



Atividade Avaliada 1 -

Data Entrega : até o dia 09/09 pelo sistema AVA

- 1 Implemente um sistema para controlar dois blocos de apartamentos, cada bloco contem 4 apartamentos. Abaixo, estão as estruturas (struct) para criar os tipos Pessoa, Apartamento, Data.

```
4  //Struct para definir pessoa
5  typedef struct pessoa{
6      char nome [20];
7      char cpf [14];
8  }Pessoa;
9
10 //struct para definir data
11 typedef struct data{
12     int dia,mes,ano;
13 }Data;
14
15 //struct para definir um apartamento
16 typedef struct apartamento{
17     char bloco; //codigo do bloco
18     int num; //numero apartamento
19     int andar; //andar do apartamento
20     int comodoss; //quantidade de peças
21     double vAluguel; //valor do aluguel
22     double vCondominio; //valor do condominio
23     int box; //número da garagem
24     Pessoa *locatario; //locatário do apto
25     Pessoa proprietario; //dono do apto
26     Data dataLocacao; //data da locacao
27     int status;
28 }Apto;
```

- 2 Detalhes das Estruturas para controle dos apartamentos.
 - 2.a A estrutura **Data** armazena as informações referente a uma data (dia, mês e ano);
 - 2.b A estrutura **Pessoa** armazena as informações referente aos dados de uma pessoa (nome e cpf);
 - 2.c A estrutura **Apto(Apartamento)** armazena as informações referente a um apartamento: bloco, número, andar, cômodos, valor de aluguel, valor de condomínio, número do box, uma referência para locatário, um registro do proprietário, uma data e o status do apartamento (1 locado, 0 para livre);
- 3 O sistema deve implementar as funções:
 - 3.a Função **defineData**, recebe os parâmetros dia,mes,ano e retorna o tipo Data;
 - 3.b Função **mostraData**, recebe por parâmetro um Data e mostra na tela a data;
 - 3.c Função **criaPessoa**, recebe os parâmetros relativos a um pessoa(nome,CPF), e retorna o tipo Pessoa
 - 3.d Função **mostraPessoa**, recebe um parâmetro tipo Pessoa e mostra os dados;
 - 3.e Função **numeroBox**, recebe o parâmetro tipo Apartamento e retorna o número do box(int);
 - 3.f Função **valorAluguel**, recebe o parâmetro tipo Apartamento e retorna o valor do aluguel(double);
 - 3.g Função **valorCondomínio**, recebe o parâmetro tipo Apartamento e retorna o valor do condomínio(double);
 - 3.h Função **totalCondomínio**, recebe um vetor de Apartamentos e retorna o somatório de todos os valores de condomínios dos apartamento do bloco recebido;
 - 3.i Função **criaApartamento**, recebe os parâmetros para definir um novo apartamento e retorna um ponteiro para o apartamento criado. Atenção, a referência para locador (Pessoa) deve ser inicializada com NULL, o Status com 0(zero) e a data padrão (0/0/0);
 - 3.j Função **mostraApartamento**, recebe o parâmetro tipo Apartamento e mostra os dados do apartamento;
 - 3.k Função **locacaoApartamento**, recebe o parâmetro tipo Apartamento, tipo Pessoa e Data, retorna o status (1 se o apartamento foi locado, 0 para falha de locação, caso o apartamento já esteja locado);
 - 3.l Função **mostraBloco**, recebe por parâmetro um vetor de Apartamentos e mostra os dados de todos os apartamentos;
 - 3.m Função **mostraBloco2**, recebe por parâmetro um ponteiro de ponteiro de Apartamento e mostra os dados de todos os apartamentos;
- 4 O programa principal deve realizar no mínimo o registro de 2 blocos (A,B) de apartamentos contendo 4 apartamentos cada:
 - 4.a Os blocos de apartamentos são vetores de ponteiros (4) para Apartamentos;
 - 4.b Registro de 3 pessoas, que serão proprietários ou locadores de apartamentos;
 - 4.c O sistema irá criar e inserir os apartamentos aos blocos;
 - 4.d O sistema deve realizar a locação de 3 apartamentos, por três pessoas;
 - 4.e O sistema deve mostrar todos os dados dos apartamentos de cada um dos blocos;
 - 4.f O sistema deve realizar a liberação de um apartamento (remover o locador), e locar para outro Pessoa;
 - 4.g Por fim, o sistema deve mostrar o total de arrecadação com a cota de condomínio dos apartamentos de cada bloco;

Observação:

O trabalho deve ser implementado em C, usando o arquivo enviado em Anexo (Atividade 1.C). Por fim deve ser enviado a plataforma AVA, até o prazo final.

Bom trabalho!