

# Algoritmo s e Técnicas de Programa

Prof. Me. Anderson E. Macedo

Conçalves



# Aula 1: Procedimen tos e Funções



UNIDADE 4 – Funções e recursividade

# Objetivo da aula

1.

## **Procedimentos / Funções**

Aborda como criar procedimentos e funções na linguagem C.

2.

## **Tipos de Funções**

Funções com e sem retorno.

3.

## **Realidade profissional**

Aplicar exemplos da realidade profissional.

4.

## **Palavras-chave**

Procedimentos, Funções e ponteiros.

# Procedimentos

São blocos de códigos separados do programa principal, que realizam tarefas específicas, porém não retornam valor.

```
1  #include <stdio.h>
2
3
4  void saudacao() {
5      printf("Olá, mundo!\n");
6  }
7
8  int main() {
9      saudacao(); // Chamando o procedimento
10     return 0;
11 }
12
13
14
```

# Funções

Similares aos procedimentos, porém podem retornar um valor, além de ter um tipo definido para retornar valores.

```
1  #include <stdio.h>
2
3  ✓ int soma(int a, int b) {
4      return a + b;
5  }
6
7  ✓ int main() {
8      int resultado = soma(3, 5); // Chamando a função e armazenando o resultado
9      printf("A soma é: %d\n", resultado);
10     return 0;
11 }
12
```

# Funções com ponteiros

Variáveis que armazenam endereços de memória, permitindo o acesso direto e a modificação dos dados naquela posição.

```
int numero = 10;  
int *ptr_numero = &numero; // inicializando o ponteiro com o  
                               endereço de 'numero'
```

# Vetor com ponteiros

Funções podem retornar ponteiros, permitindo a alocação dinâmica de memória ou o retorno de endereços de variáveis locais.

```
1  #include <stdio.h>
2
3  ∨ int* aloca_memoria() {
4      int *ponteiro = (int *)malloc(sizeof(int)); // Alocação dinâmica de memória
5      *ponteiro = 10;
6      return ponteiro;
7  }
8
9  ∨ int main() {
10     int *ptr = aloca_memoria(); // Recebe o ponteiro retornado pela função
11     printf("Valor alocado dinamicamente: %d\n", *ptr);
12     free(ptr); // Libera a memória alocada dinamicamente
13     return 0;
14 }
```

# Realidade Profissional





# Revisando

1.

## **Procedimentos / Funções**

Aborda como criar procedimentos e funções na linguagem C.

2.

## **Tipos de Funções**

Funções com e sem retorno.

3.

## **Realidade profissional**

Aplicar exemplos da realidade profissional.

4.

## **Palavras-chave**

Procedimentos, Funções e ponteiros.