








# Java X Fluxogramas

Prof. Dr. Marcelo Fernando Rauber

## Atenção: em Java é Case-Sensitive!

Isso significa que caracteres/comandos/variáveis em MAIÚSCULAS e em minúsculas são tratados de modo diferente. Mesmo uma única letra...

Figura do Fluxograma e Código em C	Significado	Java -Descrição	Java - Exemplos
		Comentários: São linhas/comandos que são ignoradas quando o programa é executado. - Usado para orientar outros desenvolvedores;	Temos dois formatos: // Tudo na direita nesta linha é comentário  /* Aqui está um bloco de comentário... e uma segunda linha comentada */
	Define o início e o fim de um algoritmo, usando as palavras Início ou Fim.	Programa principal inicia no <b>main!</b>  { } São marcadores de bloco de instruções. Significa que as várias coisas, que estão entre os símbolos, serão executadas.  ; O Ponto e vírgula indica o fim de uma instrução.	package ingresso;  public class Ingresso {  public static void main(String[] args) { //Aqui você escreve o seu código  } //Aqui já terminou o programa }
	Identifica as variáveis e constantes do algoritmo, bem como seu tipo primitivo.  <b>Java é fortemente tipado!</b>  Todas as variáveis devem ser declaradas antes de serem utilizadas!	<b>Tipos primitivos:</b> Deve ser a primeira coisa onde você escreve o seu código.  Inteiro - <b>int</b> Real - <b>double</b> Texto - <b>String</b> Caracter – <b>char</b> Lógico - <b>boolean</b>  <b>Ao dar nome para as variáveis use nomes significativos!!!!</b>	int idade; double salario, salario_liquido; String NomeCompleto; char sexo; int x = 8; int a = 3, b = a * 2; int idade2 = 30; boolean casado = true; boolean menorDeldade = idade2 < 18;
	Identifica a sequência de operações, permitindo a conexão entre as outras figuras.	Não há representação. Simplesmente pela sequência de instruções/ linhas.	
	Comando de atribuição (uma pequena seta para esquerda): nos permite armazenar um valor em uma variável, como o resultado de um cálculo.	=  Um sinal de igualdade.	salario = 1500 * 2;
	Usada para o processamento de cálculos e atribuições.	Simplesmente escreva o cálculo em uma linha. Não há comando ou símbolo especial. Mas termine com ;	mensagem = “um teste”;  Resp = salario * 2;
  printf("Digite 2 números");	<b>Saída de dados</b> na tela do computador.	System.out.println( );  Use este comando. Coloque dentro dos parênteses o que você quer que saia na tela. Utilize o sinal de + para juntar texto e variáveis.	System.out.println("Um teste");  System.out.println("O salário é R\$ " + salario);

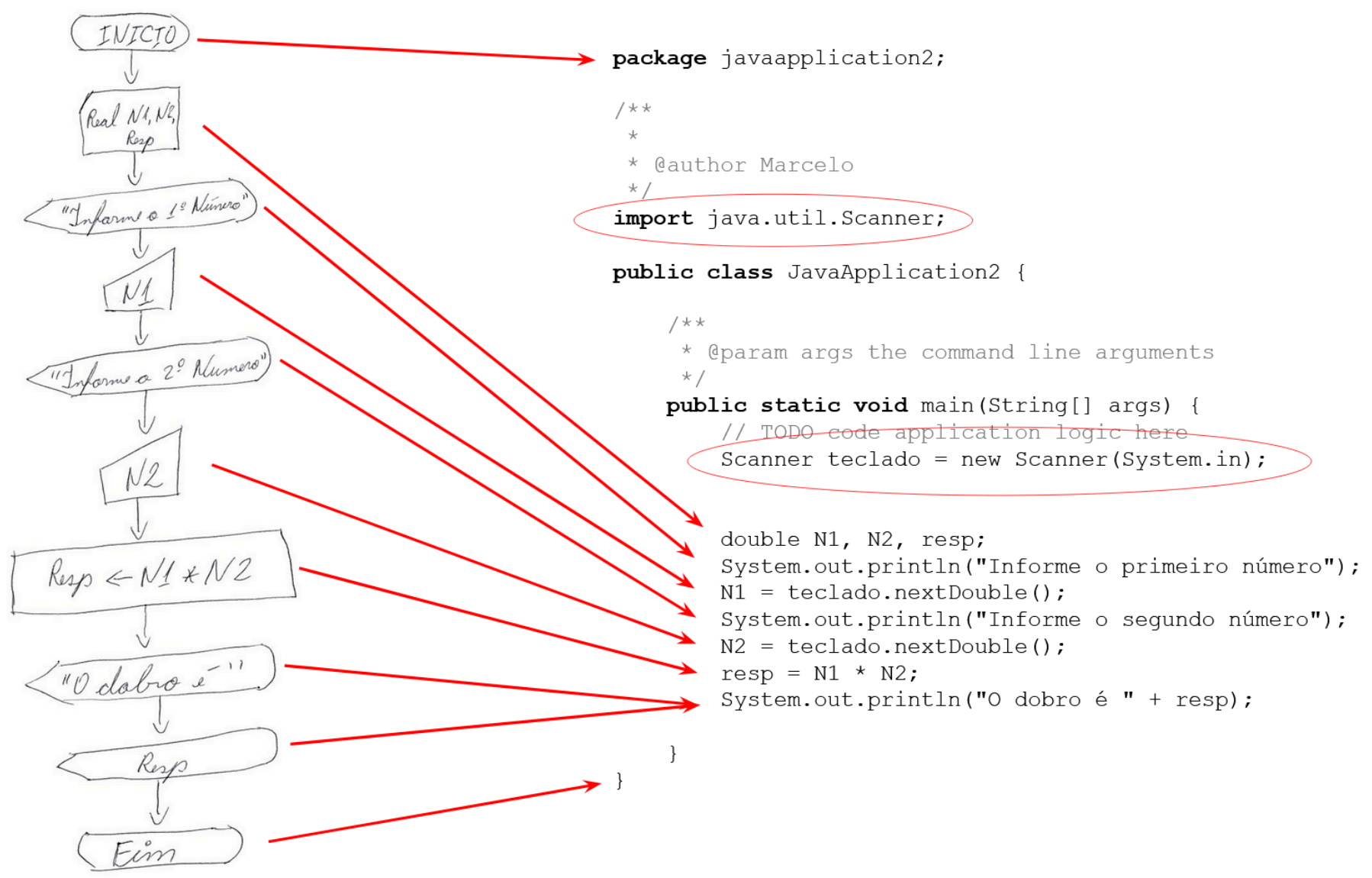
  scanf("\$d %d", &a &b);	<b>Entrada de dados</b> via teclado. Armazenando esse dato em uma variável.	Esse é um pouco mais complicado, pois faz necessário usar uma biblioteca extra, declarar um objeto para controlar a entrada de dados e especificar o tipo de dato da entrada. Mas a linha principal que usamos é do tipo:  x = teclado.nextDouble();	/*Importando uma biblioteca, vai antes de public class... */ <b>import java.util.Scanner;</b>  /* Declarando objeto de controle, pode ser a primeira coisa do programa, dentro do <b>main</b> */ Scanner teclado = new Scanner(System.in);  /*Finalmente, a leitura da variável em si , lembrando que varia conforme o tipo de dato*/ <b>salario = teclado.nextDouble();</b> <b>Nome = teclado.nextLine();</b> <b>idade = teclado.nextInt();</b>

