

Quinta lista de exercícios de Introdução à Programação de Computadores

Para todos os exercícios desta lista:

- Elabore um conjunto significativo de casos de testes.
- Implemente o programa usando GCC no Ubuntu.
- Use o padrão ISO C 2011 (C11).
- Teste o programa usando os dados do item *a*.

Exercício 1

1. Declare uma matriz 5 x 5. Preencha com 1 a diagonal principal e com 0 os demais elementos. Escreva ao final a matriz obtida.
2. Leia uma matriz 4 x 4, imprima a matriz e retorne a localização (linha e a coluna) do maior valor.
3. Gerar e imprimir uma matriz de tamanho 10 x 10, onde seus elementos são da forma:

$$A[i][j] = \begin{cases} 2i + 7j, & \text{se } i < j \\ 3i^2 - 1, & \text{se } i = j \\ 4i^3 - 5j^2 + 1, & \text{se } i > j \end{cases}$$

4. Leia uma matriz 10 x 3 com as notas de 10 alunos em 3 provas. Em seguida, escreva o número de alunos cuja pior nota foi na prova 1, o número de alunos cuja pior nota foi na prova 2, e o número de alunos cuja pior nota foi na prova 3. Em caso de empate das piores notas de um aluno, o critério de desempate é arbitrário, mas o aluno deve ser contabilizado apenas uma vez.
5. Faça um programa que leia uma matriz de 5 linhas e 4 colunas contendo as seguintes informações sobre alunos de uma disciplina, sendo todas as informações do tipo inteiro:
 - Primeira coluna: número de matrícula (use um inteiro)
 - Segunda coluna: média das provas
 - Terceira coluna: média dos trabalhos
 - Quarta coluna: nota final

Elabore um programa que:

- a) Leia as três primeiras informações de cada aluno
- b) Calcule a nota final como sendo a soma da média das provas e da média dos trabalhos

- c) Imprima a matrícula do aluno que obteve a maior nota final (assuma que só existe uma maior nota)
 - d) Imprima a média aritmética das notas finais
6. Faça um programa que leia uma matriz A de tamanho 3×3 e calcule $B = A^2$.
 7. Faça programa que leia uma matriz 3×6 com valores reais.
 - a) Imprima a soma de todos os elementos das colunas ímpares.
 - b) Imprima a média aritmética dos elementos da segunda e quarta colunas.
 - c) Substitua os valores da sexta coluna pela soma dos valores das colunas 1 e 2.
 - d) Imprima a matriz modificada.
 8. Crie um programa que calcula o comprimento de uma string (não use a função `strlen`).
 9. Entre com um nome e imprima o nome somente se a primeira letra do nome for 'a' (maiúscula ou minúscula).
 10. Faça um programa que receba uma palavra e a imprima de trás-para-frente.
 11. Faça um programa que receba uma palavra e calcule quantas vogais (a, e, i, o, u) possui essa palavra. Entre com um caractere (vogal ou consoante) e substitua todas as vogais da palavra dada por esse caractere.
 12. Leia uma cadeia de caracteres e converta todos os caracteres para maiúscula. Dica: subtraia 32 dos caracteres cujo código ASCII está entre 97 e 122.
 13. Escreva um trecho de código para fazer a criação dos novos tipos de dados conforme solicitado abaixo:
 - Horário: composto de hora, minutos e segundos.
 - Data: composto de dia, mês e ano.
 - Compromisso: composto de uma data, horário e texto que descreve o compromisso.
 14. Implemente um programa que leia o nome, a idade e o endereço de uma pessoa e armazene os dados em uma estrutura.
 15. Construa uma estrutura aluno com nome, número de matrícula e curso. Leia do usuário a informação de 5 alunos, armazene em arranjo dessa estrutura e imprima os dados na tela.
 16. Faça um programa que armazene em um registro de dados (estrutura composta) os dados de um funcionário de uma empresa, compostos de: Nome, Idade, Sexo (M/F), CPF, Data de Nascimento, Código do Setor onde trabalha (0-99), Cargo que ocupa (string de até 30 caracteres) e Salário. Os dados devem ser digitados pelo usuário, armazenados na estrutura e exibidos na tela.
 17. Faça um programa que leia os dados de 10 alunos (Nome, matrícula, Média Final), armazenando em um arranjo. Uma vez lidos os dados, divida estes dados em 2 novos arranjos, o arranjo dos aprovados e o arranjo dos reprovados, considerando a média mínima para aprovação como sendo 5.0. Exibir na tela os dados do arranjo de aprovados, seguido dos dados do arranjo de reprovados.
 18. Faça um programa que seja uma agenda de compromissos e:
 - Crie e leia um arranjo de 5 estruturas de dados com: compromisso (máximo 60 letras) e data. A data deve ser outra estrutura de dados contendo dia, mês e ano.

