

SSC0101 - ICC1 – Teórica

---

Introdução à Ciência da Computação I

## Comandos de Decisão – Parte II

Prof. Vanderlei Bonato: [vbonato@icmc.usp.br](mailto:vbonato@icmc.usp.br)

Prof. Claudio Fabiano Motta Toledo: [claudio@icmc.usp.br](mailto:claudio@icmc.usp.br)

---

# Sumário

---

- Comando else-if
- Problema do else oscilante
- Estrutura CASE
- Operador condicional ternário
- Exercícios

## Comando <else if>

---

```
if (expressão de teste_1)
    instrução_1;
else if (expressão de teste_2)
    instrução_2;
```

# Exemplo

```
1  #include <conio.h>
2  #include <string.h>
3  #include <stdio.h>
4
5  void main(){
6
7      /* declaração de variáveis*/
8      int totalHoras;
9      float salarioMensal, valorHora, gratificacao;
10
11     //Atribui valor da hora de trabalho e gratificacao
12     valorHora = 25.5;
13     gratificacao = 500;
14     salarioMensal=0;
15
16     //Recebe total de horas
17     printf("Total horas:"); scanf("%d",&totalHoras);
18
19     if((totalHoras > 80)&&(totalHoras<160)){
20         valorHora = 20;
21         salarioMensal = totalHoras*valorHora - 0.2*gratificacao;
22     }
23
24     else if(totalHoras==160)
25         salarioMensal = totalHoras*valorHora + gratificacao;
26
27     else if(totalHoras > 160){
28         valorHora = 30;
29         salarioMensal = totalHoras*valorHora + 1.1*gratificacao;
30     }
31
32     printf("Salario mensal: R$ %5.2f", salarioMensal);
33
34 }
```

# Problema do else oscilante

---

- O else é opcional dentro de uma estrutura do tipo if-else
- A omissão de um else em uma sequência de ifs aninhados poderá gerar ambiguidade
- O compilador associa o else ao if mais recente (interno) que esteja sem else

# Problema do else oscilante

---

Exemplo 1:

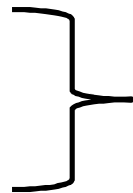
```
If(n>0)
```

```
  if(a>b)
```

```
    z=a;
```

```
  else
```

```
    z=b;
```



Aqui o else  
corresponde  
ao if mais  
interno

Exemplo 2:

```
If(n>0){
```

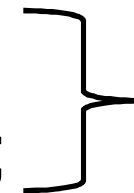
```
  if(a>b)
```

```
    z=a;
```

```
}
```

```
else
```

```
  z=b;
```



Aqui o else  
corresponde  
ao if mais  
externo

# Exemplo

```
1  #include <conio.h>
2  #include <string.h>
3  #include <stdio.h>
4
5  void main(){
6
7      /* declaração de variáveis*/
8      int valor1, valor2;
9
10     printf("Valor 1 = "); scanf("%d",&valor1);
11     printf("Valor 2 = "); scanf("%d",&valor2);
12
13     if(valor1>10)
14         if(valor2>10)
15             printf("valor1 > 10 e valor 2 > 10");
16     else
17         printf("valor1 <= 10");
18
19
20 }
21
22
```

**ERRO!!**

```
Valor 1 = 11
Valor 2 = 10
valor1 <= 10
```

**Correto**

```
if(valor1>10){
    if(valor2>10)
        printf("valor1 > 10 e valor 2 > 10");
}
else
    printf("valor1 <= 10");
```

# Estrutura CASE

---

- Comando de decisão alternativo ao <else-if>
- Considerado mais flexível e de formato mais claro, principalmente quando o programa precisa escolher “uma” entre várias alternativas de decisão
- Indicada para selecionar entre casos mutuamente exclusivos



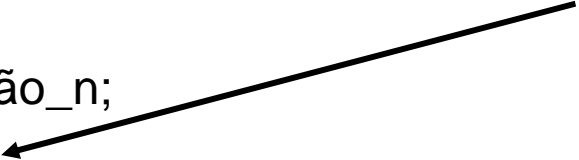
# Estrutura CASE

---

switch (expressão constante)

```
{  
  case constante_1:  
    instrução_1;  
    ...  
    instrução_n;  
    break;  
  case constante_2:  
    instrução_1;  
    ...  
    instrução_n;  
    break;  
  default:  
    instrução_1;  
    ...  
    instrução_n;  
}
```

O que ocorre se remover o <break> ?



