

UNIP
UNIVERSIDADE PAULISTA

Lógica de Programação Estruturada

LPE 01 – Estrutura Sequencial – Parte III

Prof. Msc Wanderley Gonçalves Freitas

Wanderley.unip@gmail.com
www.professor.wanderley.nom.br

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 1 de 33

UNIP
UNIVERSIDADE PAULISTA

Operadores


- Temos três tipos de operadores:
 - Operadores Aritméticos
 - Operadores Relacionais
 - Operadores Lógicos

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 4 de 33

UNIP
UNIVERSIDADE PAULISTA

Agenda

- Operadores
 - Aritméticos (+, -, *, /, ** ou ^)
 - » Resultados numéricos
 - Relacionais (>, <, >=, <=, =, <> ou #)
 - » Resultados lógicos (V ou F)
 - Lógicos (e, ou, não)
 - » Combinam resultados lógicos
- Teste de Mesa



Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 2 de 33

UNIP
UNIVERSIDADE PAULISTA

Operadores

OPERADORES ARIMÉTICO

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 5 de 33

UNIP
UNIVERSIDADE PAULISTA

Operadores

Operadores

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 3 de 33

UNIP
UNIVERSIDADE PAULISTA

Operadores Aritméticos

- **Dados de entrada:** tipo numérico (inteiro ou real)
- **Resultado:** tipo numérico (inteiro ou real)

| Descrição | Algoritmo | Em Java |
|---------------|-----------|---------|
| Multiplicação | * | * |
| Divisão real | / | / |
| Módulo | mod | % |
| Adição | + | + |
| Subtração | - | - |

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 6 de 33

Exercício(1) - Operadores Aritméticos

Considerando as expressões aritméticas abaixo, informe o resultado da operação

1. numero = 3;
2. soma =(2 + 3);
3. calculo = 7 * 7;
4. divisao = 4 / 2;
5. alfa = 8 / 2;
6. resto = 10 % 3;
7. delta = 5 * 5 - 4 * 2 + 6;

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 7 de 33

Operadores Lógicos

- Estabelecem uma relação de comparação entre valores ou expressões
- Resultam sempre em um valor lógico
 - verdadeiro ou falso

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 10 de 33

Exercícios(1)

```

3 public class Exercicio01OperadorAritmetico {
4
5 public static void main(String[] args) {
6     int numero;
7     double soma;
8     int calculo;
9     int alfa;
10    int resto;
11    int delta;
12    int divisao;
13
14    numero = 3;
15    soma=(2.1 + 3);
16    calculo= 7 * 7;
17    alfa= 9 / 2;
18    divisao = 8 / 3;
19    resto= 10 % 3;
20    delta= 5 * 5 - 4 * 2 + 6;
21
22    System.out.println("numero " + numero);
23    System.out.println(" soma " + soma);
24    System.out.println(" calculo " + calculo);
25    System.out.println(" alfa " + alfa);
26    System.out.println(" divisao " + divisao);
27    System.out.println(" resto " + resto);
28    System.out.println(" delta " + delta);
29

```

Declaração de Variável

Operadores

```

<terminated> Exercicio01
numero 3
soma 5.1
calculo 49
alfa 4
divisao 2
resto 1
delta 23

```

O que será impresso na linha 23,25 e 26 estão Corretos?

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 11 de 33

Operadores Lógicos

| Descrição | Algoritmo | Java |
|-----------|-----------|------|
| E | e | && |
| OU | ou | |
| NÃO | Não | ! |

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 11 de 33

Operadores

OPERADORES LÓGICO

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 9 de 33

Operadores Lógicos

Os operadores lógicos mais utilizados são:

- E → &&
- OU → ||
- NÃO → !
- Tais operadores retornam valores lógicos como (V)erdadeiro ou (F)also

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 12 de 33

Exercício(2) - Operadores Lógicos

Avalie as expressões abaixo e classifique o resultado como verdadeiro ou falso:

- condicao1 = true && false;
- condicao2 = false || true;
- condicao3 = ! Condicao1;
- condicao4 = (true && false) || (5 > 3);
- condicao5 = ! condicao2 ;

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 13 de 33

Operadores Lógicos

O operador ! (não) faz a negação de uma sentença:

'Pedro mora na Vila Rica'

Ao aplicar o operador ! (não), a sentença passa a ser: 'Pedro **NÃO** mora na Vila Rica'

| | |
|---|----|
| A | !A |
| V | F |
| F | V |

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 16 de 33

Exercício 02

```

3 public class Exercicio02OperadorLogico {
4
5     public static void main(String[] args) {
6         boolean condicao1;
7         boolean condicao2;
8         boolean condicao3;
9         boolean condicao4;
10        boolean condicao5;
11
12        condicao1 = true && false;
13        condicao2 = false || true;
14        condicao3 = !condicao1;
15        condicao4 = (true && false) || (5 > 3);
16        condicao5 = !condicao2;
17
18        System.out.println(" condicao1 " + condicao1);
19        System.out.println(" condicao2 " + condicao2);
20        System.out.println(" condicao3 " + condicao3);
21        System.out.println(" condicao4 " + condicao4);
22        System.out.println(" condicao5 " + condicao5);
23
24    }
25 }
26
27
    
```

Declaração de Variável

Operadores Lógico

O que será impresso na linha 18,19,20,21,22?

```

<terminated> OperadorLogico
condicao1 false
condicao2 true
condicao3 true
condicao4 true
condicao5 false
    
```

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 17 de 33

Operadores

OPERADORES RELACIONAIS

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 17 de 33

Operadores Lógicos

Tabela-verdade para os operadores && e ||

| A | B | A && B | A B |
|---|---|--------|--------|
| V | V | V | V |
| V | F | F | V |
| F | V | F | V |
| F | F | F | F |

&& – somente resulta em **VERDADEIRO** quando **todas** as sentenças avaliadas são **verdadeiras**

|| – somente resulta em **FALSO** quando **todas** as sentenças avaliadas são **falsas**

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 15 de 33

Operadores Relacionais

| Descrição | Algoritmo | Em Java |
|----------------|-----------|---------|
| Maior | > | > |
| Maior ou igual | >= | >= |
| Menor | < | < |
| Menor ou igual | <= | <= |
| Igualdade | = | == |
| Desigualdade | <> | != |

Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 18 de 33

Exercícios(3)

1 - Determine o resultado lógico das expressões, considerando os seguintes valores: X=4, A=5, B=3, C=4 e D=6.

1. condicao1 = **!(X > 3)** ;
2. condicao2 = (X < 1) **&&** (B > D);
3. condicao3 = (D < 9) **&&** (C > 3);
4. condicao4 = **!(X > 3) ||** (C < 7);
5. condicao5 = (A > B) **||** (C > B);
6. condicao6 = **!(D > 3) ||** !(B < 4);

UNIP UNIVERSIDADE PAULISTA
Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 19 de 33

Exercícios(4)

```

public class Exercicio04OperadorRelacional {
    public static void main(String[] args) {
        boolean condicao1;
        boolean condicao2;
        boolean condicao3;
        boolean condicao4;
        boolean condicao5;

        int X = 8;
        int Y = 5;

        condicao1 = (X == 4) && (Y == 7);
        condicao2 = (X < 3) || (Y != 7);
        condicao3 = (X >= 2) && (Y == 5);
        condicao4 = !(X != 2) && (Y > 4);
        condicao5 = ((X < 5) && (Y > 2)) || (X != 7);

        System.out.println(" condicao1 " + condicao1);
        System.out.println(" condicao2 " + condicao2);
        System.out.println(" condicao3 " + condicao3);
        System.out.println(" condicao4 " + condicao4);
        System.out.println(" condicao5 " + condicao5);
    }
}
    
```

<terminated> Exercicio04OperadorRelacional.java
 condicao1 false
 condicao2 true
 condicao3 true
 condicao4 false
 condicao5 true

O que será impresso na linha 20,21,22,23,24?