**Observações:** em Java temos vários separadores:

- ; O Ponto e vírgula indica o fim de uma instrução. A quebra de linha não separa instruções.
- . O ponto serve para separar a parte inteira da fracionária em um número, tal como na notação inglesa. Ou seja, enquanto escrevemos 2,5 para representar dois e meio, no nosso código em Java escrevemos: 2.5
- , A vírgula serve para separar itens, elementos, membros, como o que ocorre na atribuição de valores aos vetores, por exemplo: int Janelas, Portas;
- { } Chaves são utilizadas para separar blocos de programação e os colchetes [] são utilizados para indicar/separar os índices de vetores e declará-los.
- " " Aspas são usadas para delimitar uma cadeia de caracteres, isto é, palavras ou frases.
- ' ' Apóstrofes são usados para delimitar um literal do tipo caracter (char).
- + O símbolo de mais, além de ser um operador aritmético, é usado para concatenar saídas de dados, isto é, intercalar um texto e o nome da variável.
- // Comentário de 1 linha, tudo que estiver a direita
- /\* Marca o início de várias linhas de comentário
- E para indicar o fim do comentário, usamos os seguintes símbolos \*/

A indentação do Código é considerada uma boa prática. Mas não define um bloco de instruções. Usamos { }

**Exemplo -** Faça um algoritmo para mostrar o resultado da multiplicação de dois números informados pelo usuário.



	Identifica o desvio condicional, permitindo a escolha entre dois caminhos possíveis.	<p>Temos duas opções:</p> <p>a) Somente com o caminho da verdade;</p> <p>b) Com o caminho da verdade e da falsidade;</p> <p>O comando usado é o if:</p> <pre>if ( <u>Teste Lógico</u> ) {      }</pre> <p>Atenção: <u>Teste Lógico</u> deve ser substituído por uma expressão, usando operadores relacionais/lógicos.</p> <p>DIV    / MOD   %</p> <p>Atenção 2: Em Java não há operador específico para o DIV. Simplesmente use com variáveis inteiras e <u>não</u> será considerada a parte fracionária.</p>	<p>//Exemplo:</p> <p><i>/* a) Desvio condicional simples (somente caminho verdadeiro) */</i></p> <pre>if (salario &gt; 2300) {     System.out.println("Oba!!!!!!"); }</pre> <p><i>/*b) Desvio condicional composto (com caminho verdadeiro e falso) */</i></p> <pre>if (salario &gt; 1200) {     //Se for verdade executa esse código     System.out.println("Um bom salário"); } else {     //Se o teste do if der falso executa esses     resp = x + 2 * 3;     System.out.println("Podia melhorar "); }</pre>
	Conector: Conecta fluxos que não podem ser representados em uma mesma página. Devem ser sequencialmente numerados para possibilitar identificar os fluxos.	Não tem.	

