

Estruturas de Seleção em C

Prof. André Bezerra

Comando **IF**

- Na linguagem C, o comando **if** é utilizado sempre que é necessário escolher entre dois caminhos dentro do programa, ou quando se deseja executar um ou mais comandos que estejam sujeitos ao resultado de um teste.

Forma geral do comando **IF**

```
if (condição){}
```

```
if(x > 10){  
    /* ..do something */  
}
```

```
if(!(i < 3) && (j == 4){  
    /* ..here goes the code */  
}
```

Exemplo 01

```
#include <stdio.h>
```

```
int main(){  
    float media;  
    printf("Media do Aluno:");  
    scanf("%f", &media);  
  
    if (media >= 7.0)  
        printf("Aluno aprovado");  
  
    return 0;  
}
```

Exemplo 02

```
#include <stdio.h>
int estaAprovado( float media);
int main(){
    float media;
    printf("Media do Aluno:");
    scanf("%f", &media);
    if (estaAprovado(media)){
        printf("Aluno aprovado");
    }
    return 0;
}
int estaAprovado( float media){
    if (media >= 7.0){
        return 1;
    }
    if (media <= 7.0){
        return 0;
    }
}
```

Exemplo 03

```
#include <stdio.h>

int estaAprovado( float media);

int main(){
    float media;
    printf("Media do Aluno:");
    scanf("%f", &media);
    if (estaAprovado(media)){
        printf("Aluno aprovado");
    }
    return 0;
}

int estaAprovado( float media){
    return media >= 7.0;
}
```

Exercício 01

- Escreva uma função que receba dois valores inteiros por parâmetro e retorna verdadeiro **se** o primeiro valor é **maior** que o segundo, caso contrário, retorne falso.

Resposta Exercício 01

```
#include <stdio.h>

int primeiroEMaior( int x, int y);

int main(){
    int a, b;
    printf( "Valor para A:");
    scanf( "%d", &a);
    printf( "Valor para B:");
    scanf( "%d", &b);
    if( primeiroEMaior( a,b)){
        printf( "Primeiro Numero eh maior");
    }
    return 0;
}

int primeiroEMaior( int x, int y){
    return x > y;
}
```


Comando ELSE

O comando **else** pode ser entendido como sendo um complemento do comando **if**. Ele auxilia o comando **if** na tarefa de escolher dentre os vários caminhos a ser seguido dentro do programa.

O comando **else** é opcional e sua sequencia de comandos somente será executada se o valor da condição que está sendo testada pelo comando **if** for FALSA.

Forma geral do comando **ELSE**

```
if (condição){  
/* ..if true, do this */  
} else {  
/* ..or do this one */  
}
```

Exemplo 04

```
#include <stdio.h>

int main(){
    float media;
    printf("Media do Aluno:");
    scanf("%f", &media);

    if (media >= 7.0)
        printf("Aluno aprovado");
    else
        printf("Aluno reprovado");

    return 0;
}
```

E quem tem nota menor que 7,
mas maior ou igual a 4?
Deve ir para **NAF**

Exemplo 05

```
#include <stdio.h>

int main(){
    float media;
    printf("Media do Aluno:");
    scanf("%f", &media);
    if (media >= 7.0)
        printf("Aluno aprovado");
    else if ((media >= 4.0) && (media < 7.0))
        printf("NAF");
    else
        printf("Aluno reprovado");
    return 0;
}
```

```
#include <stdio.h>
```

```
int main(){  
    float media;  
    printf("Media do Aluno:");  
    scanf("%f", &media);  
  
    if (media >= 7.0)  
        printf("Aluno aprovado");  
    else if ((media >= 4.0) && (media < 7.0))  
        printf("NAF");  
    else  
        printf("Aluno reprovado");  
  
    return 0;  
}
```

Comando SWITCH

Além dos comandos **if** e **else**, a linguagem C possui um comando de seleção múltipla chamado **switch**. Esse comando é muito parecido com o aninhamento de comandos **if-else-if**.

O comando **switch** é muito mais limitado que o comando **if-else**: enquanto o comando **if** pode testar expressões lógicas ou relacionais, o comando **switch** somente verifica se uma variável (do tipo **int** ou **char**) é ou não igual a um certo valor constante.

Forma geral do comando **SWITCH**

```
switch(variável){  
    case valor1:  
        /*Comandos*/  
        break;  
    case valor2:  
        /*Comandos*/  
        break;  
    default:  
        /*Comandos*/  
        break;  
}
```

Exemplo 06

```
#include <stdio.h>
```

```
int main(){  
    char sexo;  
    printf("Sexo da pessoa:");  
    scanf("%c", &sexo);  
  
    switch(sexo){  
        case 'f':  
            printf("Sexo Feminino");  
            break;  
        case 'm':  
            printf("Sexo Masculino");  
            break;  
        default:  
            printf("Sexo invalido");  
            break;  
    }  
    return 0;  
}
```


Exemplo 07

```
#include <stdio.h>
```

```
int main(){  
    int num;  
    printf("Digite um numero:");  
    scanf("%d", &num);  
  
    switch(num){  
        case 10:{  
            int x;  
            printf("%d, x + num");  
        }  
        break;  
        default:  
            printf("Valor Digitado Diferente de 10");  
            break;  
    }  
    return 0;  
}
```

Exercício 02

Escreva uma função que recebe um número de 0 a 9 e imprima este número por extenso.

Resposta Exercício 02

```
#include <stdio.h>
```

```
void imprimeNumero(int n){  
    switch(n){  
        case 1:  
            printf("Um");  
            break;  
        case 2:  
            printf("Dois");  
            break;  
        default:  
            printf("O numero deve está entre 0 e 9");  
            break;  
    }  
}
```

```
int main() {  
    imprimeNumero(2);  
    return 0;  
}
```

Operador ?

- O operador ? é também conhecido como operador ternário. Trata-se de uma simplificação do comando *if-else* na sua forma mais simples, ou seja, com apenas um comando e não blocos de comandos.

Exemplo 08

```
#include <stdio.h>
```

```
int main(){  
    int x, y;  
    x = 5;  
    y = (x == 5) ? 3 : 2;  
    printf("%d", y);  
    return 0;  
}
```

Exercício 03

- Escreva uma função que recebe dois números inteiros e retorna o maior número.

Resposta Exercício 03

```
#include <stdio.h>
```

```
int maiorNumero(int x, int y){  
    return x > y ? x : y;  
}
```

```
int main() {  
    printf("Maior numero : %d",  
        maiorNumero(4,8));  
    return 0;  
}
```

Exercício 04

- Escreva uma função que receba um número inteiro e retorna verdadeiro caso o número seja **par** e falso se o número for **ímpar**.

Resposta Exercício 04

```
#include <stdio.h>
```

```
int ePar(int x){  
    return !(x % 2) ? 1 : 0;  
}  
int main() {  
    if (ePar(8))  
        printf("Numero Par");  
    else  
        printf("Numero Impar");  
    return 0;  
}
```

Para próxima aula

- Cap. 04 pág 96