# Exemplo com CASE

```
int main()
{
    float num1, num2;
    char op;
    printf("Digite na seguinte ordem: valor 1 \"operador\" valor 2 \n");
    scanf("%f %c %f", &num1, &op, &num2);

    switch(op)
    {
        case '+':
            printf(" = %f\n", num1 + num2);
            break;

        case '-':
            printf(" = %f\n", num1 - num2);
            break;

        default:
            printf("Operador desconhecido \n");
    }

    system("PAUSE");
}
```

# Estrutura CASE

---

- Variável é obrigatoriamente um char, unsigned char, int, unsigned int, short int, long ou unsigned long
- O valor constante em cada case deve corresponder a valores possíveis de serem atribuídos à variável
- Se um valor não tem um case associado, default é executado. Todavia, default é opcional
- O comando break finaliza a execução do switch, mas seu uso não é obrigatório

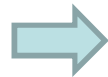
# Exemplo

```
1  #include <conio.h>
2  #include <string.h>
3  #include <stdio.h>
4
5  void main(){
6
7      /* declaração de variáveis*/
8      int totalHoras;
9      float salarioMensal, valorHora, gratificacao;
10
11
12
13     //Recebe total de horas e valor da hora de trabalho
14     printf("Total horas:"); scanf("%d",&totalHoras);
15     printf("valor hora:"); scanf("%f",&valorHora);
16
17     switch (totalHoras){
18         case 80: gratificacao = 500;
19             break;
20         case 160: gratificacao = 1000;
21             break;
22         case 180: gratificacao = 1500;
23             break;
24
25         default: gratificacao = 200;
26     }
27
28     salarioMensal = valorHora*totalHoras + gratificacao;
29
30     printf("Salario mensal: R$ %5.2f", salarioMensal);
31
32 }
33
34
```

# Operador condicional ternário

---

```
If (condição)
    expressão_1;
else
    expressão_2;
```



```
condição ? expressão_1 : expressão_2;
```

- Maneira compacta de expressar uma simples instrução if-else
- Único operador ternário em C

Exemplo:

```
max = (num1 > num2) ? num1 : num2;
```

# Exemplo

---

```
if(totalHoras < 160)
    salarioMensal = 1000.;

else
    salarioMensal = 2500;

printf("Salario mensal: R$ %5.2f", salarioMensal);

salarioMensal = totalHoras < 160 ? 1000 : 2500;

printf("Salario mensal: R$ %5.2f", salarioMensal);
```

# Exercício

---

- Faça um programa que receba o código correspondente ao cargo de um funcionário e seu salário atual. Exiba o cargo, o valor do aumento e seu novo salário.

Código	Cargo	Percentual
1	Escriturário	50%
2	Secretário	35%
3	Caixa	20%
4	Gerente	10%
5	Diretor	Não tem aumento

# Referências

---

Ascencio AFG, Campos EAV. Fundamentos de programação de computadores. São Paulo : Pearson Prentice Hall, 2006. 385 p.

VICTORINE VIVIANE MIZRAHI, Treinamento em Linguagem C – Módulo 1 e Módulo 2, Makron Books, 1990.

Kernighan, B.W.; Ritchie, D.M. C, *A Linguagem de Programação: padrão ANSI*. 2ª Edição. Rio de Janeiro: Campus, 1989, 290p.



---

# FIM Aula 6

---