



Instituto Federal de Educação, Ciência e Tecnologia de Brasília – Câmpus Taguatinga
Ciência da Computação – Estruturas de Dados e Algoritmos
Lista de Exercícios I – Revisão de Algoritmos
Prof. Daniel Saad Nogueira Nunes

Aluno: _____

Matrícula: _____

Exercício 1

Dado o critério de aprovação da nossa disciplina de ED1, desenvolva um programa que leia do usuário as respectivas notas e o percentual de frequência e verifique se o aluno foi aprovado ou não.

Exercício 2

Um número primo natural é aquele que pertence aos naturais, é maior que 1 e divisível somente por ele mesmo e por 1. Elabore um programa que imprima os números primos naturais. Note que seu programa deverá ser um loop infinito, uma vez que temos infinitos números primos naturais.

Exercício 3

Seja a sequência de Fibonacci:

$$F(n) = \begin{cases} 1, & 0 \leq n \leq 1 \\ F(n-1) + F(n-2), & n \geq 2 \end{cases} \quad (1)$$

Crie um programa que dado um n lido pelo usuário calcule $F(n)$ recursivamente.

Exercício 4

Repita o mesmo do exercício anterior só que utilizando laços de repetição em vez de recursão.

Exercício 5

Nosso sistema monetário consiste das moedas de 1, 5, 10, 25, 50 e 100 centavos. Projete um algoritmo que leia a quantidade de troco em centavos que um atendente deve dar e imprima a quantidade mínima de moedas para que ele dê o troco. Por exemplo, se o troco for 83 centavos, são necessários no mínimo 6 moedas (uma de 50, uma de 25, uma de 5 e três de 1).

Exercício 6

Projete o algoritmo da questão anterior de maneira recursiva.

Exercício 7

Elabore um programa que leia dois vetores de inteiros V_1 e V_2 de N e M elementos respectivamente, onde N e M devem ser informados previamente pelo usuário e retorne um novo vetor V_3 contendo:

- A interseção dos dois vetores. A interseção não deve conter números repetidos, e deve conter números que ocorrem simultaneamente nos vetores V_1 e V_2 .

Observação: como N e M são lidos do usuário, deverá ser utilizada alocação dinâmica de memória para criar os vetores.

Exercício 8

Repita o exercício anterior, mas para considerando nomes em vez de inteiros. Suponha que os nomes podem ter no máximo até 20 caracteres.

Exercício 9

Desenvolva um programa que leia um inteiro N a ser informado pelo usuário e uma string com N caracteres e verifique se a mesma é um palíndromo.

Observação: como N é lido do usuário, a string deverá ser alocada dinamicamente.

Exercício 10

Suponha a seguinte struct:

```
typedef struct pessoa_t{
    char nome[20];
    char cpf[20];
    char sexo; /*M ou F*/
    char estado_civil[20]; /*solteiro,casado,divorciado*/
    double salario; /*Salario do cidadao*/
    int idade;
    int nro_filhos; /*Numero de filhos do cidadao*/
}pessoa_t;
```

Faça um programa que permita:

- Cadastrar uma pessoa e salvar os dados em um arquivo informado pelo usuário.
- Leia um nome de arquivo do usuário e imprima todas as pessoas cadastradas.
- Leia um nome de arquivo do usuário e um CPF e imprima os dados da pessoa cadastrada com o CPF informado.
- Leia um nome de arquivo do usuário e imprima a média aritmética simples dos salários das pessoas com mais de 1 filho.
- Leia um nome de arquivo do usuário e imprima a média aritmética simples das idades das pessoas de sexo feminino que tenham pelo menos 3 filhos.

Exercício 11

Abaixo segue uma lista de problemas recomendados do URI Online Judge:

- <https://www.urionlinejudge.com.br/judge/pt/problems/view/2030>

-
- <https://www.urionlinejudge.com.br/judge/pt/problems/view/1180>
 - <https://www.urionlinejudge.com.br/judge/pt/problems/view/1383>
 - <https://www.urionlinejudge.com.br/judge/pt/problems/view/1253>