



UNIVERSIDADE ESTADUAL DO PARÁ

IGOR DANTAS QUINTANILHA

Atividade avaliativa v

Belém-PA

2021

IGOR DANTAS QUINTANILHA

Atividade Avaliativa v

Trabalho avaliativo apresentado à Universidade

Do Estado do Pará, como solicito do
curso Bacharelado em Engenharia de
Software.

Professor: Thiago Nicolau Magalhães De Souza Conte.

Belém-PA

2021

1. Seja uma estrutura para descrever os carros de uma determinada revendedora, contendo os seguintes campos:

marca: string de tamanho 15
ano: inteiro
cor: string de tamanho 10
preço: real

- a) Escrever a definição da estrutura carro.
- b) Declarar o vetor vetcarros do tipo da estrutura definida acima, de tamanho 20 e global. Crie um menu para:
- c) Definir um bloco de programa para ler o vetor vetcarros.
- d) Definir um bloco de programa que receba um preço e imprima os carros (marca, cor e ano) que tenham preço igual ou menor ao preço recebido.
- e) Defina um bloco de programa que leia a marca de um carro e imprima as informações de todos os carros dessa marca (preço, ano e cor).
- f) Defina um bloco de programa que leia uma marca, ano e cor e informe se existe ou não um carro com essas características. Se existir, informar o preço.

```
#include <stdio.h>
```

```
#include <string.h>
```

```
#include <locale.h>
```

```
struct carro
```

```
{
```

```
char marca[25], cor[25];
```

```
int ano;
```

```
float preco;
```

```
};
```

```
int main ()
```

```
{
```

```
    setlocale(LC_ALL, "");
```

```
    struct carro vetcarros[20];
```

```
    int i = 1, ano_consulta;
```

```
    float preco_consulta;
```

```
    char marca_consulta[20], cor_consulta[15];
```

```
    for (i = 1; i <= 20; i++)
```

```
{printf("Informe a marca do carro %d: ", i);  
    scanf("%s", &vetcarros[i].marca);
```

```
    printf("Informe o ano do carro: ");  
    scanf("%d", &vetcarros[i].ano);
```

```
    printf("Informe a cor do carro: ");  
    scanf("%s", &vetcarros[i].cor);
```

```
    printf("Informe o preco do carro: ");  
    scanf("%f", &vetcarros[i].preco);
```

```
    printf("\n");}
```

```
printf("\nInforme o preço do carro para consultar: ");  
scanf("%f", &preco_consulta);
```

```
for(i = 1; i <= 20; i++)  
{if (vetcarros[i].preco <= preco_consulta)  
    {printf("\nA marca do carro disponível é: %s", vetcarros[i].marca);  
    printf("\nA cor do carro disponível é: %s", vetcarros[i].cor);  
    printf("\nO ano do carro disponível é: %d", vetcarros[i].ano);  
    printf("\nO preço do carro disponível é: R$%.2f", vetcarros[i].preco);  
    }  
}
```

```
    printf("\n\nInforme a marca do carro para consultar: ");  
    scanf("%s", &marca_consulta);
```

```
for(i = 1; i <= 20; i++)  
{if(strcmp(marca_consulta, vetcarros[i].marca) == 0)
```

```

        {printf("\nO preço do carro disponível é: R$%.2f", vetcarros[i].preco);
        printf("\nA cor do carro disponível é: %s", vetcarros[i].cor);
        printf("\nO ano do carro disponível é: %d", vetcarros[i].ano);}}

printf("\n\nInforme a marca do carro para consultar: ");
scanf("%s", &marca_consulta);

printf("\nInforme o ano do carro para consultar: ");
scanf("%d", &ano_consulta);

printf("\nInforme a cor do carro para consultar: ");
scanf("%s", &cor_consulta);


        for(i = 1; i <= 20; i++)

        { if (strcmp(cor_consulta,vetcarros[i].cor) == 0 && ano_consulta == vetcarros[i].ano &&
        strcmp(marca_consulta,vetcarros[i].marca) == 0)

            {printf("\nO carro disponível tem o preço de: R$%.2f", vetcarros[i].preco);}

        else

            {printf("\nNão há carros com as especificações informadas!");}}

return 0;

}

```

2. Elaborar um algoritmo que auxilie no controle de uma fazenda de gado que possui um total de 2000 cabeças de gado. A base de dados é formada por um conjunto de estruturas (registros) contendo os seguintes campos referentes a cada cabeça de gado:

- código: código da cabeça de gado,
- leite: número de litros de leite produzido por semana,
- alim: quantidade de alimento ingerida por semana - em quilos,
- nasc: data de nascimento - mês e ano,
- abate: 'N' (não) ou 'S' (sim).

