

# Contabilizando a Frequência de Alunos

Professora Patrícia Dockhorn Costa, patricia.d.costa@ufes.br

---

## Descrição

Considere um programa que tem por objetivo contabilizar as **presenças e faltas** de alunos de uma turma da UFES. O programa recebe como entrada: (a) um arquivo de configuração com o número **N** de aulas lecionadas no semestre e os nomes dos alunos matriculados; (b) **N arquivos** contendo a presença dos alunos da turma em cada aula (onde **N** é o número de aulas lecionadas).

Considere o arquivo a seguir, de configuração de entrada (entrada.txt), onde o número **N** na primeira linha representa o **número de aulas** (que corresponde ao número de arquivos contendo as chamadas) e as linhas consecutivas contém os nomes dos alunos da turma:

```
N 5  
BRUNO  
DENER  
DENIS  
FABIO  
FELLIPE  
FERNANDO  
FILIPE  
FLAVIO  
GABRIEL  
IGOR  
JOAO  
LEONARDO  
LUCAS  
LUIZ  
MATHEUS  
OTAVIO  
PABLO  
PEDRO  
TASSIO
```

Portanto, para o exemplo acima, seu programa vai precisar contabilizar a frequência de 19 alunos, em 5 aulas. As frequências de cada aula estarão em arquivos separados.

O arquivo a seguir mostra um exemplo de um arquivo de frequência (e.x., entrada1.txt), de uma determinada aula (P para presença e F para falta):

```
BRUNO P
DENER P
DENIS P
FABIO F
FELLIPE F
FERNANDO F
FILIPE P
FLAVIO P
GABRIEL P
IGOR F
JOAO F
LEONARDO F
LUCAS P
LUIZ F
MATHEUS P
OTAVIO F
PABLO F
PEDRO P
TASSIO P
```

Escreva um programa que leia as informações de um arquivo de configuração chamado “entrada.txt”, e leia os outros arquivos com as presenças dos alunos em cada aula. Os arquivos serão nomeados de acordo com a sequência das aulas. Se N for 5, por exemplo, serão considerados 5 arquivos de chamada: entrada1.txt, entrada2.txt, entrada3.txt, entrada4.txt e entrada5.txt.

O arquivo de saída (saida.txt) deve fornecer as informações contabilizadas sobre faltas e presenças:

```
BRUNO 3P 2F
DENER 5P 0F
DENIS 4P 1F
FABIO 1P 4F
FELLIPE 2P 3F
FERNANDO 3P 2F
FILIPE 5P 0F
FLAVIO 1P 4F
GABRIEL 5P 0F
IGOR 4P 1F
JOAO 3P 2F
LEONARDO 1P 4F
LUCAS 5P 0F
LUIZ 4P 1F
MATHEUS 3P 2F
OTAVIO 4P 1F
PABLO 0P 5F
PEDRO 5P 0F
TASSIO 2P 3F
```

**Utilize uma Tabela Hash** como estrutura de dados para armazenamento e busca de dados. O tratamento de colisão deve ser feito com **listas encadeadas**.

#### **Sugestão de fluxo:**

- Ler o arquivo “entrada.txt”; guardar valor de N; dimensionar tamanho da tabela hash com base no número de alunos;
- Iniciar a tabela e inserir os alunos, usando uma função de hash adequada para isso (caso haja colisão, usar uma lista encadeada);
- Ler cada um dos arquivos de frequência (cujos nomes serão gerados de acordo com o valor de N). Para cada arquivo, ler linha a linha e faça:
  - Dado o nome do aluno e informação P ou F, busca na tabela e contabiliza frequência do aluno;
- Imprimir as informações da tabela em um arquivo saida.txt (não precisa estar ordenado por nome do aluno);
- Liberar toda a memória alocada.

#### **Regras importantes:**

1. Deve-se usar uma **Tabela Hash** como estrutura de dados para armazenamento e busca de dados. O tratamento de colisão deve ser feito com **listas encadeadas**;
2. O nome do aluno deve ser considerado como chave de busca (ou seja, deve ser usado como entrada para a função de hash);

3. Somente nomes simples serão utilizados;
4. Utilize tipos de dados estruturados;
5. As turmas da UFES têm no máximo 100 alunos;
6. NÃO utilize variáveis globais;
7. Libere toda a memória alocada (use valgrind);
8. O arquivo com a função main deve ter o seu nome: <nome>.c;
9. Escreva seu nome em TODOS os arquivos produzidos;
10. Deixe os arquivos na pasta RESPOSTA;

**BOA PROVA!**

## Ajudinha extra!

- Usando fscanf para ler dados de um arquivo:

```
#include <stdio.h>

int main()
{
    FILE* ptr = fopen("testes.txt", "r");
    if (ptr == NULL) {
        printf("no such file.");
        return 0;
    }

    char nome[100];
    char presenca;
    while (fscanf(ptr, "%s %c", nome, &presenca) == 2)
        printf("%s %c\n", nome, presenca);

    return 0;
}
```