



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE

COMANDOS DE DESVIO CONDICIONAL ANINHADOS E OPERADORES LÓGICOS

Disciplina: Algoritmos

Professora: Alba Lopes

alba.lopes@ifrn.edu.br

<http://docente.ifrn.edu.br/albalopes>

Baseado nos slides de aula do prof. Jalerson Lima

RELEMBRANDO COMANDO CONDICIONAL

- O comando de desvio condicional permite executar um bloco de código quando uma expressão lógica for verdadeira

```
if <expressão-lógica>{  
    <bloco de código>  
}
```

- Um **se** pode estar associado a um **senão**, que executa um bloco de código quando a expressão lógica é falsa

```
if <expressão-lógica> {  
    <bloco-de-código>  
}else{  
    <bloco-de-código>  
}
```

INTRODUÇÃO

- Os comandos de desvio condicional podem estar aninhados
 - Ou seja, um comando dentro do outro

```
if <expressão-lógica> {  
    if <expressão-lógica> {  
        <bloco-de-código>  
    }  
}
```



EXEMPLO 1

- Crie um algoritmo que leia a nota de um aluno e mostre se ele está aprovado (≥ 60), reprovado (< 20), e em recuperação (< 60)

```
public class SituacaoAluno{
    public static void main(String [] args){
        int nota;
        nota = Leia.inteiro();
        if (nota >= 60) {
            System.out.println("Aprovado!");
        }else{
            if (nota >= 20){
                System.out.println("Recuperação!");
            }else{
                System.out.println("Reprovado!");
            }
        }
    }
}
```

EXEMPLO 2

- O imposto de renda de uma pessoa varia segundo uma tabela.
 - Se o salário for menor do que R\$ 1.000,00, não há imposto
 - Se for entre R\$ 1.000,00 e R\$ 2.200,00, o imposto é de 13% do valor do salário
 - Se for maior do que R\$ 2.200,00, o imposto é de 22%
 - Faça um algoritmo que dado um valor, em reais, correspondente a um salário, informe o valor que será recebido (total menos imposto).



EXEMPLO 2

```
public class Imposto {  
  
    public static void main(String[] args) {  
        float salario_bruto, salario_liquido, imposto;  
        System.out.println("Digite o valor do salario bruto: ");  
        salario_bruto = Leia.real();  
        if (salario_bruto < 1000){  
            imposto = 0;  
            System.out.println("Não há imposto a pagar");  
        }else{  
            if (salario_bruto < 2200){  
                imposto = (salario_bruto * 13)/100;  
            }else{  
                imposto = (salario_bruto * 22)/100;  
            }  
        }  
        salario_liquido = salario_bruto - imposto;  
        System.out.println("O salario liquido é: " + salario_liquido);  
    }  
}
```

EXEMPLO 2

- Faça um algoritmo que leia o IMC (Índice de Massa Corporal) de uma pessoa.
 - Se o resultado for abaixo de 18,5, informe uma mensagem de que a pessoa está abaixo do peso;
 - Se o resultado for entre 18,5 e 24,99, informe uma mensagem de que a pessoa está com peso normal
 - Se o resultado for acima de 25, informe uma mensagem de que a pessoa está acima do peso.



OPERADORES LÓGICOS

- Os operadores lógicos são usados nas expressões lógicas
 - Operadores lógicos binários: que combinam duas expressões lógicas
 - e (&&)
 - ou (||)
 - Operador lógico unário: que precisa de apenas uma expressão lógica
 - não (!)



OPERADORES LÓGICOS

- O operador lógico 'e' combina duas expressões lógicas, formando uma expressão lógica maior

`<expressão-lógica-a> && <expressão-lógica-b>`

- A expressão lógica maior vai ser verdadeira quando as duas expressões lógicas forem verdadeiras



OPERADORES LÓGICOS

○ Exemplos:

- Resultam verdadeiro:

$3 < 4 \ \&\& \ 6 > 2$

$5 > 2 \ \&\& \ 7 > 1 \ \&\& \ 4 \geq 4$

- Resultam falso:

$10 < 5 \ \&\& \ 7 > 4$

$20 > 10 \ \&\& \ 4 < 7 \ \&\& \ 6 < 6$

TABELA VERDADE DO E (&&)

A	B	A && B
V	V	V
V	F	F
F	V	F
F	F	F



EXEMPLO 1

- Crie um algoritmo que leia 3 (A, B, C) valores, que representam lados de um triângulo. Um triângulo é válido quando a soma de dois lados é maior que o terceiro lado:
 - $A + B > C$ e $A + C > B$ e $B + C > A$