O campo nasc. é do tipo struct data que por sua vez, possui dois campos:

- mês
- ano

Elaborar funções para:

1

a) Ler a base de dados (código, leite, alim, nasc.mês e nasc.ano), armazenado em um vetor de estruturas.

b) Preencher o campo abate, considerando que a cabeça de gado irá para o abate caso:

- tenha mais de 5 anos, ou;
- produza menos de 40 litros de leite por semana, ou;
- produza entre 50 e 70 litros de leite por semana e ingira mais de 50 quilos de alimento por dia.

Crie o menu de opções para:

c) Devolver a quantidade total de leite produzida por semana na fazenda.

d) Devolver a quantidade total de alimento consumido por semana na fazenda.

e) Devolver a quantidade total de leite que vai ser produzido por semana na fazenda, após o abate

f) Devolver a quantidade total de alimento que vai ser consumido por semana na fazenda, após o abate

g) Devolver número de cabeças de gado que iram para o abate.

h) Sair do programa.

```
#include <stdio.h>
```

```
#include <locale.h>
```

```
struct nasc
```

```
{  
    int mes;  
    int ano;  
};
```

```
struct gado
```

```
{  
    int codigo;  
    float leite;  
    float alim;  
    struct nasc nascimento;  
    int idade;  
};
```

```
int main ()
```

```
{  
    setlocale(LC_ALL, "");  
    struct gado x[2000];  
    int i;  
    int abateCont = 0;  
    int r = 1;  
    float totalleite, totalalim, totalleiteabate, alimentodia, totalalimabate;
```

```
    for (i = 1; i <= 2000; i++){
```

```
        printf("informe o código do gado %d: \n", i);
```

```
        scanf("%d", &x[i].codigo);
```

```
        printf("informe quantos litros de leite foi produzido na semana: \n");
```

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

```
        printf("informe quantos quilos de alimento foi consumido na semana: \n");
```

```

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

printf("informe o mês de nascimento: \n");
scanf("%d", &x[i].nascimento.mes);

printf("informe o ano de nascimento: \n");
scanf("%d", &x[i].nascimento.ano);

x[i].idade = 2021 - x[i].nascimento.ano;

totalleite = x[i].leite + totalleite;

totalalim = x[i].alim + totalalim;

alimentodia = x[i].alim / 7;

if( x[i].idade > 5 || x[i].leite < 40 || x[i].leite > 50 && x[i].leite < 70 && alimentodia >
50){

    abateCont++;

} else{

    totalleiteabate = totalleiteabate + x[i].leite;

    totalalimabate = totalalimabate + x[i].alim;
}
}

printf("\nQuantidade total de leite produzida por semana na fazenda é: %.1f litros",
totalleite);

printf("\nQuantidade total de alimento consumido por semana na fazenda é: %.2f kg",
totalalim);

printf("\nQuantidade total de leite produzida por semana na fazenda após o abate é:
%.1f litros", totalleiteabate);

```



```
printf("\nQuantidade total de alimento consumido por semana na fazenda após o abate  
é: %.2f kg", totalalimabate);
```

```
printf("\nQuantidade total de gados que vão pro abate na fazenda: %d gados",  
abateCont);
```

```
return (0);
```

```
}
```

3. Seja um algoritmo para controlar os produtos do estoque de um supermercado. Para cada produto, tem-se os seguintes campos:

nome: string de tamanho 15
setor: caracter
quantidade: inteiro
preço: real //preço por unidade do produto

Escrever a definição da estrutura produto.

- a) Declarar o vetor estoque do tipo da estrutura definida acima, de tamanho 100 e global.
- b) Crie um menu para:
 - a) Definir um bloco de instruções para ler o vetor estoque.
 - b) Definir um bloco de instruções que receba um setor e devolva o número de diferentes produtos desse setor.
 - c) Definir um bloco de instruções que calcule e devolva o total de capital investido em produtos do supermercado.
 - d) Sair do Programa.

