

Linguagem de Programação I - ESP201

Prof^a Ana Carolina Sokolonski

Bacharelado em Sistemas de Informação
Instituto Federal de Ciência e Tecnologia da Bahia
Campus de Feira de Santana

carolsoko@ifba.edu.br

June 3, 2025

Estruturas Condicionais

1 Estruturas Condicionais

- Estrutura Condicional **IF**
- Estrutura Condicional **IF-ELSE**
- Estrutura Condicional **Operador Ternário - (?:)**
- Estrutura Condicional **SWITCH-CASE**
 - Estrutura Condicional **SWITCH-CASE-BREAK**
 - Estrutura Condicional **SWITCH-CASE-SEM BREAK**

2 Referências

Estruturas Condicionais

Estruturas Condicionais

As Estruturas Condicionais permitem realizar testes nas variáveis e alterar o fluxo do programa de acordo com as informações testadas.

Na linguagem C, existem três tipos de Estruturas Condicionais:

- 1 if
- 2 if - else
- 3 operador ternário - (?:)
- 4 switch-case

Essas estruturas são fundamentais para controlar o fluxo de execução do programa de acordo com condições específicas.

Estrutura Condicional IF

Uma estrutura de decisão examina uma ou mais condições e decide quais instruções serão executadas dependendo se a condição foi atendida ou não [Schildt e Mayer 1997]. A Estrutura Condicional IF é uma estrutura de decisão muito utilizada em todas as linguagens de programação. **O IF só executa a instrução se a condição de teste for verdadeira, nada fazendo se a expressão for falsa.**

```
if(condicao){  
    instrução ou instruções para condição verdadeira;  
}
```

Estrutura Condicional IF - exemplo 1

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int main(void){
4     int A, B, Soma;
5
6     printf("Digite um numero inteiro: ");
7     scanf("%d", &A);
8
9     printf("Digite um numero inteiro: ");
10    scanf("%d", &B);
11
12    Soma = A + B;
13    printf ("O Valor da soma = %d\n", Soma);
14
15    if(Soma > 10){
16        printf("O valor da soma eh maior que 10\n");
17    }
18
19    return(0);
20 }
```

Estrutura Condicional IF - exemplo 2

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(void){
4      int x, y;
5
6      printf("Digite um numero inteiro: ");
7      scanf("%d", &x);
8
9      printf("Digite outro numero inteiro: ");
10     scanf("%d", &y);
11
12     if(x > y) printf("O primeiro número é maior!\n");
13
14     if(y > x) printf("O segundo número é maior!\n");
15
16     if(x==y) printf("Os números são iguais!\n");
17
18     return(0);
19 }
```

Estrutura Condicional IF-ELSE

A Estrutura Condicional **IF-ELSE** é uma estrutura de decisão que permite que o programa assuma dois caminhos diferentes dependendo do estado de suas variáveis. O comando **if** só executa a instrução caso a condição de teste seja verdadeira. O comando **else** executará a instrução caso a condição seja falsa.

```
if(condicao){  
    instrução ou instruções para condição verdadeira;  
}else{  
    instrução ou instruções para condição falsa;  
}
```


Estrutura Condicional IF-ELSE - exemplo 1

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(void){
4      int x, y;
5
6      printf("Digite um numero inteiro: ");
7      scanf("%d", &x);
8
9      printf("Digite outro numero inteiro: ");
10     scanf("%d", &y);
11
12     if(x > y){ printf("O primeiro número é maior!\n")}
13     else{printf("O segundo número é maior!\n");}
14
15     return(0);
16 }
```

Estrutura Condicional IF-ELSE - exemplo 2

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(void){
4      int x, y;
5
6      printf("Digite um numero inteiro: ");
7      scanf("%d", &x);
8
9      printf("Digite outro numero inteiro: ");
10     scanf("%d", &y);
11
12     if(x > y){printf("O primeiro número é maior!\n");}
13     else if(y>x){printf("O segundo número é maior!\n");}
14     else{printf("Os números são iguais!\n");}
15
16     return(0);
17 }
```

Estrutura Condicional Operador Ternário - (?:):

O operador ternário **?:** é uma forma compacta de expressar uma instrução **IF-ELSE**:

(condicao)? expressao1 : expressao2

- *max = (num1 > num2)? num1 : num2*
- *if(num1 > num2) max = num1;*
else max = num2;
- *modulo = (num < 0)? - num : num;*

Estrutura Condicional - Operador Ternário - (?:) - exemplo 1

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  void main() {
4      int a,b,c,d;
5      printf("Digite o 1º Número: ");
6      scanf("%d",&a);
7      printf("\nDigite o 2º Número: ");
8      scanf("%d",&b);
9      printf("\nDigite o 3º Número: ");
10     scanf("%d",&c);
11
12     d = (a > b)? a : b;
13
14     if (c > d) printf("\n0 Maior é %d",c);
15     else printf("\n0 Maior é %d",d);
16 }
```

Estrutura Condicional - Operador Ternário - (?:) - exemplo 2

