



**Técnicas de Programação**  
*[AN32F-IF62C]*

**PROFESSOR**

*Diogo Cezar Teixeira Batista*

# **LISTA DE EXERCÍCIOS - 02**

## **REGISTROS, ENUMS E PONTEIROS**

Cornélio Procópio

2017

# 1 INSTRUÇÕES

- Essa lista de exercícios deverá ser entregue em um arquivo único compactado no formato .zip ou .rar;
- Cada exercício deverá estar separado em um arquivo com o seguinte padrão de nomenclatura: L02\_EXON.c onde N é o número da questão;
- O trabalho é individual;
- Todo arquivo fonte deve ter como comentário o nome do aluno, da turma e o que o exercícios faz;
- Cada aluno poderá ser convocado para explicar um dos exercícios resolvidos da lista para o professor, ou para toda a turma;
- Trabalho iguais serão anulados;
- Não se esqueça de organizar, e indentar o seu código;
- Utilize nomes significativos para as suas variáveis;
- Escreva as instruções para o usuário do programa que está criando;

# 2 EXERCÍCIOS

1. Defina uma estrutura (struct) chamada FichaCliente com os seguintes campos: nome e rua do tipo vetor de char de 50 posições, número do tipo int, cidade e estado do tipo vetor de char de 30 posições e CEP do tipo vetor de char de 9 posições. Crie uma variável chamada cliente do tipo estrutura FichaCliente e faça a entrada dos dados, imprimindo depois na tela a entrada feita pelo usuário.
2. Crie uma estrutura (struct) chamada Contato, com os campos: nome do tipo vetor de char de 40 posições e telefone do tipo vetor de char de 12 posições. Crie duas variáveis chamadas contato1 e contato2 do tipo Contato. Faça a leitura dos dados da variável contato1 e depois copie o conteúdo de contato1 para contato2, logo após isto imprima na tela as informações de contato2.

3. Faça um programa que crie uma estrutura de dados (struct) chamada Aluno. Esta estrutura deve conter um campo chamado nome, do tipo vetor de char, e quatro campos do tipo int: Nota1, Nota2, Nota3 e Nota4. Defina um vetor de 10 posições do tipo da estrutura Aluno. Faça a entrada de todos os dados para as 10 posições e depois imprima na tela estas informações em formato de colunas.
4. Passagem de dados por referência. Crie um programa que leia dois valores inteiros na função principal, depois chame uma função chamada Troca passando as duas variáveis inteiras lidas como parâmetros por referência. Esta função irá inverter o valor das duas variáveis. Depois na função principal imprima na tela os novos valores das duas variáveis.
5. Retorno de dados em funções. Na função main crie três variáveis inteiras chamadas Raiz, A e Num, e um ponteiro do tipo inteiro chamado Exp inicializado com NULL. Faça Exp receber o endereço de A. Pergunte ao usuário um número e armazene em Num, depois chame uma função chamada Contas passando a variável Raiz por referencia, o ponteiro Exp e a variável Num por Valor. Esta função deve calcular a Raiz do valor armazenado em Num e armazenar em Raiz, calcular o valor de Num elevado a 2 e armazenar no endereço apontado por Exp. Depois na função principal imprima o valor da variável Raiz e o valor apontado por Exp.
6. Crie um programa que deverá receber como entrada um número de 1 a 12. Valide para que apenas este intervalo seja aceito. O programa deverá ter 2 enums: Meses e Estacoes. O primeiro deverá ser um enum com cada mês do ano: JANEIRO = 1, FEVEREIRO, MARCO, ABRIL, MAIO, JUNHO, JULHO, AGOSTO, SETEMBRO, OUTUBRO, NOVEMBRO, DEZEMBRO. O segundo deverá ter as estações do ano: PRIMAVERA = 1, VERÃO, OUTONO, INVERNO. Crie ainda definições de tipo: Meses → tipoMes e Estacoes → tipoEstacap. O programa deverá utilizar o número digitado para criar uma nova variável do tipo tipoMes, e passá-la para uma função chamada validaEstacao que verifica qual mês foi passado e retorna outro enum com a estação correspondente (tipoEstacao). Utilize a variável retornada pela função validaEstacao na sua função main, para imprimir a estação utilizando o comando switch. Considere: Primavera → de março a junho, Verão → de junho a setembro, Outono → de setembro a dezembro, Inverno → de dezembro a março.