

Lista de Exercícios 07 – Linguagem C
Assuntos: “*structs*, *enums*, *typedef* e funções”

CURSO: ENGENHARIA DA COMPUTAÇÃO – UFC – CAMPUS DE SOBRAL

DISCIPLINA: PROGRAMAÇÃO COMPUTACIONAL

PROFESSOR: FERNANDO RODRIGUES DE ALMEIDA JÚNIOR

ALUNO: _____ DATA: ____/____/2025

Dados os seguintes exercícios, escreva programas em ANSI C89 para resolver tais questões:

1. Implemente um programa que crie um novo tipo ***pessoa***, que será uma estrutura para armazenar o nome (com até 40 caracteres), a idade, o CPF (com 11 dígitos) e o estado civil (este só podendo assumir um dos seguintes valores: ‘solteiro’, ‘casado’, ‘separado’ ou ‘viúvo’) de uma pessoa e então leia e armazene esses dados em uma estrutura. Em seguida, imprima na tela os dados da estrutura lida, mostrando primeiro o CPF, depois o nome, o estado civil e finalmente a idade.
2. Desenvolva uma função chamada ***potencia*** que receba dois números inteiros (x e y) e, sem utilizar outra função qualquer (nem a biblioteca *math.h*) e usando o princípio das multiplicações sucessivas para cálculo de potenciação, retorne o valor de x elevado a y. Escreva um programa que solicite ao usuário os respectivos valores dos parâmetros e chame a função criada para mostrar o resultado do cálculo ao usuário.
3. O fatorial de um número inteiro positivo N, denotado por N!, é definido como o produto dos inteiros positivos menores do que ou iguais a N. Por exemplo $4! = 4 \times 3 \times 2 \times 1 = 24$. Desenvolva uma função chamada ***fatorial*** que receba um número inteiro (x) e, sem utilizar outra função qualquer (nem a biblioteca *math.h*) retorne um inteiro longo sem sinal (*unsigned long long int*) constando do resultado do cálculo do fatorial do valor x. Escreva um programa que solicite ao usuário os respectivos valores dos parâmetros e chame a função criada para mostrar o resultado do cálculo ao usuário.
4. A seguinte sequência de números 0 1 1 2 3 5 8 13 21... é conhecida como *Série de Fibonacci*. Nessa sequência, cada número, depois dos 2 primeiros (fixados em 0 e 1), é igual à soma dos 2 anteriores. Escreva um algoritmo que defina uma função “*NFibonacci*” que receba como parâmetro um inteiro positivo N ($0 < N < 46$) e retorne um inteiro sendo o N-ésimo termo da *Série de Fibonacci*. No corpo da função “*main*”, leia um valor inteiro positivo (no mesmo padrão de N acima) e crie um laço que chame a função criada (*NFibonacci*) para os valores de 1 a N, mostrando os N primeiros números dessa série.
5. Desenvolva uma função em C que receba como entrada os valores reais das incógnitas a, b e c de uma função matemática do 2º grau e, usando a Fórmula de Báskara, calcule e imprima os valores das 2 raízes (x’ e x”) de tal função. Tal função deve ainda retornar “verdadeiro” (caso a função tenha raíz(es) real(is)) ou “falso” (caso as raízes não sejam reais).
6. Desenvolva uma função em C que receba como entrada um valor / número real (fracionário) em decimal (como um *float*) e retorne a representação deste número em binário: devendo ser exibida na tela a partir do valor retornado como um “inteiro binário” – que deverá ser representado como uma estrutura com dois campos sendo arrays de inteiros – um campo representando a parte inteira (*p_int*) com até 10 bits (cada bit só pode ser ‘0’ ou ‘1’), e um segundo campo representando a parte fracionária (*p_frac*) com até 4 bits.

Bom trabalho!