



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA – IFCE
CAMPUS MARACANAÚ
CURSO TÉCNICO EM INFORMÁTICA
INTRODUÇÃO À PROGRAMAÇÃO – 2014.1
LISTA DE EXERCÍCIOS 8

ALUNO: _____

PROFESSOR: JEAN MARCELO

1. Faça um programa em C que solicite N números e imprima o resultado da soma dos mesmos. A operação deve ser feita por uma função que retorna o resultado, tendo como argumento um **vetor** com os números a serem somados.
2. Faça um programa em C que calcule a área de N esferas em metros quadrados e imprima os resultados. A área deve ser calculada através de uma função, cujos os argumentos sejam um **vetor** com os raios das esferas e um **vetor** que armazenará as áreas calculadas dentro da função. A área de uma esfera é dada por $\text{área} = 4 \times PI \times \text{raio}^2$, com $PI = 3,14$.
3. Faça um programa em C que converta uma *string* com somente letras minúsculas para uma *string* com somente letras maiúsculas. A conversão deve ser feita por uma função, cujo argumento é uma *string*, que será alterada dentro da função.
4. Escreva um programa em C que tenha uma função com o seguinte protótipo: **void trocaCaractere(char nome[], char a, char b)**. Os argumentos passados para a função devem ser lidos do teclado usando as funções **gets()** e **getchar()**. O primeiro argumento é uma *string*, o segundo argumento é o caractere que deve ser substituído na *string* pelo caractere do terceiro argumento. A função deve imprimir na tela a *string* resultante da substituição de caracteres, utilizando a função **puts()**.
5. Escreva um programa em C que leia N números inteiros do teclado e os imprima em ordem crescente. Um **vetor** contendo os números deve ser passado como argumento de uma função que fará a ordenação e imprimirá os resultados.
6. Escreva um programa em C que solicite a quantidade N de alunos de uma turma. Em seguida, para cada aluno, leia duas notas, faça a média ponderada das notas e imprima, além da média individual de cada aluno, “Aprovado” se a média for igual ou superior a 6,0, senão imprima “Reprovado”. A média ponderada deve ser calculada através de uma função, cujos argumentos são um vetor de notas 1, um vetor de notas 2 dos alunos e um vetor de médias, que serão calculadas pela função. Ao final, o programa deve imprimir as médias individuais dos alunos e a média aritmética geral da turma e o usuário deve ter a opção de consultar as notas e a média de um aluno em particular, informando o número do aluno. O programa deve permanecer executando até que o usuário tecle 'q' para sair. Observação: a média ponderada de cada aluno é dada por $\text{média ponderada do aluno}_n = \frac{2 \times \text{nota1} + 3 \times \text{nota2}}{5}$.
7. Faça um programa em C que solicite o número de temperaturas a serem lidas em graus Celsius e imprima para cada temperatura de entrada o valor correspondente em graus Fahrenheit. Imprima ao final a média aritmética das temperaturas em graus Celsius e em graus Fahrenheit com duas casas decimais. Se a média das temperaturas estiver no intervalo de 18 a 20 graus Celsius imprima: “Temperatura agradável.”. Se a média das temperaturas estiver abaixo de 18 graus Celsius imprima: “Está um pouquinho frio.”. Se a média das temperaturas estiver acima de 20 graus Celsius imprima: “Está quente.”. Utilize no laço de

cálculo a estrutura de repetição **while**. A conversão das temperaturas deve ser feita por uma função, que retorna o valor em graus Fahrenheit. A média das temperaturas deve ser calculada por uma outra função cujo o argumento é um vetor com as temperaturas e o retorno é a média delas. O programa deve permanecer executando enquanto o usuário não solicitar para sair, teclando 'q'. Fórmula de conversão: $T_f = \frac{T_c \times 9}{5} + 32$, em que T_c e T_f são, respectivamente, a temperatura em graus Celsius e a temperatura em graus Fahrenheit.

8. Implemente um programa em C que leia duas *strings* do teclado, utilizando a função **gets()**, e imprima: “As *strings* são iguais”; ou “As *strings* são diferentes”, conforme o caso. O programa deve também informar se a primeira *string* tem um número de caracteres maior, menor ou igual aos da segunda *string*. As *strings* devem ser passadas como argumentos de uma função que imprimirá os resultados. Utilize as funções de manipulação de *strings* da biblioteca **string.h**.
9. Escreva um programa em C que leia uma *string* de até 30 caracteres, utilizando a função **gets()** e, em seguida, a imprima na tela utilizando a função **puts()**. Deve ser solicitado ao usuário se o mesmo deseja contar: (A) - Número total de caracteres da *string*; (B) – Número de letras maiúsculas; (C) – Número de letras minúsculas. O programa deve utilizar uma função com dois argumentos, conforme o seguinte protótipo: **int contaLetras(char s[], char opcao)**. O primeiro argumento é a *string*, o segundo é a opção de contagem. A função retorna o número de caracteres contados para o programa principal de acordo com a opção desejada, sendo em seguida impresso na tela pelo programa principal. Lembre que toda *string* é finalizada com o caractere '\0'.
10. Faça um programa em C que leia duas matrizes quadradas de números inteiros e imprima o resultado da soma das duas matrizes. As matrizes devem ser passadas como argumentos de uma função que imprimirá o resultado. Utilize matrizes bidimensionais do C. Exemplo: **int matriz[i][j];**
11. Implemente um programa em C que leia *N* números e os imprima em ordem **decrescente**.
12. Implemente um programa em C que leia *N* números e depois imprima o **maior** número.
13. Faça um programa em C que leia uma *string* e a copie para outra *string* em ordem inversa. Imprima as duas *strings*.
14. Faça um programa que leia uma *string* do teclado com a função **gets()**. A *string* deve ser passada como argumento de uma função que irá retornar o número de palavras da *string*.