

Universidade Federal da Paraíba

Centro de Informática

Departamento de Informática

Linguagem de Programação I

Estruturas de Controle em Java

▶ Tiago Maritan

▶ tiago@ci.ufpb.br

Motivação

- ▶ O controle do fluxo da execução em Java utiliza os mesmos comandos existentes em outras linguagens
 - ▶ **Decisão:** `if-else`, `switch-case`
 - ▶ **Repetição:** `for`, `while`, `do-while`
 - ▶ **Desvios:** `return`, `continue`, `break`



Estruturas de Decisão



Estrutura if-else

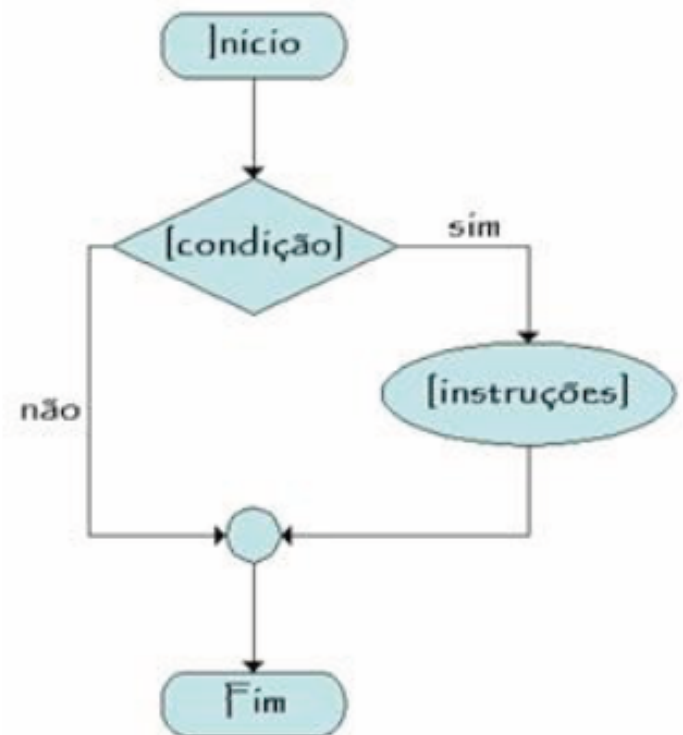
► Estrutura if simples

- Permite executar ou não **instruções**, de acordo com o resultado de uma determinada condição.

► Sintaxe:

```
if (<condição>)  
    <instrução simples>
```

```
if (<condição>)  
{  
    <instruções>  
}
```



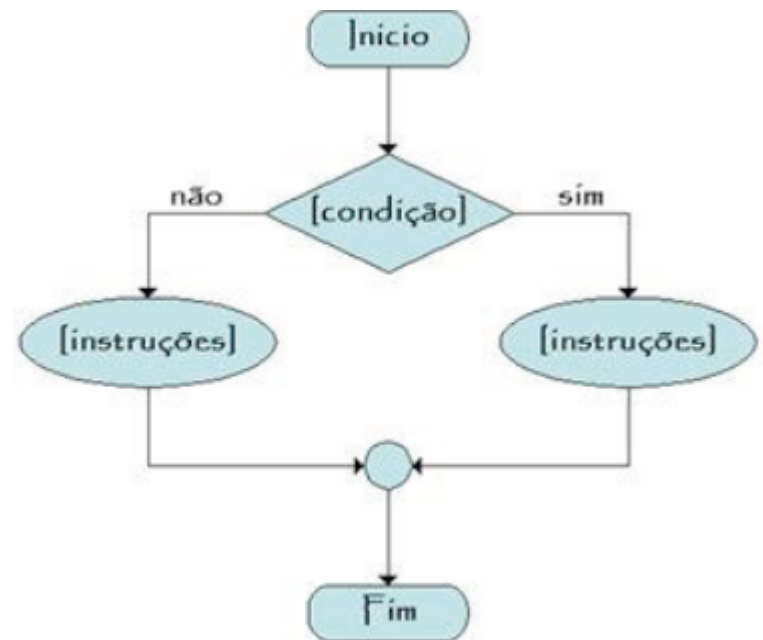
Estrutura if-else

► Estrutura if-else

- Permite selecionar um **entre dois conjuntos de instruções**, de acordo com o resultado de uma determinada condição;

► Sintaxe:

```
if (<condição>)  
{  
    <instruções1>  
}  
else  
{  
    <instruções2>  
}
```



Estrutura if-else

► Estrutura if-else aninhada

- Permite selecionar um entre **múltiplos conjuntos de instruções**;
- **Sintaxe:**

```
if (<condição1>) {  
    <instruções1>  
}  
else if (<condição2>) {  
    <instruções2>  
}  
...  
else {  
    <instruçõesn>  
}
```



Estrutura if-else

- ▶ Exemplo 1: Avaliar performance e calcular bônus do funcionário;

```
if (vendas >= meta){  
    performance = "Satisfatória";  
    bonus = 100;  
}  
else{  
    perfomance = "Insatisfatória";  
    bonus = 0;  
}
```



