

## Lista de Exercícios 5 – Arquivos

1) Considere um arquivo chamado `paciente.txt` que armazena os dados dos 10 pacientes de um endocrinologista. O arquivo é organizado onde cada paciente tem seguintes dados:

**<nome do paciente>**  
**<idade> <peso> <altura>**

Faça um programa que leia os dados e guarde em um arquivo chamado `acimadopeso.txt` os nomes e as idades dos pacientes que estão acima do peso ideal fornecido pela relação  $\text{PesoIdeal} = (\text{altura} - 1) \times 100$

2) Faça um programa que leia o número de alunos de uma turma. O seu programa deverá ler os nomes e as respectivas notas finais obtidas pelos alunos. A saída do seu programa será um arquivo chamado `aprovado.txt` que conterá os nomes dos alunos que obtiveram nota final igual ou superior a 5. Monte o arquivo com o formato que achar mais adequado.

3) Faça o exercício anterior considerando vários arquivos que armazenam as informações dos alunos de várias turmas distintas. Considere que os nomes de todos os arquivos a serem processados encontra-se num arquivo principal chamado `arquivos.txt`. Ao armazenar os nomes dos alunos aprovados no mesmo arquivo `aprovado.txt`, indique na frente do nome de qual arquivo este aluno veio.

4) Considere que exista um arquivo chamado `voo.txt` que armazena 10 voos entre várias cidades. Cada voo contém as seguintes informações:

**<empresa aérea> <número do voo> <origem> <destino> <valor da passagem>**

Empresa aérea é um código de 2 letras, o número do voo tem 4 algarismos, origem e destino são códigos com 3 letras e o valor da passagem é um número real. Faça um programa para informar ao interessado na compra de uma passagem se existem voos entre as cidades fornecidas pelo teclado. A saída do seu programa, quando não existir o voo desejado será "NÃO EXISTEM VOOS". Caso existam, imprimir a empresa, o número do voo e o valor da passagem. Pode haver mais de um voo entre as cidades desejadas.

5) Dado um arquivo com o seguinte conteúdo:

```
1,Universidade Federal do Espírito Santo,Vitória,2006,12,15,40.5
2,Universidade Federal de Minas Gerais,Belo Horizonte,2006,12,01,90.3
3,Universidade Federal do Rio de Janeiro,Rio de Janeiro,2007,01,29,77.8
4,Universidade Federal do Rio Grande do Sul,Porto Alegre,2007,01,16,85.9
5,Universidade Federal de Pernambuco,Recife,2006,11,18,53.0
```

Crie um programa que separe e imprima as diferentes informações contidas nestas *strings*, sendo:

- Código (inteiro);
- Nome da universidade (*string*);
- Cidade onde será realizado o vestibular (*string*);
- Ano (inteiro), mês (inteiro) e dia (inteiro) da data que será realizado o vestibular;
- Razão de alunos por vaga (real).

Exemplo:

```
Código: 0001
Universidade: Universidade Federal do Espírito Santo
```



**Local do vestibular: Vitória**  
**Data do vestibular: 15/12/2006**  
**Alunos por vaga: 40.5**

Dica: leia o arquivo linha por linha e separe os dados usando a função `strtok()`, como no exemplo abaixo:

```
#include <stdio.h>
#include <string.h>

main() {
    char linha[] = "Um,Dois,Três,Quatro";
    char delim[] = ",";
    char *trecho;

    trecho = strtok(linha, delim);
    while (trecho != NULL) {
        printf("%s\n", trecho);
        trecho = strtok(NULL, delim);
    }
}
```

6) Considere um arquivo chamado `matrizes.txt` que armazena 5 matrizes quadradas de dimensão  $4 \times 4$  de números inteiros. Faça um programa para ler as matrizes do arquivo e criar um arquivo chamado `transp.txt` que conterà as matrizes transpostas das matrizes lidas.

7) Seja o arquivo chamado `mat8x8.txt` que armazena 5 matrizes de dimensão  $8 \times 8$  de componentes reais. Faça um programa para ler essas matrizes e verifique se tais matrizes são diagonalmente dominante, isto é, se a seguinte condição é satisfeita:

$$\forall i: 1 \leq i \leq 8, |a_{ii}| > \sum_{j=1, j \neq i}^8 |a_{ij}|$$