

# Estruturas de Desvios

## Parte 3

# Sumário

- Aninhando sentenças
  - If
  - If-else
- Ambiguidade do else
- Construção else-if
- Exercício

# Aninhando Sentenças

É possível colocar uma sentença condicional dentro da outra

```
1.  /* Programa "positivo-negativo e par-ímpar" */
2.  #include <iostream>
3.  using namespace std;
4.  int main ()
5.  {
6.    int num;
7.    cout << "Entre com um numero inteiro: ";
8.    cin >> num;
9.    if (num >= 0){
10.     if (num % 2 == 0)
11.       cout << "O numero e par e positivo\n";
12.     else
13.       cout << "O numero e impar e positivo\n";
14.    }
15.    else {
16.     if (num % 2 == 0)
17.       cout << "O numero e par e negativo\n";
18.     else
19.       cout << "O numero e impar e negativo\n";
20.    }
21.    return 0;
22. }
```

# Ambiguidade do else

```
/* Programa "numero de peras" */
```

```
1. #include <iostream>
```

```
2. using namespace std;
```

```
3. int main ()
```

```
4. {
```

```
5.     int num;
```

```
6.     // Obtem um numero do usuario
```

```
7.     cout << "Entre com o numero de peras: ";
```

```
8.     cin >> num;
```

```
9.     // Imprime uma mensagem dizendo se o numero de peras eh 0 ou 1
```

```
10.    // (** isto esta' errado !! **)
```

```
11.    if (num != 0)
```

```
12.        if (num == 1)
```

```
13.            cout << "Voce tem uma pera.\n";
```

```
14.        else
```

```
15.            cout << "Voce nao tem nenhuma pera.\n";
```

```
16.    return 0;
```

```
17. }
```

**Regra de associação:**

Um else está associado com a última ocorrência do if sem else.

# Ambiguidade do else

```
/* Programa "numero de peras" */
```

```
1. #include <iostream>
```

```
2. using namespace std;
```

```
3. int main ()
```

```
4. {
```

```
5.     int num;
```

```
6.     cout << "Entre com um numero inteiro: ";
```

```
7.     cin >> num;
```

```
8.     // Obtem um numero do usuario
```

```
9.     cout << "Entre com o numero de peras: ";
```

```
10.    cin >> num;
```

```
11.    // Imprime uma mensagem dizendo se o numero de peras eh 0 ou 1
```

```
12.    // (** isto está errado !! **)
```

```
13.    if (num != 0)
```

```
14.        if (num == 1)
```

```
15.            cout << "Voce tem uma pera.\n";
```

```
16.        else
```

```
17.            cout << "Voce nao tem nenhuma pera.\n";
```

```
18.    return 0;
```

```
19. }
```

Como o compilador entende

# Aninhando Sentenças

```
1.  /* Programa "numero de peras" */
2.  #include <iostream>
3.  using namespace std;
4.  int main ()
5.  {
6.      int num;
7.      cout << "Entre com um numero inteiro: ";
8.      cin >> num;
9.      // Obtem um numero do usuario
10.     cout << "Entre com o numero de peras: ";
11.     cin >> num;
12.     // Imprime uma mensagem dizendo se o numero de peras eh 0 ou 1
13.     if (num != 0) {
14.         if (num == 1)
15.             cout << "Voce tem uma pera.\n";
16.     }
17.     else
18.         cout << "Voce nao tem nenhuma pera.\n";
19.     return 0;
20. }
```