- Leia dois inteiros M e A e mostre todos os compromissos do mês M do ano A. Repita o procedimento até ler $M = 0$.
19. Faça uma função que receba a data atual (dia, mês e ano em inteiro) e exiba-a na tela no formato textual por extenso. Exemplo: Data: 01/01/2000, Imprimir: 1 de janeiro de 2000.
 20. Faça uma função que receba 3 números inteiros como parâmetro, representando horas, minutos e segundos, e os converta em segundos.
 21. Escreva uma função que receba um número inteiro maior do que zero e retorne a soma de todos os seus algarismos. Por exemplo, ao número 251 corresponderá o valor 8 ($2+5+1$). Se o número lido não for maior do que zero, o programa terminará com a mensagem “Número inválido”.
 22. Faça uma função que retorne o maior fator primo de um número.
 23. Faça uma função que receba um arranjo de inteiros e retorne quantos valores pares ele possui.
 24. Faça uma função que receba como parâmetro um arranjo X de 30 elementos inteiros e retorne, também por parâmetro, dois arranjos A e B. O arranjo A deve conter os elementos pares de X e o arranjo B, os elementos ímpares.
 25. Crie uma função recursiva que receba um número inteiro positivo N e calcule o somatório dos números de 1 a N.
 26. Faça uma função recursiva que permita somar os elementos de um arranjo de inteiros.

Exercício 2

1. Escreva um programa que contenha duas variáveis inteiras. Compare seus endereços e exiba o maior endereço.
2. Implemente uma função que calcule a área da superfície e o volume de uma esfera de raio R. Essa função deve obedecer ao protótipo:

```
void calc_esfera(float R, float *area, float *volume)
```

A área da superfície e o volume são dados, respectivamente, por:

$$A = 4\pi R^2$$

$$V = \frac{4}{3}\pi R^3$$

3. Escreva uma função que aceite como parâmetro um arranjo de inteiros com N valores, e determina o maior elemento do arranjo e o número de vezes que este elemento ocorreu no arranjo. Por exemplo, para um arranjo com os seguintes elementos: 5, 2, 15, 3, 7, 15, 8, 6, 15, a função deve retornar para o programa que a chamou o valor 15 e o número 3 (indicando que o número 15 ocorreu 3 vezes). A função deve ser do tipo void.
4. Crie uma função para somar dois arranjos. Esta função deve receber dois arranjos e retornar a soma em um terceiro arranjo. Caso o tamanho do primeiro e segundo arranjo seja diferente então a função retornará ZERO (0). Caso a função seja concluída com sucesso a mesma deve retornar o valor UM (1). Utilize aritmética de ponteiros para manipulação do arranjo.

5. Escreva uma função que receba um arranjo de inteiros `V` e os endereços de duas variáveis inteiras, `min` e `max`, e armazene nessas variáveis o valor mínimo e máximo do arranjo. Escreva também uma função `main` que use essa função.
6. Escreva uma função que receba uma string e troque todo caractere após um branco pelo seu equivalente maiúsculo.
7. Escreva um programa que:
 - a) Crie ou abra um arquivo texto de nome `arq.txt`.
 - b) Permita que o usuário grave diversos caracteres nesse arquivo, até que o usuário entre com o caractere `'0'`
 - c) Feche o arquivoAgora, abra e leia o arquivo, caractere por caractere, e escreva na tela todos os caracteres armazenados.
8. Faça um programa que receba do usuário um arquivo texto e mostre na tela quantas linhas esse arquivo possui.
9. Faça um programa que receba do usuário um arquivo texto e mostre na tela quantas letras são vogais.
10. Faça um programa que receba do usuário um arquivo texto e um caractere. Mostre na tela quantas vezes aquele caractere ocorre dentro do arquivo.
11. Faça um programa que receba dois arquivos do usuário, e crie um terceiro arquivo com o conteúdo dos dois primeiros juntos (o conteúdo do primeiro seguido do conteúdo do segundo).
12. Faça um programa que receba o nome de um arquivo de entrada e outro de saída. O arquivo de entrada contém em cada linha o nome de uma cidade (ocupando 40 caracteres) e o seu número de habitantes. O programa deverá ler o arquivo de entrada e gerar um arquivo de saída onde aparece o nome da cidade mais populosa seguida pelo seu número de habitantes.
13. Faça um programa que permita que o usuário entre com diversos nomes e telefone para cadastro, e crie um arquivo com essas informações, uma por linha. O usuário finaliza a entrada com `'0'` para o telefone.
14. Faça um programa que recebe um arranjo de 10 números, converta cada um desses números para binário e grave a sequência de 0s e 1s em um arquivo texto. Cada número deve ser gravado em uma linha.
15. Faça um programa que receba do usuário um arquivo que contenha o nome e a nota de diversos alunos (da seguinte forma: NOME: JOÃO NOTA: 8), um aluno por linha. Mostre na tela o nome e a nota do aluno que possui a maior nota.
16. Assuma que a variável `direcao` seja declarada desta forma:

```
enum {NORTE, SUL, LESTE, OESTE} direcao;
```

Sejam `x` e `y` variáveis `int`. Escreva um comando `switch` que teste o valor de `direcao`, incrementando `x` se `direcao` for `LESTE`, decrementando `x` se `direcao` for `OESTE`, incrementando `y` se `direcao` for `SUL`, decrementando `y` se `direcao` for `NORTE`.