

Programação Básica em ANSI C - 2

Prof. Nilton Correia da Silva

October 17, 2024

Estrutura Básica de um Programa

```
#include <stdio.h>

int main() {
    printf("Hello, \uWorld!\n");
    return 0;
}
```

- Um programa simples em C que imprime "Hello, World!".

Operadores

- ▶ Operadores aritméticos: +, -, *, /, %

```
#include <stdio.h>

int main() {
    int a = 10;
    int b = 3;
    printf("Soma: \u00d1%d\u00d1n", a + b);
    printf("Subtracao: \u00d1%d\u00d1n", a - b);
    printf("Multiplicacao: \u00d1%d\u00d1n", a * b);
    printf("Divisao: \u00d1%d\u00d1n", a / b);
    printf("Resto: \u00d1%d\u00d1n", a % b);
    return 0;
}
```

Estruturas de Controle - Condicional if

```
#include <stdio.h>

int main() {
    int numero;
    printf("Digite um numero: ");
    scanf("%d", &numero);

    if (numero > 0) {
        printf("0 numero e positivo.\n");
    } else if (numero < 0) {
        printf("0 numero e negativo.\n");
    } else {
        printf("0 numero e zero.\n");
    }
    return 0;
}
```

Estruturas de Controle - Switch Case

```
#include <stdio.h>
int main() {
    int escolha;
    printf("Menu de Opcoes:\n");
    printf("1. Opcao 1\n");
    printf("2. Opcao 2\n");
    printf("Escolha uma opcao: ");
    scanf("%d", &escolha);
    switch (escolha) {
        case 1:
            printf("Voce escolheu 1.\n"); break;
        case 2:
            printf("Voce escolheu 2.\n"); break;
        default:
            printf("Opcao invalida!\n");
    }
    return 0;
}
```

Estruturas de Controle - Loop for

```
#include <stdio.h>

int main() {
    int i;
    for (i = 0; i < 5; i++) {
        printf("i_=%d\n", i);
    }
    return 0;
}
```

- Um laço de repetição simples que imprime valores de 0 a 4.

Estruturas de Controle - Loop while

```
#include <stdio.h>

int main() {
    int i = 0;
    while (i < 10) {
        printf("i_=%d\n", i);
        i++;
    }
    return 0;
}
```

- Um laço de repetição simples que imprime valores de 0 a 9.

Estruturas de Controle - Loop do-while

```
#include <stdio.h>

int main() {
    int i = 0;
    do {
        printf("i_ = %d\n", i);
        i++;
    } while (i < 10);
    return 0;
}
```

- ▶ Um laço de repetição do-while que imprime valores de 0 a 9.
- ▶ A condição é verificada após a execução do bloco de código.

Procedimentos em C

```
#include <stdio.h>

void saudacao() {
    printf("Hello, \uWorld!\n");
}

int main() {
    saudacao(); // Chamada do procedimento que
               imprime uma mensagem
    return 0;
}
```

- ▶ Procedimento que imprime "Hello, World!" na tela.
- ▶ Procedimentos em C são definidos com o tipo void e não retornam valor.

Funções sem Parâmetro de Entrada

```
#include <stdio.h>
int getNumeroMisterioso() {
    return 42; // Retorna um numero "misterioso"
}
int main() {
    int numero = getNumeroMisterioso();
    printf("O número misterioso : %d\n",
        numero);
    return 0;
}
```

- ▶ Função `getNumeroMisterioso` que não aceita nenhum parâmetro e retorna o valor 42.
- ▶ Útil para fornecer valores constantes ou realizar operações internas sem interação externa.

Funções

```
#include <stdio.h>

int soma(int a, int b) {
    return a + b;
}

int main() {
    int resultado = soma(5, 3);
    printf("Resultado: \u00d1%d\u00d1n", resultado);
    return 0;
}
```

- Função que realiza a soma de dois números.