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#define TAM 100
struct estoque
{
    char nome_produto[30];
    char setor[30];
    int qtd_produtos[TAM];
    float preco_prod[TAM];
};
struct estoque estoque[TAM];

int cadastro(struct estoque estoque[TAM]);
{
    for(int i = 0; i<=TAM; i++)
    {
        printf("Qual o nome do produto?\n:");
```

```

        setbuf(stdin, NULL);
        fgets(estoque[i].nome_produto, 30, stdin);
        printf("\n");
        printf("Qual o setor do produto?\n:");
        setbuf(stdin, NULL);
        fgets(estoque[i].setor, 30, stdin);
        printf("\n");
        printf("Qual a quantidade desse produto?\n:");
        setbuf(stdin, NULL);
        scanf("%d", &estoque[i].qtd_produtos[i]);
        printf("\n");
        printf("Qual o preco do produto?\nR$");
        setbuf(stdin, NULL);
        scanf("%f", &estoque[i].preco_prod[i]);
    }
}

int setor(struct estoque estoque[TAM];)
{
    char consulta[30];
    int cont = 0;
    printf("Qual o setor de consulta?\n");
    setbuf(stdin, NULL);
    fgets(consulta, 30, stdin);
    for(int i = 0; i<=TAM; i++)
    {
        if(strcmp(consulta, estoque[i].setor)==0)
        {
            cont++;
        }
    }
    printf("\n");
    printf("Neste setor tem: %i produtos", cont);
}

int investimento(struct estoque estoque[TAM];)
{
    float investimento[TAM];
    float soma = 0.0;
    float total[TAM];
    int j=0;
    for(int i = 0; i<=TAM; i++)
    {
        investimento[i] = soma + estoque[i].preco_prod[i]; //a operação n
        total[i] = investimento[i] * estoque[i].qtd_produtos[i];
        printf("\no total de capital em cada setor eh:\n");
        printf("\no setor %d o valor investido eh R$ %.2f\n", j++, total
[i]);
    }
}

```

```

    }

}

int main()
{
    int op, retorno;
    do
    {
        inicio:
        printf(">>>>>>>>>MENU<<<<<<<<<");
        printf("\n1-cadastro de produtos.");
        printf("\n2-consulta de produtos por setor");
        printf("\n3-consulta de capital investido no setor");
        printf("\n4-sair do programa");
        printf("\n\n");
        printf("Qual a opcao desejada?\n:");
        scanf("%i", &op);
        switch (op)
        {
            case 1:
                system("cls");
                cadastro();
                printf("\nVoce deseja voltar para o menu inicial?");
                printf("\n1-sim 2-nao");
                scanf("%i", &retorno);
                if(retorno == 1)
                {
                    system("cls");
                    goto inicio;
                }
                else
                {
                    system("cls");
                    printf("OBRIGADO!");
                    op = 4;
                }
                break;
            case 2:
                system("cls");
                setor();
                printf("\nVoce deseja voltar para o menu inicial?");
                printf("\n1-sim 2-nao");
                scanf("%i", &retorno);
                if(retorno == 1)
                {
                    system("cls");
                    goto inicio;
                }
            }
        }
    } while (op != 4);
}

```

