

Prof. esp. Thalles Canela

- **Graduado:** Sistemas de Informação - Wyden Facimp
- **Pós-graduado:** Segurança em redes de computadores - Wyden Facimp
- **Professor:** Todo núcleo de T.I. (Graduação e Pós) - Wyden Facimp
- **Diretor:** SCS
- **Gerente de Projetos:** Motoca Systems

Redes sociais:

- **Linkedin:** <https://www.linkedin.com/in/thalles-canela/>
- **YouTube:** <https://www.youtube.com/aXR6CyberSecurity>
- **Facebook:** <https://www.facebook.com/axr6PenTest>
- **Instagram:** https://www.instagram.com/thalles_canela
- **Github:** <https://github.com/ThallesCanela>
- **Github:** <https://github.com/aXR6>
- **Twitter:** <https://twitter.com/Axr6S>

Introdução à Modularização

A thick yellow horizontal bar spans the width of the slide, with a vertical yellow bar extending downwards from its right end.

- Conceito de Modularização.
- Importância da modularização na programação.

O que é Modularização?

- Modularização é o processo de dividir um programa em módulos ou funções menores.
- Cada módulo realiza uma tarefa específica.
- Facilita a manutenção, leitura e reutilização do código.

Vantagens da Modularização

- Organização: Mantém o código mais estruturado.
- Reutilização: Permite usar o mesmo módulo em diferentes partes do programa ou em diferentes programas.
- Facilidade na Manutenção: Problemas podem ser isolados a um módulo específico.

Conceitos Gerais de Procedimentos e Funções


A thick yellow horizontal bar spans the width of the slide, with a vertical yellow bar extending downwards from its right end.

- Introdução ao conceito de Procedimentos e Funções.
- Como eles são a base da modularização em C.

Procedimentos em Ação

- Definição de Procedimentos.
- Procedimentos são sobre ações e não retornam valor.
- Exemplo:
 - Vamos considerar um procedimento que inicializa um array de inteiros com zeros.

c

 Copy code

```
void inicializarArray(int arr[], int tamanho) {  
    for(int i = 0; i < tamanho; i++) {  
        arr[i] = 0;  
    }  
}
```


Procedimentos em Ação

- void: Indica que não há retorno.
- inicializarArray: Nome do procedimento.
- Parâmetros: arr[] (o array que queremos inicializar), tamanho (número de elementos no array).

Funções em Ação

- Definição de Funções.
- Funções são usadas para cálculos ou operações e retornam valor.
- Exemplo:
 - Considerando o array anterior, vamos criar uma função para calcular a soma de seus elementos.

c

 Copy code

```
int somarElementos(int arr[], int tamanho) {  
    int soma = 0;  
    for(int i = 0; i < tamanho; i++) {  
        soma += arr[i];  
    }  
    return soma;  
}
```


Funções em Ação

- int: Tipo de retorno - nossa função retorna a soma dos elementos.
- somarElementos: Nome da função.
- Lógica: Percorre cada elemento do array, adiciona ao total e retorna a soma.

Comparando Procedimentos e Funções

| | Procedimentos | Funções |
|---------------------|---------------------------------|-------------------------------|
| Retorno | Não retorna valor | Retorna valor |
| Finalidade | Realizar ação | Calcular/Obter valor |
| Exemplo dado | <code>`inicializarArray`</code> | <code>`somarElementos`</code> |

- Ambos ajudam na modularização do código.
- Diferenças chave: retorno e propósito.

Aplicação Prática dos Conceitos

```
c Copy code  
  
int main() {  
    int numeros[5];  
    inicializarArray(numeros, 5);  
    int total = somarElementos(numeros, 5);  
    printf("Soma: %d", total);  
    return 0;  
}
```

- Vamos imaginar um cenário em que desejamos inicializar um array e, em seguida, calcular a soma de seus elementos.

Aplicação Prática dos Conceitos

- Criamos um array chamado `numeros` com 5 elementos.
- Usamos `inicializarArray` para definir todos os elementos como 0.
- Usamos `somarElementos` para calcular a soma (que será 0).
- Exibimos a soma.

Referência Bibliográfica:

- Knuth, D. E. (1968). The Art of Computer Programming: Volume 1: Fundamental Algorithms. Addison-Wesley.
- Este livro é uma obra clássica na área de ciência da computação, escrito por Donald E. Knuth. Ele cobre muitos aspectos fundamentais da programação e é uma referência valiosa para qualquer pessoa interessada em algoritmos e estruturas de dados.

Revista Científica:

- Journal of Computer and System Sciences.
- Uma revista renomada que aborda pesquisas de ponta em ciência da computação e sistemas. Ela cobre uma ampla variedade de tópicos, incluindo algoritmos, teoria da computação, e mais.

Artigos Científicos:

- Attiya, H., & Welch, J. (2004). Sequential consistency versus linearizability. ACM Transactions on Computer Systems (TOCS), 12(2), 91-122.
- Este é um exemplo de artigo que discute a consistência em sistemas distribuídos, um tópico fundamental em ciência da computação moderna.

Periódicos:

- ACM Computing Surveys.
- Publicado pela Association for Computing Machinery, este periódico oferece pesquisas extensivas e revisões de literatura sobre diversos tópicos em ciência da computação.

Sites Científicos:

- Google Scholar (<https://scholar.google.com/>)
- É uma ferramenta de busca especializada que ajuda na localização de literatura acadêmica, incluindo artigos, teses, livros e relatórios de conferências.
- arXiv (<https://arxiv.org/>)
- Uma plataforma gratuita que hospeda pré-publicações de artigos científicos em diversas áreas, incluindo ciência da computação.
- DBLP (<https://dblp.uni-trier.de/>)
- Uma base de dados online de referências bibliográficas em ciência da computação. É uma excelente ferramenta para rastrear publicações em informática e áreas relacionadas.