

UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
CAMPUS PAU DOS FERROS
DISCIPLINA: ALGORITMOS E ESTRUTURAS DE DADOS I
PROFESSORA: ROSANA REGO
LISTA DE EXERCÍCIOS - TIPOS ESTRUTURADOS

Questão 1) Crie um tipo estruturado para armazenar dados de um ingresso. Uma estrutura deste tipo possui os seguintes campos: preço, local e atração.

- a) Escreva uma função que receba como parâmetro o endereço de uma estrutura do tipo Ingresso e preencha seus campos com valores fornecidos pelo usuário via teclado. Protótipo: void preenche(Ingresso* i);
- b) Escreva uma função que receba como parâmetro o endereço de uma estrutura do tipo Ingresso e imprima os valores dos seus campos. Protótipo: void imprime(Ingresso* i);
- c) Escreva uma função que receba, como parâmetros, o endereço de uma estrutura do tipo Ingresso e um novo valor de preço e atualize o preço do ingresso para o novo valor. Protótipo: void altera_preco(Ingresso* i, float valor);
- d) Escreva uma função que receba, como parâmetros, um vetor de ingressos e o tamanho do vetor e imprima os eventos de ingresso mais barato e mais caro. Protótipo: void imprime_menor_maior_preco(int n, Ingresso* vet);
- e) Escreva uma função main para testar as funções anteriores.

Questão 2) Escreva um programa em C que manipula dados de um conjunto de alunos da disciplina de Estrutura de Dados e Programação da UFERSA Pau dos Ferros. Para representar um aluno são necessárias as seguintes informações: matrícula (inteiro), nome (80 caracteres), turma (caractere), três notas (reais em um vetor) e uma média (real). Seu programa deve levar em consideração que é ofertado um número máximo de vagas para a disciplina. Sabendo dessas informações, faça o que se pede nos itens a seguir:

- a) Implemente uma função que recebe os dados de um aluno e o matricula na disciplina, caso haja vaga disponível. Protótipo: void matricula(int n, Aluno** alunos);
- b) Implemente uma função que lança as notas e calcula a média dos alunos da disciplina. Protótipo: void lanca_notas(int n, Aluno** alunos);
- c) Implemente uma função que imprime os dados de todos os alunos da disciplina. Protótipo: void imprime_tudo(int n, Aluno** alunos);
- d) Implemente uma função que imprime os dados de todos os alunos de uma turma. Protótipo: void imprime_turma(int n, Aluno** alunos, char turma);
- e) Implemente uma função que imprime os alunos aprovados de uma determinada turma. Para ser aprovado, o aluno deve ter média maior que ou igual a 7,0. Protótipo: void imprime_turma_aprovados(int n, Aluno** alunos, char turma);

Sugestões: i) Use um vetor de ponteiros para estrutura; ii) Use uma constante simbólica para representar o número máximo de alunos. Iii) Antes de manipular o vetor de ponteiros, inicialize cada um de seus elementos com NULL (você pode escrever uma função para isso).

OBS.: Os exercícios apresentados sugerem a implementação de diferentes funções. Sendo assim, o programador deve escrever um programa (função main) para testar sua implementação.

Questão 3) Defina estruturas para representar retângulos (dadas a base e a altura) e círculos (dado o raio), e implemente as funções a seguir:

- a) Dado um círculo, crie e retorne o maior retângulo circunscrito de altura h. Considere que h é menor do que o diâmetro do círculo.
Protótipo: Ret* ret_circunscrito(Circ* c, float h);
- b) Dado um retângulo, crie e retorne o maior círculo interno ao retângulo.
Protótipo: Circ* circ_interno(Ret* r);