Estrutura if-else

- ▶ Exemplo 2: Verificar se ano é bissexto;

```
if (ano < 0) {  
    System.out.println("Não é um ano");  
}  
else if ((ano%4 == 0) && (ano%100 != 0)) {  
    System.out.println("É bissexto");  
}  
else if (ano%400 == 0) {  
    System.out.println("É bissexto");  
}  
else {  
    System.out.println("Não é bissexto");  
}
```



Estrutura switch-case

- ▶ Permite **selecionar um** entre **múltiplos conjuntos de instruções**
 - ▶ Alternativa para o `if-else` aninhado
- ▶ Sintaxe:

```
switch (<seletor_inteiro>){  
    case valor_1:   
        <instruções1>;  
        break;  
    case valor_2:   
        <instruções2>;  
        break;  
    ...  
    default:  
        <instruçõesn>  
}
```

*Expressão que resulte em
um valor inteiro (incl. char)*

*Constante inteira
(incl. char)*

Estrutura switch-case

- ▶ Exemplo 1: Escolher entre as opções 1, 2 e 3;

```
switch (opcao) {  
    case 1:  
        ...  
        break;  
    case 2:  
        ...  
        break;  
    case 3:  
        ...  
        break;  
    default:  
        //entrada invalida  
        ...  
}
```

Estrutura switch-case

- ▶ Exemplo 1: Escolher entre as opções de letras;

```
switch (letra){  
    case 'A':  
        System.out.println("a");  
        break;  
    case 'B':  
        System.out.println("b");  
        break;  
    case 'C':  
        System.out.println("c");  
        break;  
    default:  
        System.out.println("?");  
        ...  
        break;  
}
```

Operador Ternário (? :)

- ▶ Retorna um valor ou outro dependendo do resultado de uma expressão booleana

- ▶ **Sintaxe:** `var = <expressão> ? <valorT> : <valorF>;`

- ▶ Exemplo:

```
String tit = (sex == 'f') ? "Sra." : "Sr.";
int x = (y != 0) ? 50 : 500;
```

- ▶ Cuidado!!!! Código pode ficar difícil de entender;



Estruturas de Repetição



Estruturas while e do-while

► Sintaxe:

```
while (<expressao>)  
{  
    <instruções>;  
}
```

```
do{  
    <instruções>;  
} while (<expressao>);
```



Estruturas while e do-while

► Exemplos:

```
int x = 0;
while (x < 10){
    System.out.println ("item " + x);
    x++;
}
```

```
int x = 0;
do {
    System.out.println ("item " + x);
    x++;
} while (x < 10);
```



Estrutura for

► Sintaxe:

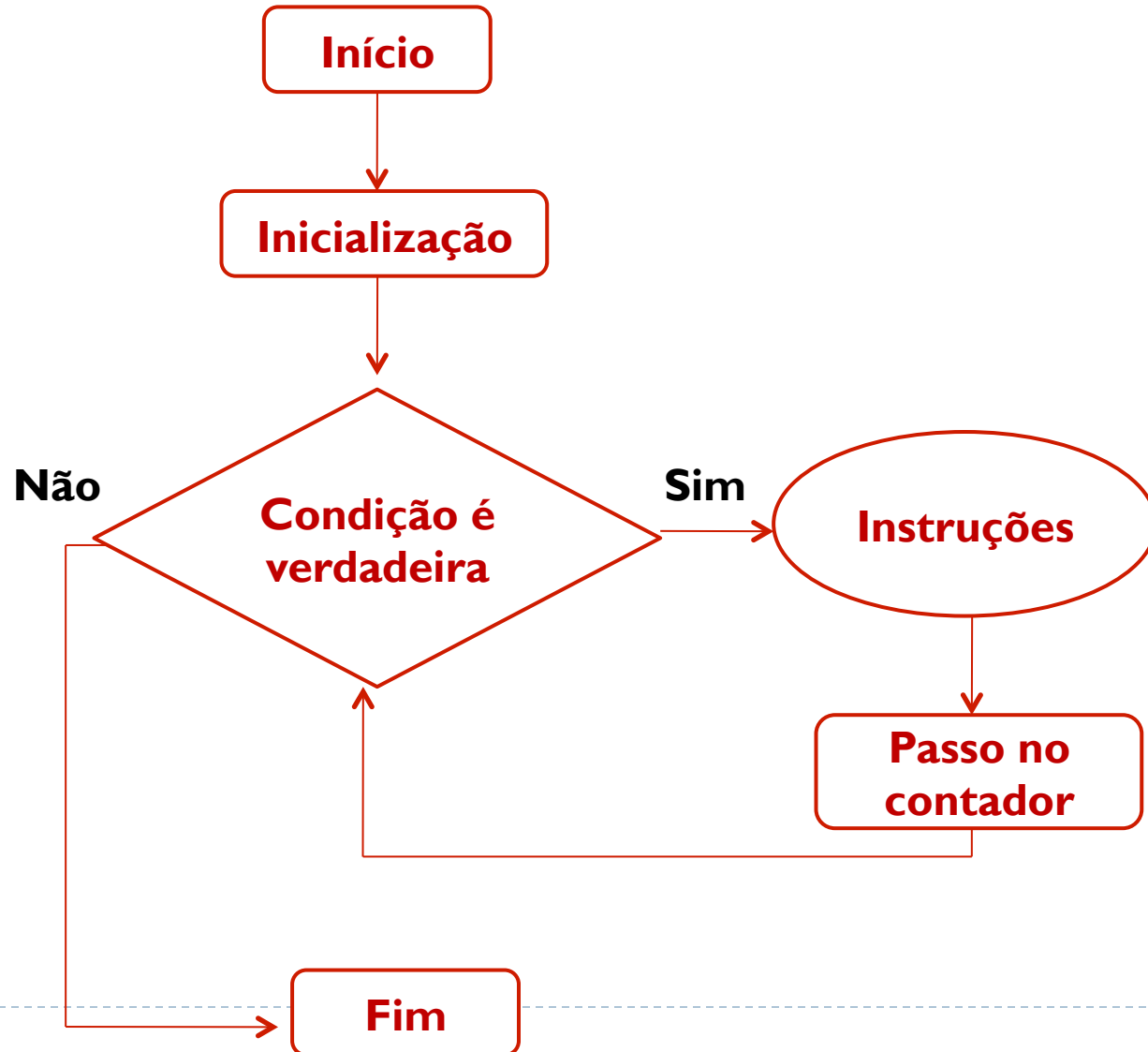
```
for (<inicialização>; <condição>; <passo>)  
{  
    <instruções>;  
}
```

