

Prova P5

1. Implemente um programa em C que receba como entrada o nome de um arquivo de entrada e o nome de um arquivo de saída. Cada linha do arquivo de entrada possui colunas de tamanho de 30 caracteres. No arquivo de saída deverá ser escrito o conteúdo do arquivo de entrada de forma inversa. Veja um exemplo:

Arquivo de entrada:

Hoje é dia de prova de IP

A prova está muito fácil

Vou tirar uma boa nota

Arquivo de saída:

Aton aob amu rarit uov

Licáf otium átse avorp A

PI ed avorp ed aid é ejoH

2. Faça um programa em C, que declare uma *struct* para gerenciar uma agenda de contatos. Para cada contato armazene o nome, o telefone e o aniversário (dia e mês). O dia e o mês devem ser validados. Inicialmente deve-se gerar um vetor (alocado dinamicamente) com 5 contatos e gravar no arquivo chamado `contatos.dat`. O programa deve permitir **(1)** inserir contato, **(2)** remover contato, **(3)** pesquisar um contato pelo nome, **(4)** listar todos os contatos, **(5)** listar os contatos cujo nome inicia com uma dada letra, **(6)** imprimir os nomes dos aniversariantes do mês. Cada uma das opções de (1) a (5) devem ser implementadas através de funções, que recebe como parâmetro por referência o ponteiro para o arquivo e atualiza o arquivo, conforme a operação definida para a função.

OBS: A cada inserção ou remoção, o tamanho do arquivo deve ser alterado usando a função `realloc(ptr,tamanho);`

3. Implemente um programa em C que declare uma estrutura (registro) para o cadastro de alunos.
 - 3.1 Deverão ser armazenados, para cada aluno: matrícula, sobrenome (apenas um), e ano de nascimento.

- 3.2 Ao início do programa, o usuário deverá informar o número de alunos que serão armazenados;
- 3.3 O programa deverá alocar dinamicamente a quantidade necessária de memória para armazenar os registros dos alunos.
- 3.4 O programa deverá pedir ao usuário que entre com as informações dos alunos.
- 3.5 Em seguida, essas informações deverão ser gravadas em um arquivo.
- 3.6 Ao final, mostrar os dados armazenados e liberar a memória alocada.

OBS: Fique à vontade para implementar de acordo com sua criatividade.

INSTRUÇÕES:

- 1. Implementar cada algoritmo em seu arquivo particular.
- 2. Compactar os dois arquivos *.c em um único arquivo com o seguinte nome: nome-aluno_p5.zip;
- 3. Entregar pela plataforma Turing até o dia 10/12/2020, às 23h59min.