

Lista de Exercícios 8 – Vetores e Arquivos – Segundo Semestre

Construa os programas solicitados nessa lista validando os dados de entrada e usando o tratamento de todas as exceções que possam ocorrer em cada um deles.

Crie os arquivos necessários para as entradas solicitadas.

1. Escreva um programa Java que leia um arquivo de texto chamado "numerosInteiros.txt", que contém uma sequência de números inteiros separados por vírgulas. O programa deve ler esse arquivo, armazenar os números em um vetor e, em seguida, realizar as seguintes operações:
 - a) Encontrar o maior e o menor número no vetor e exibi-los na tela.
 - b) Calcular a média dos números no vetor e exibir o resultado.
 - c) Verificar se um determinado número inteiro fornecido pelo usuário está presente no vetor
2. Crie um programa Java que lê um arquivo de texto chamado "textoOriginal.txt" e implementa um algoritmo de compressão personalizado que substitua sequências repetidas de letras por um número e a letra repetida (por exemplo, "aaa" seria substituído por "3a"). O programa deve salvar o texto comprimido em um novo arquivo chamado "textoComprimido.txt" e, também exibir na tela a taxa de compressão obtida (em porcentagem).
3. Desenvolva um programa Java que lê um arquivo CSV chamado "alunos.csv". Cada linha do arquivo contém informações sobre um aluno, com os seguintes campos: nome, nota1, nota2 e nota3. O programa deve realizar as seguintes operações:
 - a) Calcular a média final de cada aluno (média aritmética das três notas) e armazenar em um novo arquivo chamado "mediaAlunos.txt", com o seguinte formato: "Nome do Aluno: Média Final".
 - b) Identificar o aluno com a maior média e exibir na tela seu nome e sua média final.
 - c) Calcular a média da turma (média aritmética das médias finais de todos os alunos) e exibir na tela o resultado
4. Crie um programa Java que implemente uma agenda de contatos. O programa deve permitir ao usuário adicionar novos contatos, cada um com um nome, telefone e email. As informações dos contatos devem ser armazenadas em um arquivo chamado "contatos.txt". O arquivo deve ser lido ao rodar o programa ou criado, caso não exista. Os dados devem ser manipulados em vetores e salvos no arquivo ao encerrar o programa. O programa deve ter as seguintes funcionalidades:
 - a) Opção para adicionar um novo contato ao arquivo. O usuário deve fornecer o nome, telefone e email do contato.
 - b) Opção para listar todos os contatos cadastrados no arquivo, exibindo o nome, telefone e email de cada um.
 - c) Opção para buscar um contato pelo nome e, se encontrado, permitir que o usuário atualize o telefone e o email do contato.
 - d) Opção para excluir um contato pelo nome, removendo-o do arquivo.

5. Crie um programa Java que simule um diário de notas. O programa deve permitir ao usuário adicionar novas notas, cada uma com uma data e o valor da nota. As notas devem ser armazenadas em um arquivo chamado "notas.txt". O arquivo deve ser criado pelo programa ou lido caso já exista. Os dados devem ser manipulados em vetores e salvos no arquivo ao encerrar o programa. O programa deve ter as seguintes funcionalidades:
 - a) Opção para adicionar uma nova nota ao arquivo. O usuário deve fornecer a data e o conteúdo da nota.
 - b) Opção para listar todas as notas cadastradas no arquivo, exibindo a data e o conteúdo de cada uma.
 - c) Opção para buscar uma nota por data e, se encontrada, permitir que o usuário atualize o conteúdo da nota.
 - d) Opção para excluir uma nota por data, removendo-a do arquivo.