UNIP UNIVERSIDADE PAULISTA
Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 20 de 33

Exercícios(3)

```

public class Exercicio03OperadorRelacional {
    public static void main(String[] args) {
        boolean condicao1;
        boolean condicao2;
        boolean condicao3;
        boolean condicao4;
        boolean condicao5;
        boolean condicao6;

        int X = 4;
        int A = 5;
        int B = 3;
        int C = 4;
        int D = 6;

        condicao1 = !(X > 3);
        condicao2 = (X < 1) && (B > D);
        condicao3 = (D < 9) && (C > 3);
        condicao4 = !(X > 3) || (C < 7);
        condicao5 = (A > B) || (C > B);
        condicao6 = !(D > 3) || !(B < 4);

        System.out.println(" condicao1 " + condicao1);
        System.out.println(" condicao2 " + condicao2);
        System.out.println(" condicao3 " + condicao3);
        System.out.println(" condicao4 " + condicao4);
        System.out.println(" condicao5 " + condicao5);
        System.out.println(" condicao6 " + condicao6);
    }
}
    
```

condicao1 false
 condicao2 false
 condicao3 true
 condicao4 true
 condicao5 true
 condicao6 false

O que será impresso na linha 23,24,25,26,27,28?

UNIP UNIVERSIDADE PAULISTA
Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 21 de 33

Teste de Mesa

Teste de Mesa

UNIP UNIVERSIDADE PAULISTA
Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 23 de 33

Exercícios(4)

Considerando X = 8 e Y = 5, avalie as expressões abaixo e classifique o resultado como verdadeiro ou falso:

- a) condicao1 = (X == 4) **&&** (Y == 7);
- b) condicao2 = (X < 3) **||** (Y != 7);
- c) condicao3 = (X >= 2) **&&** (Y == 5);
- d) condicao4 = **!(X != 2) &&** (Y > 4);
- e) condicao5 = ((X < 5) **&&** (Y > 2)) **||** (X != 7);

UNIP UNIVERSIDADE PAULISTA
Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 21 de 33

Teste de Mesa

Objetivos

- Aprender a **verificar** se o algoritmo (ou programa) leva a um resultado esperado através de simulação de valores, utilizando a técnica de **TESTE DE MESA**

UNIP UNIVERSIDADE PAULISTA
Prof. Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 24 de 33

Teste de Mesa

- O teste de mesa **simula** a execução de um algoritmo sem utilizar o computador, empregando apenas “papel e caneta” (ou melhor sem utilizar um compilador ou interpretador).

Prof: Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 25 de 33

Teste de Mesa

Passos para realizar o teste de mesa

- Para indicar que o valor de uma variável **foi lido**, envolva-o entre **parênteses**
- Se o valor foi **escrito** pela instrução, envolva-o entre **chaves**
- Para valores indefinidos, isto é, aqueles que ainda não foram determinados até uma dada instrução, utilize a **interrogação**

| LINHA | a | b | c |
|-------|------|-------|---|
| 5 | (18) | ? | ? |
| 7 | | | |
| 11 | | | |
| 13 | | {7.5} | |

Prof: Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 28 de 33

Teste de Mesa

Passos para realizar o teste de mesa

- Identifique as variáveis envolvidas em seu algoritmo;
- Crie uma tabela com linhas e colunas, em que:
 - cada **coluna** representará uma variável a ser “observada”
 - as **linhas** corresponderão às instruções observadas pelo teste de mesa e
 - a **primeira coluna** deverá identificar os números das linhas correspondentes às instruções observadas;

| LINHA | a | b | c |
|-------|---|---|---|
| 5 | ? | ? | ? |
| 7 | ? | ? | ? |

Prof: Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 26 de 33

Exercícios(6)

1. Faça o teste de mesa para encontrar o valor final das variáveis a, b, c e x. Depois escreva um **ALGORITMO** para analisar as instruções abaixo e exibe o resultado para usuário:

