. Implementar um algoritmo para calcular o número de dias decorridos entre duas datas (considerar também a ocorrência de anos bissextos), sabendo que:
 - cada par de datas é lida numa linha, a última linha contém o número do dia negativo (indicando que terminou o cálculo);
 - a primeira data lida é sempre a mais antiga.
 - O ano será digitado com 4 dígitos.
- 6. Implementar um algoritmo que leia os caracteres a partir do teclado, um a um, imprimindo-os na tela, como se fosse uma máquina de escrever. O programa deve terminar a execução quando a tecla <ESC> (caracter ASCII 27) for pressionada, imprimindo então o número de vogais, espaços em branco e <ENTER> digitados, oferecendo uma estatística sobre o total de teclas pressionadas.

B) Exercícios usando Tipos Estruturados (array, string, registros e conjuntos)

- 1. Implementar um algoritmo que leia 150 notas de provas e os respectivos nomes dos indivíduos (até 30 caracteres) e:
 - calcule a média das notas;
 - calcule quantas notas estão acima e abaixo da média (não considerar notas iguais à média);
 - quantas pessoas possuem a maior e a menor das notas, e quais são elas (isto é, seus nomes)
- 2. Implementar um algoritmo para corrigir provas de múltipla escolha. Cada prova tem 10 questões e cada questão vale 1 ponto. O primeiro conjunto de dados a ser lido será o gabarito para a correção da prova. Os outros dados serão os números dos alunos e suas respectivas respostas, e o último número, do aluno fictício, 9999, indicando que não há mais alunos. O programa deve calcular e imprimir:
 - a) para cada aluno, seu número e sua nota;
 - b) a porcentagem de aprovação, sabendo-se que a nota mínima de aprovação é 6;
 - c) a nota que teve a maior freqüência absoluta, ou seja, a nota que apareceu mais vezes (supondo a inexistência de empates).

Dicas: use o esquema de estruturas abaixo:

	Gab	arito								Nullielo	Nota
	Res	postas								Aprovados	Total
Freqüência									Maior F	orcentagem	

- 3. Uma pesquisa sobre algumas características físicas da população de uma determinada região coletou os seguintes dados, referentes a cada habitante, para serem analisados:
 - sexo (masculino, feminino)
 - cor dos olhos (azuis, verdes, castanhos)
 - cor dos cabelos (louros, castanhos, pretos)
 - idade em anos.

Para cada habitante, foi perfurado um cartão com esses dados, e o último cartão, que não corresponde a ninguém, conterá o valor da idade igual a -1. Implementar um algoritmo que determine e escreva:

- a) a maior idade dos habitantes;
- b) porcentagem de indivíduos do sexo feminino cuja idade esteja entre 18 e 35 anos, inclusive, e que tenham olhos verdes e cabelos louros (Tá bom assim?).

DICA: use o tipo **ESTRUTURA** (struct): 1 ficha = 1 registro.

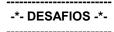
- 4. Implementar um algoritmo que (use registros):
 - a) leia e escreva o número e as medidas (busto, cintura, quadril) das moças inscritas num concurso de beleza. Para cada moça existe um cartão, contendo o seu número e seus dados. O último cartão, que não corresponde a nenhuma moça, conterá o número 0 e valores quaisquer;
 - b) calcule e escreva as duas maiores alturas e quantas moças as possuem.
 - c) descubra qual a moça com as medidas mais próximas de (busto=90, cintura=60, quadril=90). PS.: se possível, descubra seu endereço e telefone e avise o pessoal!
- 5. Implementar um algoritmo para multiplicar duas matrizes de números inteiros. A multiplicação só é possível se o número de colunas da matriz A for igual ao número de linhas da matriz B. Suponha um tamanho máximo igual a uma matriz 10x10, sendo que o usuário entrará com o tamanho e os valores de cada uma das matrizes a serem multiplicadas. (DICA: Veja Tremblay, pg 167, ex 3) Ex.:

$$\begin{bmatrix} a_{11} & \dots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \dots & a_{nm} \end{bmatrix}_{NxM} \times \begin{bmatrix} b_{11} & \dots & b_{1p} \\ \vdots & & \vdots \\ b_{m1} & \dots & b_{mp} \end{bmatrix}_{MxP} = \begin{bmatrix} (a_{11} \times b_{11}) + \dots & (a_{11} \times b_{1p}) + \dots \\ + (a_{1m} \times b_{m1}) & \dots & + (a_{1m} \times b_{mp}) \\ \vdots & & \vdots & & \vdots \\ (a_{n1} \times b_{11}) + \dots & & (a_{n1} \times b_{1p}) + \dots \\ + (a_{nm} \times b_{m1}) & \dots & + (a_{nm} \times b_{mp}) \end{bmatrix}_{NxP}$$

- Implementar um algoritmo para multiplicar uma matriz por um vetor. Use as restrições citadas no exercício 5, lembrando agora que a matriz B será um vetor-coluna, ou seja, uma matriz de dimensão B_{Mx1}
- 7. Tremblay, Jean Paul; "Ciência dos Computadores"

(OBS.: faça uma versão dos algoritmos em Pascal, testando as respostas)

- a) pg 180 implementar o algoritmo do exemplo 4.5.4
- b) pg 208 6
- c) pg 230 2



1. Implementar um algoritmo para calcular uma tabela com o sen(x), com x variando de 0 a 1,6 radianos, de décimo em décimo de radiano, usando a série

sen
$$x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots$$

com erro inferior a 0,0001. Imprimir também o número de termos usados.

Implementar um algoritmo que calcule e imprima os números primos existentes entre 1000 e 7000.
Obs.: número primo é todo aquele que só é divisível por 1 e por ele mesmo. Ex. 2, 3, 5, 7, 11, ... (o número 1 não é considerado primo)

DICA: use o método chamado "Crivo de Eratóstenes", em homenagem ao grego que o propôs (Eratóstenes, 276-194 a.C.). Pergunte ao seu professor caso não encontre bibliografia a respeito.

3. Tremblay, Jean Paul; "Ciência dos Computadores"

(OBS.: faça uma versão dos algoritmos em Pascal, testando as respostas)

- a) pg 184, ex 4
- b) pg 208, ex 4
- c) pg 230, ex 3, 5