# Fundação Hermínio Ometto Bacharelado em Sistemas de Informação

SIF009 - Linguagem de Programação I Prof. Dr. Sérgio Luis Antonello

# Plano de ensino

- Unidade I Programação estruturada e Linguagem C (objetivos b, c e d).
  - 1.1. Conceitos de programação estruturada.
  - 1.2. Estrutura de um programa de computador.
  - 1.3. Códigos fonte, objeto e executável.
  - 1.4. Biblioteca de códigos.
  - 1.5. Compiladores e Interpretadores.
  - 1.6. Processos de compilação e link edição.
  - 1.7. Identificação dos tipos de erros e alertas (léxicos, sintáticos e semânticos).
  - 1.8. Depuração de código.
  - 1.9. Palavras reservadas.
  - 1.10. Tipos de dados.
  - 1.11. Constantes. Variáveis simples e estruturadas. Escopo de variáveis.
  - 1.12. Operadores e precedência.
  - 1.13. Expressões aritméticas, lógicas e relacionais.
  - 1.14. Comandos.
  - 1.15. Ambientes de desenvolvimento e programação.
- Unidade II Estruturas de controle (sequência, decisão e repetição), registro e arquivo
  - 2.1. Comandos if e switch.
  - 2.2. Comandos for, while e do while.
  - 2.3. Blocos de comandos e aninhamento.
  - 2.4. Definição de tipos.
  - 2.5. Registro.
  - 2.6. Arquivo: leitura e gravação de dados em disco.
- Unidade III Ponteiros e Funções (objetivos a, c, d, e).
  - 3.1. Ponteiros.
  - 3.2. Funções.
  - 3.3. Passagem de parâmetro por valor.
  - 3.4. Passagem de parâmetro por referência.
- Unidade IV- Strings e Variáveis indexadas (objetivos a, c, d).
  - 4.1. Manipulação de strings.
  - 4.2. Manipulação de caracteres.
  - Declaração e manipulação de vetores.
  - 4.4. Declaração e manipulação de matrizes.

# Plano de ensino

| Data  | Atividade |
|-------|-----------|
| 04/08 | Aula 01   |
| 11/08 | Aula 02   |
| 18/08 | Aula 03   |
| 25/08 | Aula 04   |
| 01/09 | Aula 05   |
| 08/09 | Aula 06   |
| 15/09 | Prova 1   |
| 22/09 | Aula 08   |
| 29/09 | Aula 09   |
| 06/10 | Aula 10   |

| Data             | Atividade                      |  |  |  |  |
|------------------|--------------------------------|--|--|--|--|
| 13/10            | Aula 11                        |  |  |  |  |
| 20/10            | Maratona FHO de<br>Programação |  |  |  |  |
| 27/10            | Aula 13                        |  |  |  |  |
| 03/11            | Aula 14                        |  |  |  |  |
| 10/11            | Aula 15                        |  |  |  |  |
| 17/11            | Prova 2<br>Entrega Trabalho    |  |  |  |  |
| 24/11            | Semana Científica              |  |  |  |  |
| <b>01/12</b>     | Prova SUB                      |  |  |  |  |
| 08/12            | Aula 19                        |  |  |  |  |
| <del>15/12</del> | Aula 20                        |  |  |  |  |

## Sumário da aula

#### Primeiro momento (revisão)

- ✓ Conceitos de variáveis indexadas
  - ✓ Vetor
  - ✓ Matriz

#### Segundo momento (conteúdo)

✓ Aplicação prática dos conteúdos trabalhados

#### Terceiro momento (síntese)

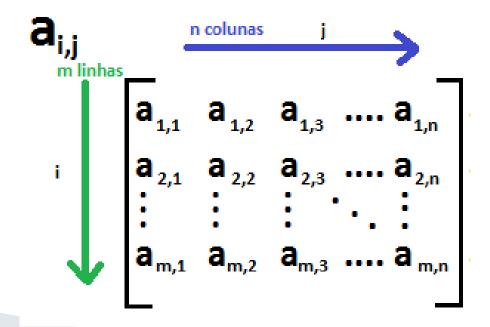
✓ Retome pontos importantes da aula

Uma matriz bidimensional precisa de 2 índices;

Uma matriz tridimensional precisa de 3 índices; etc.

Um elemento da matriz pode ser usado no programa como qualquer outra variável.





Para cada dimensão é necessário um índice;

O primeiro elemento da matriz tem **índice zero na linha** e **índice zero na coluna**.

int matriz [2] 
$$[5] = \{ \{1, 2, 3, 4, 5\}, \{6, 7, 8, 9, 10\} \}$$

| MATRIZ    | coluna [0] | coluna [1] | coluna [2] | coluna [3] | coluna [4] |
|-----------|------------|------------|------------|------------|------------|
| linha [0] | 1          | 2          | 3          | 4          | 5          |
| linha [1] | 6          | 7          | 8          | 9          | 10         |

Rotina clássica para acessar sequencialmente todos os elementos da matriz.

#### Exemplo:

```
int main() {
   int AluNotas[5][3];
   int indalu, indnot;

   for(indalu=0; indalu<5; indalu++) {
      for (indnot=0; indnot<3; indnot++) {
        scanf("%d", &AluNotas[indalu][indnot]);
      }
   }
}</pre>
```

Correção de exercícios



# 2. Segundo momento

Aplicação prática dos conteúdos trabalhados

# 3. Exercícios

# Vamos Programar!



## 4. Exercícios

1) Desenvolver conjuntamente os exercícios propostos.

- O typedef permite definir um tipo de dados a ser usado no programa.
- O struct possibilita especificar um dado estruturado (composto por mais de uma variável).
- Ler e escrever dados em um arquivo texto na linguagem C é feito por meio de stream. Os comandos básicos para isso são fopen(), fclose(), fscanf() e fprintf().

As strings são formadas por conjunto de caracteres.

Existem funções específicas para manuseio de strings. Essas funções são encontradas na biblioteca string.h.

Existem funções específicas para manuseio de caracteres. Essas funções são encontradas na biblioteca ctype.h.

As funções são rotinas que resolvem problemas específicos dentro do programa.

A comunicação de dados entre o código principal e uma função ou entre funções ocorre por meio da passagem de parâmetros.

Na passagem de parâmetros por valor, uma cópia do conteúdo do argumento é encaminhado para o parâmetro da função. Se o conteúdo do parâmetro for alterado, o valor do argumento fica preservado.

Na passagem de parâmetros por referência, o endereço de memória do argumento é encaminhado para o parâmetro da função. Se o conteúdo do parâmetro for alterado, o valor do argumento também é alterado.

As variáveis indexadas usam índices para acessar cada um de seus elementos.

Um vetor usa um índice.

Uma matriz bidimensional usa dois índices, um para especificar a linha e outro para especificar a coluna da matriz.

Rotinas com loops aninhados possibilitam acessar individualmente todos os elementos de uma matriz.