

Rodrigo Henrich

rodrigohenrich@faccat.br



Estrutura de seleção IF

- O IF permite assim como o SE dos algoritmos tomar determinadas decisões no programa
- Sua sintaxe é simples

```
if(condicao)
```

```
    instrucao
```

Estrutura de seleção IF

```
1 #include<stdio.h>
2 int main(){
3     int x = 10;
4
5     if(x<=10)
6         printf("x e menor ou igual a 10");
7 }
```

Estrutura de seleção IF

- Caso tenha uma condição de senão

if(condicao)

instrucao

else

instrucao

Estrutura de seleção IF

```
1  #include<stdio.h>
2  int main(){
3      int x = 10;
4
5      if(x<=10)
6          printf("x e menor ou igual a 10");
7      else
8          printf("x e maior do que 10");
9  }
```

Estrutura de seleção IF

- Caso tenha uma condição de senão e outro if

if(condicao)

instrucao

else if(condicao)

instrucao

else

instrucao

Estrutura de seleção IF

```
1 #include<stdio.h>
2 int main(){
3     int x = 10;
4
5     if(x<=10)
6         printf("x e menor ou igual a 10");
7     else if(x<=20)
8         printf("x e menor ou igual a 20");
9     else
10        printf("x e maior do que 20");
11    return 0;
12 }
```

Estrutura de seleção IF

- Uso das {} é necessário sempre que forem executadas mais que uma instrução de código dentro do IF ou else

Estrutura de seleção IF

```
1  #include<stdio.h>
2
3  int main(){
4      int x = 5;
5      int y = 9;
6      if(x < y && y<10){
7          printf("O valor de x e menor do que y");
8          printf("\nO valor de y e menor do que 10");
9      }
10     return 0;
11 }
```

Operador ?

- Seu funcionamento é semelhante ao do IF
- Tem a seguinte sintaxe

`expressao_condicional?comandos_se_verdade:comandos_se_falso`

Operador ?

```
1  #include<stdio.h>
2
3  int main(){
4      int x = 5;
5      if(x == 5)
6          printf("x vale 5");
7      else
8          printf("x e diferente de 5");
9      printf("\nUsando o operador ?");
10     (x == 5)?printf("x vale 5"):printf("x e diferente de 5");
11     return 0;
12 }
```

Comando switch

- Além do IF e do ? existe uma terceira estrutura de seleção
- Ela é o switch
- Ela permite avaliar os valores de uma variável do tipo char ou int
- No entanto essa estrutura funciona de forma mais limitada que o IF por exemplo
- Nela é possível apenas avaliar se determinada variável é igual a determinado valor
- Não sendo possível testar intervalos de valores por exemplo

Comando switch

```
switch(variavel){  
    case valor1:  
        sequência de comandos;  
    break;  
    case valor2:  
        sequência de comandos;  
    break;  
    ...  
    case valorn:  
        sequência de comandos;  
    break;  
    default:  
        sequência de comandos;
```

Comando switch

Exemplo de uso do comando switch

```
1  #include<stdio.h>
2
3  int main(){
4      char simbolo;
5      printf("Digite um sinal de pontuacao grafica ");
6      simbolo = getchar();
7      switch(simbolo){
8          case '.':
9              printf("Digitou um ponto.");
10             break;
11          case ',':
12              printf("Digitou uma virgula.");
13              break;
14          case ':':
15              printf("Digitou dois pontos.");
16              break;
17          case ';':
18              printf("Digitou um ponto e virgula.");
19              break;
20          default:
21              printf("Nao entendi :(");
22      }
23  }
24
```

Comando switch

O mesmo exemplo com IF

```
1  #include<stdio.h>
2
3  int main(){
4      char simbolo;
5      printf("Digite um sinal de pontuacao grafica ");
6      simbolo = getchar();
7      if(simbolo == '.')
8          printf("Digitou um ponto.");
9      else if(simbolo == ',')
10         printf("Digitou uma virgula.");
11     else if(simbolo == ':')
12         printf("Digitou dois pontos.");
13     else if(simbolo == ';')
14         printf("Digitou um ponto e virgula.");
15     else
16         printf("Nao entendi :(");
17 }
```

Resolvendo o problema dos acentos

- Para que o compilador entenda qual a codificação de caracteres do nosso programa, temos que informar para ele qual nossa localidade
- Para resolver a questão de acentuação e caracteres específicos do nosso idioma
- Existe uma biblioteca no C.
- Ela chama locale.h
- O comando para isso é o `setlocale()`;

Resolvendo o problema dos acentos

- A sintaxe do comando é a seguinte

setlocale(int categoria, const char *local)

- Sendo a categoria
 - LC_ALL constante que aponta para todas as opções de idioma

Resolvendo o problema dos acentos

- e o local é um dos possíveis valores
 - "" desta forma a função locale irá obter a localização do sistema
 - "Portuguese" desta forma vamos forçar a localização para português
 - "C" Corresponde a localização padrão

Resolvendo o problema dos acentos

```
1  #include<stdio.h>
2
3  int main(){
4      printf("\n á à â ã ç ");
5      return 0;
6  }
7
```

Resolvendo o problema dos acentos

```
1  #include<stdio.h>
2  #include<locale.h>
3
4  int main(){
5      setlocale(LC_ALL,"Portuguese");
6      printf("\ná à â ã ç");
7      return 0;
8  }
9
```

Resolvendo o problema dos acentos

```
1  #include<stdio.h>
2  #include<locale.h>
3
4  int main(){
5      setlocale(LC_ALL, "");
6      printf("\n á à â ã ç ");
7      return 0;
8  }
```

Resolvendo o problema dos acentos

```
1  #include<stdio.h>
2  #include<locale.h>
3
4  int main(){
5      printf("%s",setlocale(LC_ALL,""));
6      printf("\nná à â ã ç");
7      return 0;
8  }
9
```

Resolvendo o problema dos acentos (alternativa)

```
1  #include<stdio.h>
2  #include <windows.h>
3
4  int main(){
5      // Define codificação do console para UTF-8
6      SetConsoleOutputCP(65001);
7
8      printf("á ç ");
9      return 0;
10 }
```

Exemplo 1

Crie um programa em C que leia um número e diga qual o dia da semana equivale a ele.

Considere 1 para domingo

Informe também se é dia de aula ou se é fim de semana

Realize o exemplo com SWITCH...CASE e com IF...ELSE

Exemplo 1

Crie um programa em C que leia um número e diga qual o dia da semana equivale a ele.

Considere 1 para domingo

Informe também se é dia de aula ou se é fim de semana

Realize o exemplo com SWITCH...CASE e com IF...ELSE