```
1  #include <stdio.h>
2  void main() {
3      int a,b,c,d;
4      printf("Digite três números: ");
5      scanf("%d\n%d\n%d",&a,&b,&c);
6
7      d = (a > b)? a : b;
8
9      if (c > d) printf("\n0 Maior é %d",c);
10     else printf("\n0 Maior é %d",d);
11 }
```

Estrutura Condicional SWITCH-CASE

Forma de se substituir o comando **IF-ELSE** ao se executar vários testes ao mesmo tempo.

Em diversos programas precisamos determinar se um valor encontra-se numa lista de valores. Apesar de podermos usar uma sequência de **IF's**, este recurso além de não ser elegante, por vezes confunde o entendimento do programa.

Vejamos como o **Switch-case** funciona.

Vamos analisar o seguinte exemplo:

- 1 Faça um algoritmo que leia um número inteiro que indica um dia da semana e informe o nome desse dia, de acordo com a tabela abaixo:

Número	Nome
1	Domingo
2	Segunda
3	Terça
4	Quarta
5	Quinta
6	Sexta
7	Sábado

Resposta usando o IF-ELSE:

```
1  #include <stdio.h>
2  void main (){
3      int dia;
4      printf ("Digite o dia da semana de 1 a 7: ");
5      scanf ("%d", &dia);
6
7      if (dia == 1) printf ("Domingo\n");
8      else if (dia == 2) printf ("Segunda\n");
9      else if (dia == 3) printf ("Terça\n");
10     else if (dia == 4) printf ("Quarta\n");
11     else if (dia == 5) printf ("Quinta\n");
12     else if (dia == 6) printf ("Sexta\n");
13     else if (dia == 7) printf ("Sabado\n");
14     else printf ("Valor invalido!\n");
15 }
```


Resposta usando o SWITCH-CASE:

```
1  #include <stdio.h>
2  void main (){
3      int dia;
4      printf ("Digite o dia da semana de 1 a 7: ");
5      scanf ("%d", &dia);
6
7      switch (dia){
8          case 1: printf ("Domingo\n"); break;
9          case 2: printf ("Segunda\n"); break;
10         case 3: printf ("Terça\n"); break;
11         case 4: printf ("Quarta\n"); break;
12         case 5: printf ("Quinta\n"); break;
13         case 6: printf ("Sexta\n"); break;
14         case 7: printf ("Sabado\n"); break;
15         default: printf ("Valor invalido!\n");
16     }
17 }
```

Estrutura Condicional SWITCH-CASE

- **"variável"** deve ser uma variável do tipo inteiro ou caracter;
- **"break"** serve para terminar a sequência de comandos em execução, por serem opcionais, se forem suprimidos permitem que o **"case"** a seguir seja executado, sem haver qualquer quebra na sequência do processamento.

Estrutura Condicional SWITCH-CASE-BREAK

```
switch < variavel > {  
    case < constante1 >: < comandos >;  
    [break;]  
    case < constante2 >: < comandos >;  
    [break;]  
    case < constante3 >: < comandos >;  
    [break;]  
    [default : < comandos >;]  
}
```

Vejamos mais um exemplo:

- 1 Faça um algoritmo que implemente uma calculadora matemática com as 4 operações básicas (+, -, *, /):

O algoritmo deverá ler: (número real) (operador) (número real) e deverá exibir o resultado.

ex: o algoritmo vai ler: $5 + 7$

Deverá mostrar como resultado: $5 + 7 = 12$

Resposta usando o IF-ELSE:

```
1  #include <stdio.h>
2  void main(){
3      float n1, n2;
4      char op;
5
6      printf("Digite a expressão matemática: ");
7      scanf("%f\n%c\n%f", &n1, &op, &n2);
8
9      if (op == '+') printf("%.2f + %.2f = %.2f", n1, n2, n1+n2);
10     else if (op == '-') printf("%.2f - %.2f = %.2f", n1, n2, n1-n2);
11     else if (op == '*') printf("%.2f * %.2f = %.2f", n1, n2, n1*n2);
12     else if (op == '/')
13     {
14         if (n2 != 0) printf("%.2f / %.2f = %.2f", n1, n2, n1/n2);
15         else printf("Expressão inválida! Proibido divisão por zero!");
16     }
17     else printf("Operador inválido!");
18 }
```

Resposta usando o SWITCH-CASE:

```
1  #include <stdio.h>
2  void main(){
3      float n1, n2;
4      char op;
5
6      printf("Digite a expressão matemática: ");
7      scanf("%f\n%c\n%f", &n1, &op, &n2);
8
9      switch (op){
10         case '+': printf("%5.2f + %5.2f = %5.2f", n1, n2, n1+n2);
11                     break;
12         case '-': printf("%5.2f - %5.2f = %5.2f", n1, n2, n1-n2);
13                     break;
14         case '*': printf("%5.2f * %5.2f = %5.2f", n1, n2, n1*n2);
15                     break;
16         case '/': if (n2!=0) printf("%5.2f + %5.2f = %5.2f", n1, n2, n1+n2);
17                     else printf("Expressão inválida! Proibido divisão por zero!");
18                     break;
19         default: printf("Operador inválido!");
20     }
21 }
```

Estrutura Condicional SWITCH-CASE-SEM BREAK:

Dependendo da situação, pode ser necessário ocultar o BREAK.

```
switch < variavel > {  
    case < constante1 >: < comandos >;  
        [breakoucontinue;]  
    case < constante2 >: < comandos >;  
        [breakoucontinue;]  
    case < constante3 >: < comandos >;  
        [breakoucontinue;]  
    [default : < comandos >;]  
}
```

Referências

Referências



SCHILDT, H.; MAYER, R. *C completo e total*. [S.l.]: Pearson University, 1997. ISBN 9788534605953.