

| | | |
|---|--|------------------|
|  | LISTA I DE LTP E LPI – 1ºAI/ 1ºBI/ 1ºCI - 2º BIMESTRE Estruturas de Dados Compostas Homogêneas (Vetores e Strings) | |
| | Prof. MSC Marden Santos | Data: 20/04/2018 |
| | Aluno: | Nº |

- Desenvolva um algoritmo que leia uma variável composta unidimensional VET de dimensão igual a 10 verifique e escreva a quantidade de números armazenados que são maiores que 100.
- Fazer um algoritmo que calcule e escreva o somatório dos valores armazenados numa variável composta unidimensional A, de 10 elementos numéricos inteiros a serem lidos do dispositivo de entrada.
- Desenvolva um algoritmo que armazene 10 números inteiros em um vetor NUM e imprima na saída uma listagem numerada contendo o número e uma das mensagens: par ou ímpar.
- Fazer um algoritmo que leia dez elementos reais para cada vetor A e B. Gere e imprima um vetor C, cujos elementos sejam a diferença entre os elementos de posições equivalentes de A com B.
- Faça um algoritmo que leia em um vetor as idades de 10 alunos, calcular a média destas idades e imprimir na saída a média e a quantidade de alunos abaixo da média de idades calculada.
- Criar um algoritmo que leia dados para um vetor de 10 elementos inteiros. Imprimir o maior e o menor, sem ordenar, o percentual de números pares e a média dos elementos do vetor.
- Desenvolva um algoritmo que leia os dez elementos de um vetor A. construir um vetor B de mesmo tipo e dimensão, onde todo elemento do vetor B é o quadrado do elemento correspondente no vetor A. Mostre o conteúdo dos dois vetores.
- Crie um algoritmo que leia um conjunto de 10 elementos numéricos e informe:
 - Quantos números são iguais a 20;
 - Quantos são maiores que a média;
 - Quantos são iguais a média.
- Criar um algoritmo que leia um conjunto de 10 valores inteiros, armazene-os em um vetor e escreva-os ao contrário da ordem de leitura.
- Desenvolva um algoritmo que permita entrar com vários números, até se digitar o número 0. Imprimir quantos números iguais ao último número lido válido existe no vetor. O limite é de 15 elementos.
- Um time de basquete possui 12 jogadores. Deseja-se um programa que, dado o nome e a altura dos jogadores, determine:
 - O nome e a altura do jogador mais alto;
 - A média de altura do time;
 - A quantidade de jogadores com altura superior a media, listando o nome e a altura de cada um.
- Sabendo-se que dentro de um vetor existem letras, sendo que, cada letra ocupa uma posição, e que o tamanho do vetor é de 20 posições, elaborar um algoritmo para calcular quantas vezes aparecerá cada vogal. Imprima a vogal e sua respectiva quantidade.
- Considere os seguintes dígrafos: SS, RR, SC, XC, NH, LH, CH. Elabore um algoritmo que verifique se ocorre cada uma dessas possibilidades em um vetor de caracteres de 20 posições. Imprima na saída qual dígrafo foi encontrado no vetor.

14. Desenvolva um algoritmo que leia o nome, o CPF e a profissão de 20 pessoas, imprimir na saída quais as duas profissões que mais se repetem e quantas pessoas têm essas profissões.
15. Fazer um algoritmo que:
- Leia uma frase de 80 caracteres, incluindo brancos;
 - Conte quantos brancos existem na frase;
 - Conte quantas vezes a letra A aparece;
 - Conte quantas vezes ocorre um mesmo par de letras informado pelo usuário, na frase armazenada;
 - Imprima o que foi calculado nos itens *b*, *c* e *d*.
16. Fazer um algoritmo que:
- Leia dois vetores, contendo cada um 15 elementos numéricos;
 - Intercale os elementos destes dois conjuntos formando um novo vetor de 30 elementos;
 - Escreva o novo conjunto assim obtido.
17. Dados dois vetores R[1:10] e S[1:20], escreva um algoritmo que leia os vetores R e S e:
- Gere o vetor X correspondente à união dos vetores R e S;
 - Gere o vetor Y com os elementos comuns de R e S;
 - Gere o vetor Z com os elementos de R que não estão em S.
18. Está disponível num equipamento de entrada de dados o estoque de mercadorias de uma loja. São dados os códigos das mercadorias e as respectivas quantidades existentes. A seguir, estão os pedidos dos clientes. Fazer um algoritmo para a atualização do estoque, tal que:
- Seja lido e listado o estoque inicial (máximo de 20 mercadorias);
 - Sejam lidos os pedidos dos clientes, constituídos, cada um, do número do cliente, código da mercadoria e quantidade desejada;
 - Seja verificado, para cada pedido, se ele pode ser integralmente atendido. Em caso negativo, imprima o número do cliente e a mensagem "Estoque Insuficiente";
 - Seja atualizado o estoque após cada operação;
 - Seja listado o estoque final.
19. Desenvolva um algoritmo que imprime na saída um Triângulo de Pascal de ordem N (N lido). O algoritmo deve usar apenas um vetor e apenas um comando de repetição **for**.

