

# Estrutura de Dados

## IFTM - Campus Ituiutaba

Ciência da Computação - 2º Sem. 2025

Prof. Alencar Melo Jr., Dr. Eng.

### Atividade 1 - Assunto: Revisão de Linguagem C

## Revisão de Programação em C com Análise de Código

### Objetivo

Revisar conceitos de programação em C por meio da análise do programa fornecido, focando na compreensão de trechos específicos do código, propostas de melhorias e modificações. Essa atividade revisa aspectos importantes da linguagem C, como uniões, estruturas, vetores, controle de fluxo, além de propor soluções práticas para melhorar o código e a funcionalidade do programa.

### Instruções

#### 1. Análise de Trechos de Código

- Leia atentamente o código fornecido: `RevisaoC.c`;
- Compile e execute o programa, observando atentamente as entradas e saídas;
- Responda às seguintes questões sobre trechos específicos do código.

#### 2. Interpretação e Explicação

- De modo geral, o que o programa faz?
- Explique o papel da seguinte estrutura dentro do programa:

```
union {  
    float notaFinal;  
    char conceito;  
} desempenho;
```

- Por que foi utilizado o recurso de união (`union`) para armazenar o desempenho do aluno? Qual é a diferença em relação a uma estrutura convencional?
- O que a variável `usaNota` representa no programa? Como ela influencia o fluxo de dados e a exibição das informações?
- Quais modificações devem ser feitas no código para que o programa aceite um número maior de alunos?
- Quais modificações devem ser feitas no código para que os protótipos de funções possam ser removidos?
- 

#### 3. Função `calcularMediaNotas()`

- Analise o código da função `calcularMediaNotas()`:
  - Qual o critério utilizado para calcular a média das notas dos alunos?

- Como a função trata os alunos que possuem conceito ao invés de nota numérica? Proponha uma modificação para que o conceito também seja considerado de forma ponderada na média (ex.: "A" = 9.0, "B" = 7.0, etc.).
- 4. **Proposta de Alteração: Funcionalidade Extra**
  - Proponha uma funcionalidade adicional para o programa. Algumas sugestões incluem:
    - Permitir a remoção de um aluno da lista.
    - Incluir um campo para a matrícula do aluno, garantindo que o número de matrícula seja único.
  - Descreva como essa nova funcionalidade seria implementada e faça as modificações necessárias no código.
- 5. **Exploração de Alternativas: União vs. Estrutura**
  - O programa usa uma `union` para armazenar ou a nota final, ou o conceito. Explique as vantagens e desvantagens de usar `union` nesse contexto. Em que situação uma estrutura seria mais adequada?
- 6. **Passagem de parâmetros**
  - Explique como acontece a passagem dos dois parâmetros em:  
`void inserirAlunos(struct Aluno alunos[], int quantidade);`
  - Caso a função `inserirAlunos()` venha a alterar o parâmetro `quantidade`, o argumento `totalAlunos` na função `main()` também será alterado? Explique.
- 7. **Desafio Opcional: Média com Conceitos**
  - Modifique o programa para calcular a média considerando tanto notas numéricas quanto conceitos, transformando conceitos em valores numéricos equivalentes. Por exemplo:
    - A = 9.0
    - B = 7.0
    - C = 5.0
    - Proponha uma lógica que leve em conta esses valores e ajuste a função `calcularMediaNotas()`.

### Dica:

- Teste cada alteração proposta para garantir que o programa continue funcionando corretamente.