Exercício 1: Calculadora Modularizada

Objetivo: Praticar a criação de módulos para diferentes operações matemáticas.
Enunciado:
Crie um programa que funcione como uma calculadora simples, realizando as operações de soma, subtração, multiplicação e divisão.

- Separe cada operação matemática em um arquivo C diferente (e.g., soma.c, subtracao.c).
- Crie um arquivo de cabeçalho (operacoes.h) que declare as funções.
- O arquivo principal (main.c) deve incluir o cabeçalho e chamar as funções implementadas nos módulos.

Exercício 2: Manipulação de Strings em Módulos

Objetivo: Modularizar funções para operações com strings.
Enunciado:
Crie um programa que manipule strings e implemente as seguintes funcionalidades:

1. Converter uma string para letras maiúsculas.
2. Reverter uma string.
3. Contar o número de caracteres de uma string.
   Cada funcionalidade deve ser implementada em um arquivo C diferente (e.g., uppercase.c, revert.c, count.c), com suas respectivas declarações em um arquivo de cabeçalho (strings.h).

Exercício 3: Gerenciamento de Contas Bancárias Modularizado

Objetivo: Criar módulos para encapsular a lógica de um programa de banco simples.
Enunciado:

Implemente um programa para gerenciamento de contas bancárias com as seguintes funcionalidades:

1. Criar uma nova conta.
2. Depositar dinheiro em uma conta.
3. Sacar dinheiro de uma conta.
4. Exibir o saldo de uma conta.
   Cada funcionalidade deve ser implementada em um módulo específico. O programa principal (main.c) deve interagir com o usuário e chamar as funções dos módulos apropriados.

Exercício 4: Sistema de Notas Modularizado

Objetivo: Modularizar o cálculo e exibição de médias de alunos.
Enunciado:
Crie um programa para calcular e exibir a média de notas de um aluno.

- Implemente um módulo para entrada de dados (input.c).
- Outro módulo para calcular a média (calculo.c).
- Um terceiro módulo para exibir os resultados (output.c).
  Um arquivo de cabeçalho (sistemaNotas.h) deve declarar todas as funções necessárias.

Exercício 5: Biblioteca para Operações com Vetores

Objetivo: Criar uma biblioteca modular para trabalhar com vetores.
Enunciado:
Crie uma biblioteca chamada vetores para realizar operações comuns com vetores de inteiros, incluindo:

1. Ordenação dos elementos em ordem crescente.
2. Busca de um elemento específico no vetor.
3. Cálculo da soma de todos os elementos.
   Implemente essas funcionalidades em arquivos separados (ordenar.c, buscar.c, somar.c) e inclua as declarações em um arquivo de cabeçalho (vetores.h). No programa principal (main.c), use as funções da biblioteca para testar as funcionalidades.