

Estruturas de
Controle

Lógica de Programação

Introdução às Estruturas de Controle

As estruturas de controle permitem controlar o fluxo de execução de um programa.

Neste material, veremos:

1. Estruturas de decisão (if, switch).
2. Estruturas de repetição (for, while, do-while).
3. Exemplos práticos e aplicações.
4. Exercícios para fixação.

O que são Estruturas de Controle?

São comandos que alteram o fluxo padrão de execução (sequencial) de um programa.

Tipos:

- Decisão: Controlam o fluxo com base em condições.
- Repetição: Executam blocos de código múltiplas vezes.

Estruturas de Decisão: If

Permite executar um bloco de código com base em uma condição.

Sintaxe:

```
if (condição) {  
    // bloco de código  
}  
else {  
    // bloco alternativo  
}
```

Exemplo de If – Linguagem C

- Verificar se um número é positivo:

```
if (numero > 0) {  
    printf("Número positivo");  
} else {  
    printf("Número negativo ou zero");  
}
```

Estruturas de Decisão: Switch

Executa diferentes blocos de código com base no valor de uma variável.

Sintaxe:

```
switch (variável) {  
    case valor1:  
        // bloco de código  
        break;  
    case valor2:  
        // outro bloco  
        break;  
    default:  
        // bloco padrão  
}
```

Exemplo de Switch

Verificar o dia da semana:

```
switch (dia) {  
  case 1:  
    print('Domingo');  
    break;  
  case 2:  
    print('Segunda-feira');  
    break;  
  default:  
    print('Outro dia');  
}
```

Estruturas de Repetição: For

Repete um bloco de código por um número fixo de vezes.

Sintaxe:

```
for (inicialização; condição; incremento) {  
    // bloco de código  
}
```


Exemplo de For

Imprimir números de 1 a 5:

```
for (int i = 1; i <= 5; i++) {  
    printf(i);  
}
```

Estruturas de Repetição: While

Executa um bloco de código enquanto a condição for verdadeira.

Sintaxe:

```
while (condição) {  
    // bloco de código  
}
```

Exemplo de While

Imprimir números de 1 a 5:

```
int i = 1;  
while (i <= 5) {  
    printf(i);  
    i++;  
}
```

Estruturas de Repetição: Do-While

Executa o bloco de código pelo menos uma vez, depois verifica a condição.

Sintaxe:

```
do {
```

```
    // bloco de código
```

```
} while (condição);
```

Exemplo de Do-While

Imprimir números de 1 a 5:

```
int i = 1;  
do {  
    printf(i);  
    i++;  
} while (i <= 5);
```

While vs. Do-While

- While: verifica a condição antes de executar o bloco.
- Do-While: executa o bloco pelo menos uma vez antes de verificar a condição.

Controle de Laços: Break

Usado para sair de um laço ou switch imediatamente.

Exemplo:

```
for (int i = 1; i <= 10; i++) {  
    if (i == 5) {  
        break;  
    }  
    printf(i);  
}
```

Controle de Laços: Continue

Interrompe a iteração atual e avança para a próxima.

Exemplo:

```
for (int i = 1; i <= 10; i++) {  
    if (i == 5) {  
        continue;  
    }  
    print(i);  
}
```


Exercícios de Fixação (Parte 1)

1. Crie um programa que determine se um número é par ou ímpar usando if.
2. Use um laço for para calcular a soma dos números de 1 a 100.

Exercícios de Fixação (Parte 2)

3. Crie um programa que imprima os números de 1 a 10, exceto o 5, usando continue.
4. Use um switch para exibir os meses correspondentes a um número de 1 a 12.

Revisão do Capítulo

No material, aprendemos sobre:

- Estruturas de decisão: if, switch.
- Estruturas de repetição: for, while, do-while.
- Controle de fluxo com break e continue.

Aplicações Práticas

- Implementação de menus interativos.
- Processamento de grandes volumes de dados com laços.
- Resolução de problemas condicionais complexos.

Referências

- 1. Apostila de Lógica, Maromo
- 2. Documentação das linguagens de programação C, Java.