

## LISTA 7 – REGISTROS

1. Fazer um programa para simular uma agenda de telefones. Para cada pessoa devem-se ter os seguintes dados:

- Nome
- E-mail
- Endereço (contendo campos para Rua, número, complemento, bairro, cep, cidade, estado, país).
- Telefone (contendo campo para DDD e número)
- Data de aniversário (contendo campo para dia, mês, ano)
- Observações: Uma linha (string) para alguma observação especial.

Para tanto, declarar os seguintes tipos de dados:

```
typedef struct End
{
    char rua[30];
    int numero;
    char complemento[30];
    char bairro[30];
    char cep[10];
    char cidade[30];
    char estado[3]; /*informar a sigla do estado*/
    char pais[3]; /*informar a sigla do pais*/
} Endereco;
```

```
typedef struct Telefone
{
    char ddd[5];
    char numfone[10];
} Fone;
```

```
typedef struct Aniversario
{
    int dia;
    char mes[10];
    int ano;
} Niver;
```

```
typedef struct Dados_Pessoa
{
    char nome[50];
    char email[30];
    Endereco residencia;
    Fone fonecasa;
    Niver nasci;
    char observacoes[100];
} Agenda;
```

Agora implementar o seguinte:

- Declarar uma variável global do tipo Agenda com capacidade de armazenar informações de até 100 pessoas.
- Implementar uma função que faz o cadastro de uma pessoa na agenda. A função deve receber como parâmetro o registro com os dados da pessoa e o número de cadastro na pessoa (esse número é sequencial e inicia em zero).
- Implementar uma função que faz uma busca na Agenda pelo nome. A função deve receber como parâmetro o nome que se deseja buscar e retornar por referência 1 se o nome foi encontrado e 0 caso contrário. Além disso, a função deve retornar, pelo retorno padrão, o registro com os dados do nome pesquisado.
- Implementar uma função que imprime o nome de todas as pessoas que fazem aniversário em um determinado mês. A função deve receber como parâmetro o mês para busca.
- Definir um bloco de instruções que deleta os dados de uma pessoa: retira todos os dados dessa pessoa e desloca todos os elementos seguintes do vetor para a posição anterior. A função deve receber como parâmetro de entrada o nome que se deseja deletar e retornar por referência o registro com os dados da pessoa que foi deletada da Agenda. A função deve retornar 1 se conseguiu deletar e 0 caso contrário. Observação: antes de deletar um registro é preciso primeiro buscar o registro que se deseja deletar.
- Implementar uma função Menu com os seguintes itens:
  - 1) Cadastrar na Agenda
  - 2) Buscar os dados de uma pessoa na Agenda
  - 3) Deletar os dados de uma pessoa da Agenda
  - 4) Sair

2. Crie uma estrutura representando os alunos da disciplina de Linguagem de Programação Estruturada. A estrutura deve conter a matrícula do aluno, nome, nota da primeira prova, nota da segunda prova e nota da terceira prova.

Você deve implementar um programa em C que:

- Permita ao usuário entrar com os dados de 5 alunos.
- Permita encontrar o aluno com maior nota da primeira prova.
- Permita encontrar o aluno com a maior média geral.
- Permita encontrar o aluno com a menor média geral.
- Para cada aluno diga se ele foi aprovado ou reprovado, considerando o valor 7 para aprovação.

3. Faça um programa em C que gerencie o estoque de um mercado e:

- Crie e leia um vetor de 5 produtos, com os dados: código (inteiro), nome (máximo 15 letras), preço e quantidade.
- Leia um pedido, composto por um código de produto e a quantidade. Localize este código no vetor e, se houver quantidade suficiente para atender ao pedido integralmente, atualize o estoque e informe o usuário. Repita este processo até ler um código igual a zero.
- Se por algum motivo não for possível atender ao pedido, mostre uma mensagem informando qual erro ocorreu.

Aplicar conceito de função, registro e passagem de parâmetro.