- a=3;
- b=4;
- c=5;
- x=(a + 1) * b;
- b=a;
- x=(b- 1) * c;
- c = c + 1 ;

| Linha | a | b | c | x |
|-------|---|---|---|----|
| 1 | 3 | | | |
| 2 | | 4 | | |
| 3 | | | 5 | |
| 4 | | | | 18 |
| 5 | | 3 | | |
| 6 | | | | 10 |
| 7 | | | 6 | |

Prof: Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 29 de 33

Teste de Mesa

Passos para realizar o teste de mesa

- De cima para baixo, **preencha cada uma das linhas** da tabela com o número da linha que identifica cada instrução, seguido dos valores assumidos pelas variáveis do programa após a execução daquela instrução.

| LINHA | a | b | c |
|-------|---|---|---|
| 5 | | | |
| 7 | | | |
| 11 | | | |
| 13 | | | |

Prof: Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 27 de 33

Exercícios(7)

Defina os valores finais de A, B, C, D e X. Depois escreva um programa em ALGORITMO para analisar as instruções abaixo e exibe o resultado para usuário

- A = 10;
- B = 20;
- X = 0;
- C = 30;
- D = 40;
- A = D + A;
- A = D + B;
- C = A;
- A = D;
- B = (B + B) + (B*A);
- B = 40;
- A = B - 10;
- A = B + 1;
- X = A + B + C + D;

| Teste de Mesa | | | | | |
|---------------|-----|----|-----|----|----|
| Linha | X | A | B | C | D |
| 1 | 0 | 10 | 20 | 30 | 40 |
| 2 | | 50 | | | |
| 3 | | 60 | | | |
| 4 | | | | 60 | |
| 5 | | 40 | | | |
| 6 | | | 840 | | |
| 7 | | | 40 | | |
| 8 | | 20 | | | |
| 9 | | 41 | | | |
| 10 | 181 | | | | |

Prof: Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 30 de 33

UNIP
UNIVERSIDADE PAULISTA

Referências Bibliográficas

BRUCE, Eckel. **Thinking in Java**. 4 ed. Massachusetts: Editora Prentice hall: 2006.

FAEMAN, Julio. Design Patterns Aplicados. **Java Magazine**, Rio de Janeiro, v. 3, n.20, p. 52 a 58, 2005.
_____, mais Aplicados. Java Magazine, Rio de Janeiro, v. 3, n.21, p. 46 a 55, 2005.

FOWLER, Martin. **UMI essencial : um breve guia para a linguagem** – padrão de modelagem de objetos. Tradução Vera Pezerico e Christian Thomas Prices. 3. ed. Porto Alegre : Bookman, 2005.

FURGERI, Sergio. **Java 2 : ensino didático** : desenvolvendo e implementando aplicações. São Paulo: Editora Érica, 2002.

GRADY, Booch; RUMBAUGH, James; JACOBSON, Ivar. **UML, guia do usuário** . tradução de Fábio Freitas da Silva. Rio de Janeiro: Editora Campus. 2000.

JOHN, Metsker, Steven. **Padrões de projeto em java**. Porto Alegre: Editora Bookman. 2004.

_____. Caminhos para classpath. **Java Magazine**, Rio de Janeiro, v. 3, n.21, p. 16 a 19, 2005. Java Livre.

_____. Datas e Horas em Java. **Java Magazine**, Rio de Janeiro, v. 3, n.22, p. 8 a 13, 2005. Primeiros Passos.

SIERRA, kathy; BATES, Bert. **Certificação Sun para programadores e desenvolvedor Java 2**. Rio de Janeiro: Editora Atlas Books. 2003.

SIERRA, kathy; BATES, Bert. **Head first java 5**. 2 ed. Sebastopol: Editora O' Reilly Media. Books. 2005

Prof: Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 31 de 33

UNIP
UNIVERSIDADE PAULISTA

Perguntas



Prof: Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 32 de 33

UNIP
UNIVERSIDADE PAULISTA

Obrigado

Wanderley

Wanderley.unip@gmail.com

www.wg.pro.br

Prof: Msc Wanderley Gonçalves Freitas LPE 01 – Estrutura Sequencial 33 de 33