**Exercícios aplicando Structs**

**Soluções**

1. Faça um programa em C que cadastre o nome, altura (em metros) e peso (em Kg) de várias pessoas (pode ser 3 para facilitar a entrada de dados). Em seguida calcula, por meio de uma função, o IMC de todos e mostre cada nome e respectivo IMC.

Nota: IMC [índice de massa corporal] = peso / altura2.

#include <stdio.h>

#include <stdlib.h>

#define MAX 3

typedef struct{

char nome[51];

float altura;

int peso;

}Pessoa;

float calcIMC(int p, float h){

float imc = p / (h \* h);

return imc;

}

int main(){

Pessoa pessoas[MAX];

int i;

float imc;

for(i = 0; i < MAX; i++){

printf("Informe nome da %da. pessoa: ", i+1);

fflush(stdin);

gets(pessoas[i].nome);

printf("Informe altura da %da. pessoa: ", i+1);

fflush(stdin);

scanf("%f", &pessoas[i].altura);

printf("Informe peso da %da. pessoa: ", i+1);

fflush(stdin);

scanf("%d", &pessoas[i].peso);

}

printf("\n\n Calculo do IMC: \n");

for(i = 0; i < MAX; i++){

imc = calcIMC(pessoas[i].peso, pessoas[i].altura);

printf("\n%s tem IMC = %.2f", pessoas[i].nome, imc);

}

}

1. Um ponto é representado no plano cartesiano por sua coordenada (abscissa x e ordenada y). Faça um programa em C que que leia um ponto pelo teclado. Depois leia mais três pontos, que devem ser armazenados em um vetor. Em seguida, informa qual dos três pontos está mais próximo do primeiro. **Observação**: a distância entre dois pontos P1(x1, y1) e P2(x2, y2) pode ser calculada usando Pitágoras [ d2 = (x1 – x2)2 + (y1 – y2)2] e deve ser feita por uma função que receba como parâmetros os dois pontos e retorne a distância entre ambos.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#define MAX 3

typedef struct{

float x, y;

}Ponto;

float distancia(Ponto p1, Ponto p2){

float d = sqrt(pow(p1.x - p2.x, 2) + pow(p1.y - p2.y, 2));

return d;

}

int main(){

Ponto p, pontos[MAX];

int i, maisProx;

float d, menorD;

printf("Informe coordenadas do ponto principal: ");

fflush(stdin);

scanf("%f", &p.x);

scanf("%f", &p.y);

for(i = 0; i < MAX; i++){

printf("Informe coordenada do ponto %d: ", i+1);

fflush(stdin);

scanf("%f", &pontos[i].x);

scanf("%f", &pontos[i].y);

}

printf("\n\n Ponto principal: (%.1f, %.1f)\n", p.x, p.y);

for(i = 0; i < MAX; i++){

printf("\nPonto %d: (%.1f, %.1f)", i+1,

pontos[i].x, pontos[i].y);

}

menorD = distancia(p, pontos[0]);

maisProx = 1;

for(i = 1; i < MAX; i++){

d = distancia(p, pontos[i]);

if (d < menorD){

menorD = d;

maisProx = i+1;

}

}

printf("\n Menor distancia do ponto principal e' %.2f

do ponto %d\n\n", menorD, maisProx);

}

1. Faça um programa em C que mantenha um cadastro de 5 livros, cada livro descrito pelo título (string com até 30 caracteres), autor (string até 20 caracteres), ano de publicação (inteiro). O programa deverá carregar, pelo teclado, os dados dos livros e depois mantenha um procedimento que repetidamente pesquise um livro por título, até que o usuário informe que não deseja mais pesquisar títulos.

