**LISTA-07: (<fstream>, Enumeration e Alocação Dinâmica I)**

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

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

1. Faça um programa que receba do usuário um arquivo texto e mostre na tela quantas linhas esse arquivo possui.
2. Faça um programa que receba do usuário um arquivo texto e um caractere. Mostre na tela quantas vezes aquele caractere ocorre dentro do arquivo.
3. 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).
4. Crie um programa que tenha um enumerador para formalizar os níveis de dificuldade de um jogo (Fácil, Normal e Difícil).
5. Crie um programa que disponha de enumeradores para os modos de atendimento de um sistema (Comum e Preferencial).
6. Elabore um programa que disponha de enumeradores para representar os 4 cargos de funcionários de uma empresa, e mais 2 setores de serviço para cada cargo.

**EXEMPLO:**

**Cargos: Vendedor, Almoxarife, Gerente, Relações Públicas.**

**Setores: Interno, Externo.**

1. Crie um programa que tenha um enumerador para agrupar os meses do ano, os dias da semana e os períodos do dia (manhã, tarde, noite, madrugada).
2. Crie um programa que:
   1. Aloque dinamicamente um array de 5 números inteiros,
   2. Peça para o usuário digitar os 5 números no espaço alocado,
   3. Mostre na tela os 5 números,
   4. Libere a memória alocada.
3. Faça um programa que leia do usuário o tamanho de um vetor a ser lido e faça a alocação dinâmica de memória. Em seguida, leia do usuário seus valores e mostre quantos dos números são pares e quantos são ímpares.
4. Faça um programa que leia um número N e:

* Crie dinamicamente e leia um vetor de inteiro de N posições;
* Leia um número inteiro X e conte e mostre os múltiplos desse número que existem no vetor.

1. Faça um programa que simule a memória de um computador: o usuário irá especificar o tamanho da memória, ou seja, quantos bytes serão alocados do tipo inteiro. Para tanto, a memória solicitada deve ser um valor múltiplo do tamanho do tipo inteiro. Em seguida, o usuário terá 2 opções: inserir um valor em uma determinada posição ou consultar o valor contido em uma determinada posição. A memória deve iniciar com todos os dados zerados.
2. Faça um programa para armazenar em memória um vetor de dados contendo 1500 valores do tipo inteiro, usando a função de alocação dinâmica de memória CALLOC:
   1. Faça um loop e verifique se o vetor contém realmente os 1500 valores inicializados com zero (conte os 1500 zeros do vetor).
   2. Atribua para cada elemento do vetor o valor do seu índice junto a este vetor.
   3. Exibir na tela os 10 primeiros e os 10 últimos elementos do vetor.
3. Faça um programa que pergunte ao usuário quantos valores ele deseja armazenar em um vetor de double, depois use a função MALLOC para reservar (alocar) o espaço de memória de acordo com o especificado pelo usuário. Esse vetor deve ter um tamanho maior ou igual a 10 elementos. Use este vetor dinâmico como um vetor comum, atribuindo aos 10 primeiros elementos do vetor valores aleatórios (usando a função rand) entre 0 e 100. Exiba na tela os valores armazenados nos 10 primeiros elementos do vetor.

**DESAFIOS:**

1. Crie um programa que declare uma estrutura (struct) para o cadastro de alunos.
   1. Deverão ser armazenados, para cada aluno: matrícula, sobrenome e ano de nascimento.
   2. Ao início do programa, o usuário deverá informar o número de alunos que serão armazenados
   3. O programa deverá alocar dinamicamente a quantidade necessária de memória para armazenar os registros dos alunos.
   4. O programa deverá pedir ao usuário que entre com as informações dos alunos.
   5. Ao final, mostrar os dados armazenados e liberar a memória alocada.
2. Considere um cadastro de produtos de um estoque, com as seguintes informações para cada produto:

* Código de identificação do produto: representado por um valor inteiro
* Nome do produto
* Quantidade disponível no estoque: representado por um número inteiro
* Preço de venda: representado por um valor real
  1. Defina uma estrutura, denominada produto, que tenha os campos apropriados para guardar as informações de um produto
  2. Crie um conjunto de N produtos (N é um valor fornecido pelo usuário) e peça ao usuário para entrar com as informações de cada produto
  3. Encontre o produto com o maior preço de venda
  4. Encontre o produto com a maior quantidade disponível no estoque