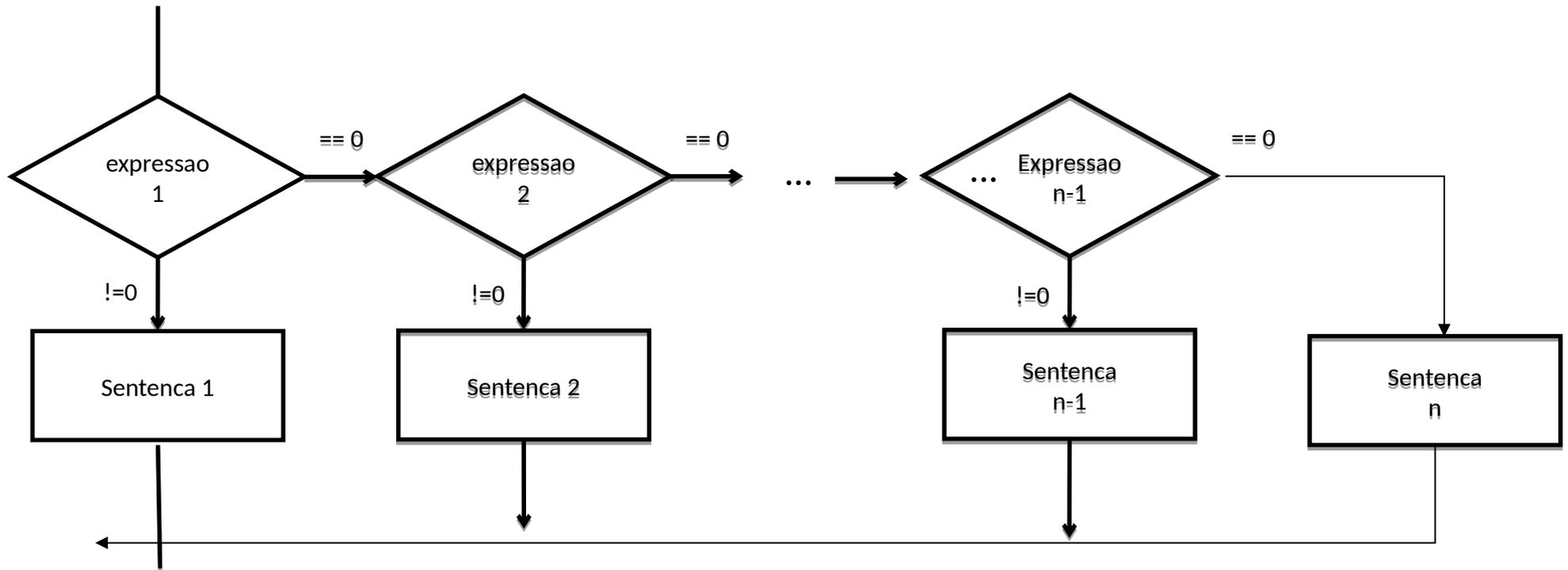
Como resolver?  
Usar chaves!

# Construção else-if

Programar decisões entre diversas alternativas

```
if (expressao1)
    sentenca1
else if (expressao2)
    sentenca2;
else if (expressao3)
    sentenca3
.
.
.
else if (expressão n-1)
    sentenca n-1
else
    sentenca n-1
```

# Construção else-if



- Se uma das expressões for verdadeira, a sentença associada será executada.
- Se nenhuma for verdadeira: a sentença  $n$ , do último *else* será executada como opção padrão.
- Se a opção padrão não for necessária, então a parte *else sentenca n* pode ser removida.

# Exemplo1

```
1.  /* Programa "else-if de três opções" */
    #include <iostream>
2.  using namespace std;
3.  int main()
4.  {
5.      int num1, num2;
6.
7.      // Obtem dois numeros do usuário
8.      cout << "Entre com um numero: ";
9.      cin >> num1;
10.     cout << "Entre com um outro numero: ";
11.     cin >> num2;
12.
13.    // Imprime a mensagem de comparação
14.    if (num1 == num2)
15.        cout << "Os numeros sao iguais\n";
16.    else if (num1 < num2)
17.        cout << "O primeiro numero e menor\n";
18.    else
19.        cout << "O primeiro numero e maior\n";
20. }
```

else-if de três opções.

# Exemplo 2

```
1.  /* Programa "numeros" */
    #include <iostream>
    using namespace std;
2.  int main( )
3.  {
4.      int num1, num2;
5.      char op;
6.      // Obtem uma expressão do usuário
7.      cout << "Entre com numero operador numero\
n ";
8.      cin >> num1 >> op >> num2;
9.      // Imprime o resultado da operação
10.     if (op == '+')
11.         cout << " = " << num1 + num2;
12.     else if (op == '-')
13.         cout << " = " << num1 - num2;
14.     else if (op == '/')
15.         cout << " = " << num1 / num2;
16.     else if (op == '*')
17.         cout << " = " << num1 * num2;
18.     else
19.         cout << " Operador invalido.";
20.     cout << endl;
21. }
```

Entre com numero  
operador numero:  
5 \* 3.5  
= 17.50

Entre com numero  
operador numero:  
10 + 0  
= 10.00

Entre com numero  
operador numero:  
10 x 5.0  
Operador invalido

## Exercícios para aula *online*

Após assistir todas as vídeo-aulas da semana, procure trabalhar na **Lista de exercícios** do Tópico **Estruturas de Desvios**, na sala virtual da disciplina na UFPR Virtual.

Estes exercícios serão usados nas aulas *online* para esclarecer e consolidar os conceitos abordados até aqui.



# Leitura complementar

Acesse o **Material complementar** do Tópico **Estruturas de Desvios**, na sala virtual da disciplina da UFPR Virtual.

Elas são importantes e auxiliam na compreensão dos temas abordados até aqui.

**Créditos:** O conteúdo original deste documento é de autoria da Profª Carmem Satie Hara (DINF/ET), e foi adaptado pela Profa. Michele Nogueira (DINF/ET) e Lígia F. Borges (Doutoranda PPGInf, UFPR) para uso na disciplina *Programação de Computadores* (CI208, CI180, CI183)

Compartilhe este documento de acordo com a licença abaixo



Este documento está licenciado com uma Licença *Creative Commons Atribuição-NãoComercial-SemDerivações* 4.0 Internacional.  
<https://creativecommons.org/licenses/by-nc-sa/4.0/>