EXEMPLO 1

```
public class Triangulo {
    public static void main(String [] args){
        int a, b, c;
        int soma_ab, soma_bc, soma_ac;
        System.out.println("Digite o valor do lado A: ");
        a = Leia.inteiro();
        System.out.println("Digite o valor do lado B: ");
        b = Leia.inteiro();
        System.out.println("Digite o valor do lado C: ");
        c = Leia.inteiro();
        soma_ab = a + b;
        soma_bc = b + c;
        soma_ac = a + c;
        if ( (soma_ab > c) && (soma_bc > a) && (soma_ac > b) ){
            System.out.println("Os valores formam um triângulo.");
        }else{
            System.out.println("Os valores não formam um triângulo.");
        }
    }
}
```



EXEMPLO 2

- Crie um algoritmo que leia uma nota de 0 a 100 e mostre o conceito relativo à nota:
 - Se a nota estiver entre 80 e 100: conceito A
 - Se a nota estiver entre 60 e 79: conceito B
 - Se a nota estiver entre 40 e 59: conceito C
 - Se a nota estiver entre 20 e 39: conceito D
 - Se a nota for menor que 20: conceito E



EXEMPLO 2

```
public class Conceito {
    public static void main(String[] args) {
        int nota;
        System.out.println("Digite a nota de 0 a 100");
        nota = Leia.inteiro();
        if ( (nota >= 80) && (nota <= 100)) {
            System.out.println("Conceito A");
        } else {
            if ( (nota >= 60) && (nota < 80)) {
                System.out.println("Conceito B");
            } else {
                if ( (nota >= 40) && (nota < 60) ) {
                    System.out.println("Conceito C");
                } else {
                    if ( (nota >= 20) && (nota < 40) ) {
                        System.out.println("Conceito D");
                    } else {
                        System.out.println("Conceito E");
                    }
                }
            }
        }
    }
}
```



OPERADOR LÓGICO OU

- O operador lógico 'ou' também combina duas expressões lógicas, formando uma expressão lógica maior

`<expressão-lógica-a> || <expressão-lógica-b>`

- A expressão lógica maior vai ser verdadeira quando pelo menos uma das expressões lógicas forem verdadeiras
- Quando as duas expressões forem verdadeiras, o resultado é verdadeiro



TABELA VERDADE DO OU (||)

A	B	A B
F	F	F
F	V	V
V	F	V
V	V	V



EXEMPLO 1

- Informe se um dado ano (informado pelo usuário) é ou não bissexto. Dica: um ano é bissexto:
 - Se ele for divisível por 400 ou;
 - Se ele for divisível por 4 e não por 100.

```
public class AnoBissexto {  
  
    public static void main(String [] args) {  
        int ano;  
        System.out.println("Digite o ano: ");  
        ano = Leia.inteiro();  
        if ( (ano % 400 == 0) || ( (ano % 4 == 0) && (ano % 100 != 0) ) ){  
            System.out.println("O ano " + ano + " é bissexto. ");  
        }else{  
            System.out.println("O ano " + ano + " não é bissexto. ");  
        }  
    }  
}
```

OPERADOR LÓGICO NÃO (!)

- O operador lógico 'nao' nega uma expressão lógica
 - Ou seja, esse operador inverte o valor lógico

! <expressão-lógica>

- Quando uma expressão lógica é verdadeira, e é usado o operador 'nao' sobre ela, a expressão lógica passa a ser falsa
- Quando uma expressão lógica é falsa, e é usado o operador 'nao' sobre ela, a expressão lógica passa a ser verdadeira



TABELA VERDADE DO NÃO

A	$\neg A$
V	F
F	V



EXERCÍCIOS

1. Faça um programa que receba três números e mostre-os em ordem crescente.
2. Leia a idade e o tempo de serviço de um trabalhador e escreva se ele pode ou não se aposentar. As condições para aposentadoria são
 - Ter pelo menos 65 anos,
 - Ou ter trabalhado pelo menos 30 anos,
 - Ou ter pelo menos 60 anos e trabalhado pelo menos 25 anos.



EXERCÍCIOS

3. Leia uma data (dia e mês) e determine se ela é válida:
 - Verifique se o mês está entre 1 e 12 e se o dia existe naquele mês. Considere o mês de fevereiro como tendo 29 dias.
4. Escreva um programa que, dada a idade de um nadador, classifique-o em uma das seguintes categorias

Idade	Categoria
Infantil A	5 a 7 anos
Infantil B	8 a 10 anos
Juvenil A	11 a 13 anos
Juvenil B	14 a 17 anos
Sênior	maiores de 18 anos

EXERCÍCIOS

- Mais exercícios na Lista de Exercícios II

