

Vetores e Strings (Capítulos 6 e 7)

1) Escreva um programa que armazena 15 números inteiros aleatórios em um vetor. O vetor deverá ser uma variável local dentro da função “main”.

Os exercícios de 2 a 9 devem ser feitos a partir do programa implementado no exercício 1.

Implemente funções que recebam o vetor de 15 números como parâmetro (implementado no exercício 1) e retornem:

- 2) a quantidade de números pares do vetor;
- 3) a soma dos números ímpares do vetor;
- 4) a quantidade de números com valor maior do que a média dos números do vetor;
- 5) o maior valor do vetor;
- 6) a maior diferença em valor absoluto entre elementos consecutivos do vetor;

Implemente procedimentos para:

- 7) colocar em ordem crescente os elementos do vetor. No programa principal, exiba o vetor antes e depois de ordená-lo.
- 8) exibir o 3o. maior elemento do vetor. Por exemplo: $v = \{2, 4, 4, 1, 3, 6, 5, 6\} \rightarrow 3o. \text{ Maior} = 4$.
- 9) eliminar números repetidos do vetor. Os números repetidos devem ser substituídos por novos números. Este procedimento deverá exibir na tela todos os elementos do vetor, antes e depois de eliminar os números repetidos.
- 10) Implemente a função **float max(float v [], int n)** que retorna o maior número entre os **n** primeiros elementos de um vetor.
- 11) Implemente a função **int compara (int vetor1 [], int vetor2 [])** que retorna 1 caso os vetores sejam iguais e 0 caso contrário. A quantidade de elementos em cada vetor pode ser diferente. Se julgar necessário, passe outros parâmetros para a função.

Exercícios de Strings

- 12) Implemente uma função que conta e retorna a quantidade de caracteres existente em um string passado como parâmetro.
- 13) Implemente o procedimento “**void concatenar (char s1[], char s2[])**” que copia o string s2 a partir do final de s1. Exemplo: s1 = “abc”, s2= “def”, após o procedimento concatenar s1 terá o conteúdo s1= “abcdef”.
- 14) Implemente o procedimento **void copia (char origem [], char destino [])** que faz uma cópia do vetor origem no vetor destino.
- 15) Implemente o procedimento **void copia (char origem [], char destino [], int n)** que copia os primeiros **n** caracteres do vetor origem para o vetor destino.
- 16) Implemente a função **int compara (char string1 [], char string2 [])** que retorna 1 caso as strings sejam iguais e 0 caso contrário.
- 17) Implemente o procedimento **void imprime_sobrenome (char nome_completo[])** que

recebe o nome completo de uma pessoa e imprime o último nome. Por exemplo, para o nome completo “Jose Maria da Silva”, o programa deve exibir: “Silva”.

18) Implemente o **void eliminar(char v [], char ch)** que elimina todas as ocorrências do caractere “ch” no vetor de caracteres “v”. Ao eliminar um caractere, os outros elementos do vetor devem ser movidos para esquerda. Por exemplo: v = “programacao de computadores” e ch = 'a'. Ao final do procedimento o conteúdo de “v” deverá ser “progrmco de computadores”. Você não pode utilizar um vetor auxiliar para implementar o procedimento.

19) Implemente um procedimento que coloca em ordem crescente os caracteres de um string “s” passado como parâmetro.

Exercícios de matrizes (Capítulo 6)

20) Escreva um programa que armazena 20 valores aleatórios em uma matriz 4x5. A matriz deverá ser uma variável local dentro da função “main”.

Os exercícios de 21 a 27 devem ser feitos a partir do programa implementado no exercício 20 .

Implemente funções que recebam a matriz 4x5 como parâmetro (implementado no exercício 20) e retornem:

- 21)** a quantidade de números pares da matriz;
- 22)** a soma dos números ímpares da matriz;
- 23)** a quantidade de números com valor maior do que a média dos números da matriz;
- 24)** o maior valor da matriz;
- 25)** o segundo maior valor da matriz;

26) Implemente procedimentos para:

27) eliminar os números repetidos da matriz. Os números repetidos devem ser substituídos por novos números. Este procedimento deverá exibir na tela todos os elementos da matriz antes e depois de eliminar os números repetidos.

28) Uma empresa tem registrado em uma tabela os consumos mensais de energia elétrica dos anos de 2003 até 2010. Cada linha representa um ano e cada coluna representa um mês. Considerando esses dados, faça um algoritmo para processar a tabela e produzir as seguintes informações:

- 1. Consumo médio em cada um dos meses;
- 2. Mês/ano em que houve o maior gasto com energia.

29) Implemente um programa que preenche com valores aleatórios uma matriz MxN e:

- 1. armazena em um vetor v1 o maior elemento cadastrado em cada coluna da matriz;
 - 2. armazena em um vetor v2 o menor elemento cadastrado em cada coluna da matriz;
- Ao final, o programa deve exibir: a matriz original, o vetor dos maiores elementos e o vetor dos menores elementos.

Obs.:

- Os valores de M e N devem ser definidos por meio da diretiva “#define”
- Você pode utilizar no máximo dois comandos “for” para encontrar os menores e maiores elementos das colunas.

A solução deve ser implementada no programa principal (“main”), sem utilizar funções.