

MP2 - 2023.1

Seu objetivo será programar um sistema que organize dados do Censo Demográfico de vários lugares. Para isso, use as duas structs abaixo:

```
Typedef struct {  
    char nome[50];  
    char raca[50];  
    int idade;  
    float salario;  
    char CPF[12];  
} Pessoa;
```

```
Typedef struct {  
    char nome[50];  
    int codigo; //código único que  
identifica cada cidade  
    Pessoa* cidadaos;  
    int capacidade;  
} Cidade;
```

A cada nova cidade cadastrada, um elemento do tipo Cidade é adicionado. Ao registrar um novo cidadão, os dados dele devem ser armazenados no censo referente à cidade em que ele mora.

Seu programa deverá conter as seguintes funções:

Cidade* NovaCidade(**Cidade*** censo, **int*** tam): Adiciona uma nova cidade ao censo. Considere que o nome e código do local serão fornecidos pelo usuário.

void InserirOuAtualizar(**Cidade*** censo, **int** tam): Insere uma nova pessoa no censo da cidade informada pelo usuário. Se o CPF informado já estiver cadastrado, apenas atualize seus dados.

void Remover(**Cidade*** censo, **int** tam): Remove uma pessoa do vetor "cidadaos" de sua cidade. O código dessa cidade e o CPF da pessoa serão informados pelo usuário.

void Procura(**Cidade*** censo, **int** tam): Procura e printa na tela os dados da pessoa com o CPF e código da cidade fornecidos. Caso nada seja encontrado, avise o usuário que o CPF não está naquela cidade.

void ViewTable(**Cidade*** censo, **int** tam): mostra na tela os dados de todas as pessoas registradas no Censo Demográfico.

void Desaloca(**Cidade*** censo, **int** tam): Dá free em todos os ponteiros do censo.

Crie um menu para o usuário escolher se quer cadastrar mais uma cidade no sistema, procurar as informações de um CPF específico, ver todos os dados registrados, adicionar/atualizar os dados de uma pessoa ou remover alguém.

Ao fim do programa, chame "Desaloca" para liberar os ponteiros. Por fim, os dados fornecidos pelo usuário deverão ser pedidos dentro da devida função, a fim de deixar a *main* o mais simples possível.

Boa sorte!