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1

```

20. Construa um algoritmo que leia doze elementos de um vetor e coloque-os em ordem crescente e decrescente. Apresente os elementos ordenados nas duas situações.
21. Desenvolva um algoritmo que leia um vetor A com doze elementos. Após a leitura, colocar os seus elementos em ordem crescente. Depois leia um vetor B também com doze elementos e coloque-os em ordem crescente. Construir um vetor C, sendo cada elemento de C a soma do elemento correspondente de A com B. colocar em ordem crescente o vetor C e apresentar seu conteúdo.
22. Desenvolva um algoritmo que leia o nome, a idade e salário de N pessoas (N lido). O algoritmo deve ordenar e imprimir os dados destas pessoas ordenados pelo nome, pelo salário e pela idade (de maneira não decrescente).

23. Desenvolva um algoritmo que leia vinte elementos de um vetor A. construir um vetor B de mesmo tipo, observando a seguinte lei de formação: todo elemento de B deve ser o cubo do elemento de A correspondente. Montar uma rotina de pesquisa seqüencial, para pesquisar os elementos armazenados no vetor B.
24. Fazer um algoritmo que:
- a) Leia uma variável composta A com 30 valores numéricos;
 - b) Leia outra variável composta B com 30 valores numéricos;
 - c) Leia o valor de uma variável X;
 - d) Verifique qual o elemento de A que é igual a X;
 - e) Imprima o elemento de B de posição correspondente à do elemento de A igual a X.
25. Desenvolva um algoritmo que leia um vetor de 20 elementos, construir um vetor B de mesma dimensão com os mesmos elementos de A, com o acréscimo do valor dois a cada elemento. Colocar os elementos do vetor B em ordem crescente. Montar uma rotina de pesquisa binária, para pesquisar os elementos armazenados no vetor B.
26. Fazer um algoritmo que:
- a) Leia o valor de n ($n \leq 1000$) e os n valores de uma variável composta A de valores numéricos, ordenados de forma crescente;
 - b) Determine e imprima, para cada número que se repete no conjunto, a quantidade de vezes que ele aparece repetido;
 - c) Elimine os elementos repetidos, formando um novo conjunto;
 - d) Imprima o conjunto obtido no item c.
27. Elaborar um algoritmo que leia uma variável vetor do tipo caracter, com 30 ocorrências, ordenando-a crescentemente e imprimindo os elementos de índices pares, ou seja, igual a 2,4,6,...,30.
28. Uma costureira tem dez freguesas fixas. Ela deseja a criação de um algoritmo que controle seu negócio, seguindo a estrutura do menu abaixo:
- 1 – Cadastrar freguesas;
 - 2 – Cadastrar preços das costuras;
 - 3 – Calcular e imprimir o total que será pago por cada freguesa;
 - 4 – Listar dados de uma cliente pesquisada;
 - 5 – Sair do algoritmo/ programa.

29. Temperatura Lunar

Sem as proteções da atmosfera e do cinturão magnético que existem na Terra, a Lua fica exposta ao ataque do Sol, que é um astro em constante explosão atômica. As explosões do Sol emitem ondas letais de partículas. Uma pessoa que ficasse desprotegida na superfície da Lua, num lugar onde o Sol incidisse diretamente, sofreria um bombardeio radioativo tão intenso quanto se estivesse nas imediações da usina russa de Chernobyl no momento do acidente que matou 31 pessoas, em 1986. Além da radiação solar, outro efeito desta falta de proteção contra o Sol que existe na Lua é a enorme variação de temperatura. Nas regiões próximas do equador lunar, a variação de temperatura é brutal, passando de cerca de 130 graus positivos durante o dia a 129 graus negativos à noite. Para estudar com mais precisão as variações de temperatura na superfície da Lua, a NASA enviou à Lua uma sonda com um sensor que mede a temperatura de 1 em 1 minuto. Um dado importante que os pesquisadores desejam descobrir é como se comporta a média da temperatura, considerada em intervalos de uma dada duração (uma hora, meia hora, oito horas, etc.). Por exemplo, para a seqüência de medições 8, 20, 30, 50, 40, 20, -10, e intervalos de quatro minutos, as médias são respectivamente $108/4=27$, $140/4=35$, $140/4=35$ e $100/4=25$.

Tarefa

Você foi recentemente contratado pela NASA, e sua primeira tarefa é escrever um algoritmo/ programa que, conhecidos a sequência de temperaturas medidas pelo sensor, e o tamanho do intervalo desejado, informe qual a maior e qual a menor temperatura média observadas, considerando o tamanho do intervalo dado.

Entrada

A entrada é composta de vários conjuntos de teste. A primeira linha de um conjunto de teste contém dois números inteiros positivos N e M , que indicam respectivamente o número total de medições de temperatura de uma sequência obtida pelo sensor, e o tamanho dos intervalos, em minutos, em que as médias devem ser calculadas. As N linhas seguintes contém um número inteiro cada, representando a sequência de medidas do sensor. O final da entrada é indicado quando $N = M = 0$.

Exemplo de Entrada

