

REA-AED

Exercícios para Treinamento

Arquivos

1) Escreva um programa que:

- Crie/abra um arquivo texto de nome “arq.txt”
- Permita que o usuário grave diversos caracteres nesse arquivo, até que o usuário entre com o caractere ‘0’
- Feche o arquivo

Agora, abra e leia o arquivo, caractere por caractere, e escreva na tela todos os caracteres armazenados.

Resolução: <https://github.com/student072/Exercicios-REA-AED/blob/master/Arquivos/exercicio1.c>

2) Faça um programa que receba do usuário um arquivo texto e mostre na tela quantas linhas esse arquivo possui.

Resolução: <https://github.com/student072/Exercicios-REA-AED/blob/master/Arquivos/exercicio2.c>

3) Faça um programa que receba do usuário um arquivo texto e mostre na tela quantas letras são vogais e quantas são consoantes.

Resolução: <https://github.com/student072/Exercicios-REA-AED/blob/master/Arquivos/exercicio3.c>

4) Faça um programa que receba do usuário um arquivo texto e um caractere. Mostre na tela quantas vezes aquele caractere ocorre dentro do arquivo.

5) Faça um programa que receba do usuário um arquivo texto e mostre na tela quantas vezes cada letra do alfabeto aparece dentro do arquivo.

6) Faça um programa que receba do usuário um arquivo texto. Crie outro arquivo texto contendo o texto do arquivo de entrada, mas com as vogais substituídas por ‘*’.

Resolução: <https://github.com/student072/Exercicios-REA-AED/blob/master/Arquivos/exercicio6.c>

7) Faça um programa que leia o conteúdo de um arquivo e crie um arquivo com o mesmo conteúdo, mas com todas as letras minúsculas convertidas para maiúsculas. Os nomes dos arquivos serão fornecidos, via teclado, pelo usuário. A função que converte maiúscula para minúscula é o `toupper()`. Ela é aplicada em cada caractere da string.

8) Faça um programa que receba dois arquivos do usuário, e crie um terceiro arquivo com o conteúdo dos dois primeiros juntos (o conteúdo do primeiro seguido do conteúdo do segundo).

- 9) Faça um programa no qual o usuário informa o nome do arquivo e uma palavra, e retorne o número de vezes que aquela palavra aparece no arquivo.
- 10) Faça um programa que permita que o usuário entre com diversos nomes e telefone para cadastro, e crie um arquivo com essas informações, uma por linha. O usuário finaliza a entrada com '0' para o telefone.
- 11) Dado um arquivo contendo um conjunto de nome e data de nascimento (MM AAAA, isto é, 2 inteiros em sequência), faça um programa que leia o nome do arquivo e a data de hoje e construa outro arquivo contendo o nome e a idade de cada pessoa do primeiro arquivo.
- 12) Faça um programa que recebe um vetor de 10 números, converta cada um desses números para binário e grave a sequência de 0s e 1s em um arquivo texto. Cada número deve ser gravado em uma linha.
- 13) Faça um programa que leia um arquivo contendo o nome e o preço de diversos produtos (separados por linha), e calcule o total da compra.
- 14) Crie um programa que receba como entrada o número de alunos de uma disciplina. Aloque dinamicamente dois vetores para armazenar as informações a respeito desses alunos. O primeiro vetor contém o nome dos alunos e o segundo contém suas notas finais. Crie um arquivo que armazene, a cada linha, o nome do aluno e sua nota final. Use nomes com no máximo 40 caracteres.
- 15) Escreva um programa que leia a profissão e o tempo de serviço (em anos) de cada um dos 5 funcionários de uma empresa e armazene-os no arquivo "emp.txt". Cada linha do arquivo corresponde aos dados de um funcionário. Utilize o comando fprintf(). Em seguida, leia o mesmo arquivo utilizando fscanf(). Apresente os dados na tela.
- 16) Faça um programa gerenciar uma agenda de contatos. Para cada contato armazene o nome, o telefone e o aniversário (dia e mês). Grave estes dados em um arquivo. Além disso, o programa deve permitir:
- Inserir contato
 - Remover contato
 - Pesquisar um contato pelo nome
 - Listar todos os contatos
 - Imprimir os aniversariantes do mês.
- 17) Crie um programa que declare uma estrutura para o cadastro de alunos.
- Deverão ser armazenados, para cada aluno: matrícula, sobrenome (apenas um), e ano de nascimento.
 - Ao início do programa, o usuário deverá informar o número de alunos que serão armazenados

- O programa deveria alocar dinamicamente a quantidade necessária de memória para armazenar os registros dos alunos.
- O programa deverá pedir ao usuário que entre com as informações dos alunos.
- Em seguida, essas informações deverão ser gravadas em um arquivo.
- Ao final, mostrar os dados armazenados e liberar a memória alocada

Ao iniciar o programa, forneça ao usuário uma opção para carregar os registros do arquivo para a memória do computador alocando dinamicamente a quantidade de memória necessária.

Dica: para que o usuário possa entrar com novos dados, além dos que foram obtidos a partir do arquivo, use a função `realloc()` para realocar a quantidade de memória usada.

18) Faça um programa para gerenciar as notas dos alunos de uma turma salva em um arquivo. O programa deverá ter um menu contendo as seguintes opções:

- Definir informações da turma;
- Inserir aluno e notas;
- Exibir alunos e medias;
- Exibir alunos aprovados;
- Exibir alunos reprovados;
- Salvar dados em Disco;
- Sair do programa (fim).