#include <stdio.h>

#include <stdlib.h>

#define MAX 5

typedef struct{

char titulo[31];

char autor[21];

int ano;

}Livro;

void pesquisa(Livro \*l){

char op, t[31];

int i;

do{

printf("\nInforme titulo para pesquisar: ");

fflush(stdin);

gets(t);

i = 0;

while(i < MAX){

if(strcmp(l[i].titulo, t) == 0)

break;

else i++;

}

if(i >= MAX)

printf("\nTitulo %s nao encontrado.\n", t);

else{

printf("\nTitulo %s encontrado. ", l[i].titulo);

printf("\nAutor: %s, Ano Publ.: %d\n", l[i].autor, l[i].ano);

}

printf("\nDeseja pesquisar outro titulo?[S/N] ");

op = toupper(getch());

}while (op == 'S');

}

int main(){

Livro livros[MAX];

int i;

for(i = 0; i < MAX; i++){

printf("Informe titulo do livro %d: ", i+1);

fflush(stdin);

gets(livros[i].titulo);

printf("Informe autor do livro %d: ", i+1);

fflush(stdin);

gets(livros[i].autor);

printf("Informe ano de publicacao do livro %d: ", i+1);

fflush(stdin);

scanf("%d", &livros[i].ano);

}

pesquisa(livros);

}

1. Faça um programa em C que cadastre o nome, a matrícula e duas notas de 4 alunos, obtem pelo teclado os dados dos alunos e em seguida imprima o nome, matrícula e média de cada aluno. Observação: a média deve ser calculada por uma função que receba as notas e retorne o valor calculado.

#include <stdio.h>

#include <stdlib.h>

#define MAX 4

typedef struct{

char nome[31];

char matr[21];

float nota1, nota2;

}Aluno;

float media(Aluno al){

float m = (al.nota1 + al.nota2)/2;

return m;

}

int main(){

Aluno alunos[MAX];

int i;

for(i = 0; i < MAX; i++){

printf("Informe nome do aluno %d: ", i+1);

fflush(stdin);

gets(alunos[i].nome);

printf("Informe matricula do aluno %d: ", i+1);

fflush(stdin);

gets(alunos[i].matr);

printf("Informe primeira nota do aluno %d: ", i+1);

fflush(stdin);

scanf("%f", &alunos[i].nota1);

printf("Informe segunda nota do aluno %d: ", i+1);

fflush(stdin);

scanf("%f", &alunos[i].nota2);

}

for(i = 0; i < MAX; i++){

printf("\n\nAluno %s, matriula: %s", alunos[i].nome, alunos[i].matr);

printf(" obteve media: %.1f!", media(alunos[i]));

}

}

1. Um supermercado mantém um cadastro com os seus produtos oferecidos. Cada produto é descrito por um código (*int*), preço (*double*) e quantidade em estoque (*int*). Faça um programa C que mantenha um vetor com 50 produtos em estoque. O programa deverá iniciar, por meio de um procedimento que receba o vetor como parâmetro por referência, e inicie todos os seus elementos com números aleatórios (códigos entre 1000 e 4000, preços entre 1.00 e 100.00 e quantidade em estoque entre 0 e 1000). Em seguida o programa chama um procedimento chamado estoqueMinimo que varre o vetor e informa o código dos produtos que estão com quantidade em estoque menor do que 50).

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define MAX 50

typedef struct{

int cod;

double preco;

int qtEst;

}Produto;

void iniciaDados(Produto \*p){

srand(time(NULL));

int i;

for(i = 0; i < MAX; i++){

p[i].cod = rand()%3001+1000;

p[i].preco = rand()%991/10.0 + 1.00;

p[i].qtEst = rand()%1001;

}

}

int main(){

Produto prods[MAX];

int i;

iniciaDados(prods);

printf("\n\n");

for(i = 0; i < MAX; i++){

printf("\n%2d) Produto %5d", i+1, prods[i].cod);

printf(" preco R$: %6.2f", prods[i].preco);

printf(" qt em estoque: %4d", prods[i].qtEst);

}

printf("\n\n");

for(i = 0; i < MAX; i++){

if(prods[i].qtEst < 50){

printf("\nProduto %d: estoque baixo: %d",

prods[i].cod, prods[i].qtEst);

}

}

printf("\n\n");

}