```
4 2
-5
-12
0
6
```

```
7 4
35
-35
5
100
100
50
50
0 0
```

Saída

Para cada conjunto de teste da entrada seu algoritmo/ programa deve produzir três linhas. A primeira linha identifica o conjunto de teste, no formato “Teste n ”, onde n é numerado a partir de 1. A segunda linha deve conter dois números inteiros X e Y , separados por ao menos um espaço em branco, representando respectivamente os valores da menor e da maior média de temperatura, conforme determinado pelo seu programa. O valor da média deve ser truncado, se a média não for um número inteiro (ou seja, deve ser impressa apenas a parte inteira). A terceira linha deve ser deixada em branco. A grafia mostrada no Exemplo de Saída, abaixo, deve ser seguida rigorosamente.

Exemplo de Saída

```
Teste 1
-8 3
Teste 2
26 75
(esta saída corresponde ao exemplo de entrada acima)
```

Restrições

$0 \leq N \leq 10000$ ($N = 0$ apenas para indicar o fim da entrada)
 $-200 \leq \text{Temperatura} \leq 200$
 $1 \leq M \leq N$

30. Vivo ou Morto

Toda criança certamente já brincou de “vivo ou morto”. A brincadeira é dirigida por um “chefe” (um adulto), que comanda dois ou mais participantes (crianças). A brincadeira é composta de rodadas. No início, os participantes são organizados pelo chefe em fila única. A cada rodada o chefe grita “vivo” ou “morto” e todos os participantes tentam seguir sua ordem, levantando-se ao ouvir a palavra “vivo” ou abaixando-se ao ouvir a palavra “morto”. Um participante que não segue a ordem do chefe

é eliminado, deixando o seu lugar na fila. Os participantes remanescentes agrupam-se novamente em fila única, preenchendo as posições dos participantes eliminados, mas mantendo suas posições relativas. O jogo continua até que uma rodada seja composta por exatamente um participante. Tal participante é dito o vencedor do jogo. Por exemplo, considere que a brincadeira inicie com cinco participantes, identificados por números inteiros de 1 a 5, e que o chefe organize a fila na ordem $3 \rightarrow 2 \rightarrow 1 \rightarrow 4 \rightarrow 5$. Se na primeira rodada forem eliminados os participantes 2 e 4, a fila da segunda rodada será formada por $3 \rightarrow 1 \rightarrow 5$; se na segunda rodada for eliminado o participante 1, a fila da terceira rodada será formada por $3 \rightarrow 5$. Se na terceira rodada o participante 3 for eliminado, o vencedor da brincadeira será o participante 5.

Tarefa

Sua tarefa é escrever um algoritmo/ programa que determine o vencedor de uma partida de “vivo ou morto”, a partir da informação das ordens dadas pelo chefe e das ações executadas pelos participantes em cada rodada.

Entrada

A entrada é constituída de vários casos de teste, cada um representando uma partida. A primeira linha de um caso de teste contém dois números inteiros P e R indicando respectivamente a quantidade inicial de participantes ($2 \leq P \leq 100$) e quantidade de rodadas da partida ($1 \leq R \leq 100$). Os participantes são identificados por números de 1 a P . A segunda linha de um caso de teste descreve a fila organizada pelo chefe, contendo P números inteiros distintos x_1, x_2, \dots, x_P , onde x_1 representa o identificador do participante no primeiro lugar na fila, x_2 representa o identificador do participante no segundo lugar na fila, e assim por diante ($1 \leq x_i \leq P$). Cada uma das R linhas seguintes representa uma rodada, contendo um número inteiro N indicando o número de participantes da rodada ($2 \leq N \leq P$), um número inteiro J representando a ordem dada pelo chefe ($0 \leq J \leq 1$) e N números inteiros A_i representando a ação do participante colocado na i -ésima posição na fila ($0 \leq A_i \leq 1$). Ordens e ações “vivo” são representadas pelo valor 1, ordens e ações “morto” pelo valor zero. Cada partida tem exatamente um vencedor, determinado somente na última rodada fornecido no caso de teste correspondente. O final da entrada é indicado por $P = R = 0$. A entrada deve ser lida do dispositivo de entrada padrão (normalmente o teclado).

Saída

Para cada caso de teste seu algoritmo/programa deve produzir três linhas. A primeira identifica o conjunto de teste no formato “Teste n ”, onde n é numerado a partir de 1. A segunda linha deve conter o identificador do vencedor. A terceira linha deve ser deixada em branco. A grafia mostrada no Exemplo de Saída, abaixo, deve ser seguida rigorosamente.

Exemplo de Entrada

```
2 2
2 1
2 1 1 1
2 1 1 0

5 4
3 2 1 4 5
5 1 1 1 1 1 1
5 0 0 1 0 1 0
3 0 0 1 0
2 1 0 1
0 0
```

Saída para o Exemplo de Entrada

```
Teste 1
2

Teste 2
5
```