```

    }
    else
    {
        system("cls");
        printf("OBRIGADO!");
        op = 4;
    }
    break;
case 3:
    system("cls");
    investimento();
    printf("\nVoce deseja voltar para o menu inicial?");
    printf("\n1-sim 2-nao");
    scanf("%i", &retorno);
    if(retorno == 1)
    {
        system("cls");
        goto inicio;
    }
    else
    {
        system("cls");
        printf("OBRIGADO!");
        op = 4;
    }
    break;
case 4:
    printf("saindo do programa.");
    printf("obrigado!");
    break;
}
} while (op != 4);
return 0;
}

```

4- Seja um sistema destinado a controlar despesas e dados de um condomínio de apartamentos.

Os dados de cada apartamento são armazenados em um vetor de estruturas. Para cada apartamento tem-se os seguintes dados:

- nome (do responsável)
- número (do apartamento)
- área (em m^2)
- número de moradores
- valor (a ser pago no mês)

Definir a estrutura acima

- a. Declarar um vetor de estruturas (global), supondo que o condomínio tem 40 apartamentos.
- b. Crie um menu para:
 - i. Escrever um bloco de instruções responsável por ler o vetor acima, exceto o campo valor.
 - ii. Escrever um bloco de instruções que retorne a área total do condomínio.
 - iii. Escrever um bloco de instruções que receba a área total do condomínio; leia o total de despesas do mesmo e calcule para cada apartamento o valor a ser pago no mês. Esse valor é proporcional à área do apartamento.
 - iv. Fazer um bloco de instruções para imprimir os dados do apartamento que tem o maior número de moradores. Em caso de haver mais de um, imprimir todos.
 - v. Escrever um bloco de instruções para sair do programa.

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#define TAM 40
struct apartamentos
{
    char nome[60];
    int numero_apt[TAM];
    int n_moradores[TAM];
    float preco[TAM];
    float area[TAM];
};
struct apartamentos apt[TAM];

int cadastro(struct apartamentos apt[TAM]);
{
    for(int i = 1; i<=TAM;i++)
    {
        printf("nome do responsavel do apartamento\n:");
        setbuf(stdin, NULL);
        fgets(apt[i].nome, 60, stdin);
```

```

        printf("numero do apartamento\n:");
        setbuf(stdin, NULL);
        scanf("%d", &apt[i].numero_apt[i]);
        printf("numero de moradores do apartamento\n:");
        setbuf(stdin, NULL);
        scanf("%d", &apt[i].n_moradores[i]);
        printf("Qual a area do seu apartamento em metros quadrados\n:");
        setbuf(stdin, NULL);
        scanf("%f", &apt[i].area[i]);
    }
}

int area_total(struct apartamentos apt[TAM];)
{
    float soma = 0.0;
    for(int i = 1; i<=TAM;i++)
    {
        soma = soma + apt[i].area[i];
    }
    printf("a soma da area do condominio eh: %.2f metros quadrados", soma
);
}

int maior_morador(struct apartamentos apt[TAM];)
{
    int maior;
    maior = apt[TAM].n_moradores[1];
    for(int i = 1; i<=TAM;i++)
    {
        if(apt[i].n_moradores[i] > maior)
        {
            maior = apt[i].n_moradores[i];
        }
    }
    printf("o maior numero de moradores eh: %i do apartamento numero %i
que tem %.2f metros quadrados", maior, apt[TAM].numero_apt[TAM], apt[TAM]
.area[TAM]);
}

int despesas(struct apartamentos apt[TAM];)
{
    float contas[TAM];
    float total[TAM];
    float area_total2;

    for(int i=1;i<=TAM;i++)
    {

```

```

        printf("\ndigite o valor em reais das despesas totais do apartame
nto por mes\nR$");
        setbuf(stdin, NULL);
        scanf("%f", &contas[i]);
        printf("digite a área total\n:");
        setbuf(stdin, NULL);
        scanf("%f", &area_total2);
        system("cls");
        total[i] = contas[i]/area_total2;
        printf("\n o valor a ser pago pelo apartamento %i eh R$ %.2f por
mes", i, total[i]);
    }
}
int main()
{
    int op, retorno;
    do
    {
        inicio:
        printf(">>>>>>>>>MENU<<<<<<<<<");
        printf("\n1-cadastro de apartamentos.");
        printf("\n2-area total do condominio");
        printf("\n3-despesas por apartamento");
        printf("\n4-apartamento com maior numero de pessoas");
        printf("\n5-sair do programa");
        printf("\n\n");
        printf("Qual a opcao desejada?\n:");
        scanf("%i", &op);
        switch (op)
        {
            case 1:
                system("cls");
                cadastro();
                printf("\nVoce deseja voltar para o menu inicial?");
                printf("\n1-sim 2-nao");
                scanf("%i", &retorno);
                if(retorno == 1)
                {
                    system("cls");
                    goto inicio;
                }
            else
            {
                system("cls");
                printf("OBRIGADO!");
                op = 4;
            }
            break;
            case 2:

```

```

        system("cls");
        area_total();
        printf("\nVoce deseja voltar para o menu inicial?");
        printf("\n1-sim 2-nao");
        scanf("%i", &retorno);
        if(retorno == 1)
        {
            system("cls");
            goto inicio;
        }
        else
        {
            system("cls");
            printf("OBRIGADO!");
            op = 4;
        }
        break;
    case 3:
        system("cls");
        despesas();
        printf("\nVoce deseja voltar para o menu inicial?");
        printf("\n1-sim 2-nao");
        scanf("%i", &retorno);
        if(retorno == 1)
        {
            system("cls");
            goto inicio;
        }
        else
        {
            system("cls");
            printf("OBRIGADO!");
            op = 4;
        }
        break;
    case 4:
        system("cls");
        maior_morador();
        printf("\nVoce deseja voltar para o menu inicial?");
        printf("\n1-sim 2-nao");
        scanf("%i", &retorno);
        if(retorno == 1)
        {
            system("cls");
            goto inicio;
        }
        else
        {
            system("cls");

```



```
        printf("OBRIGADO!");  
        op = 4;  
    }  
    break;  
case 5:  
    printf("saindo do programa.");  
    printf("obrigado!");  
    break;  
}  
}while(op != 5);  
}
```