► Exemplos:

```
for (int x = 0; x < 10; x++){  
    System.out.println("item "+ x);  
}
```



Fluxograma do for



Instruções de Desvio (break e continue)

- ▶ **break:** Interrompe a execução do bloco de repetição
 - ▶ Continua com a próxima instrução após o bloco;
- ▶ **continue:** Interrompe a execução da iteração atual
 - ▶ Reinicia o bloco com a próxima iteração;



Instruções de Desvio (break e continue)

► Exemplo

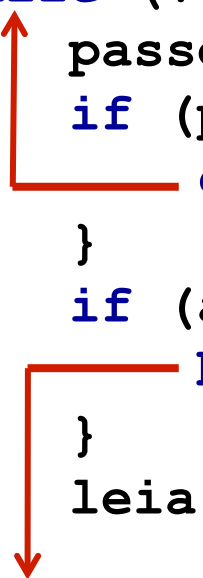
```
while (!terminado) {  
    passePagina();  
    if (paginaDePropaganda == true) {  
        continue; // pule esta iteração  
    }  
    if (alguemChamou == true) {  
        break; // cai fora do while  
    }  
    leia();  
}  
restoDoPrograma();
```



Instruções de Desvio (break e continue)

► Exemplo

```
while (!terminado) {  
    passePagina();  
    if (paginaDePropaganda == true) {  
        continue; // pula esta iteração  
    }  
    if (alguemChamou == true) {  
        break; // cai fora do while  
    }  
    leia();  
}  
restoDoPrograma();
```



Exercícios

1. Escreva um programa em Java que recebe dois números e diz se são iguais ou diferentes e exibe o maior e o menor;
2. **Escreva um programa em Java que receba dois valores pelo teclado: o valor do produto e um código de aumento no valor do produto e apresenta na tela o novo valor do produto depois de acrescida uma porcentagem (conforme Tabela 1). Utilize a estrutura switch-case.**

Código	% de Aumento
1	15
3	20
4	35
8	40

Exercícios

3. **Escreva um programa em Java que recebe um conjunto de valores, calcula e exibe o maior valor inserido. A entrada de dados deve parar quando for digitada o valor 0 (zero). Utilize a estrutura do-while.**
4. **Escreva um programa em Java que calcule e mostre a soma dos 100 primeiros números inteiros ($1+2+3+4+\dots+98+99+100$). Utilize a caixa de diálogo com a classe `JOptionPane` para saída de dados.**
5. Crie um programa em Java que leia um número e imprima o quadrado de todos os números entre 1 e o número informado.



I) Escreva um programa em Java que receba dois valores pelo teclado: o valor do produto e um código de aumento no valor do produto e apresenta na tela o novo valor do produto depois de acrescida uma porcentagem (conforme Tabela I). Utilize a estrutura switch-case.

Código	% de Aumento
1	15
3	20
4	35
8	40



Exercícios

2. Escreva um programa em Java que recebe um conjunto de valores, calcula e exibe o maior valor inserido. A entrada de dados deve parar quando for digitada o valor 0 (zero). Utilize a estrutura do-while.

3. Escreva um programa em Java que calcule e mostre a soma dos 100 primeiros números inteiros ($1+2+3+4+\dots+98+99+100$). Utilize a caixa de diálogo com a classe `JOptionPane` para saída de dados.



Universidade Federal da Paraíba

Centro de Informática

Departamento de Informática

Linguagem de Programação I

Estruturas de Controle em Java

▶ Tiago Maritan

▶ tiago@ci.ufpb.br

