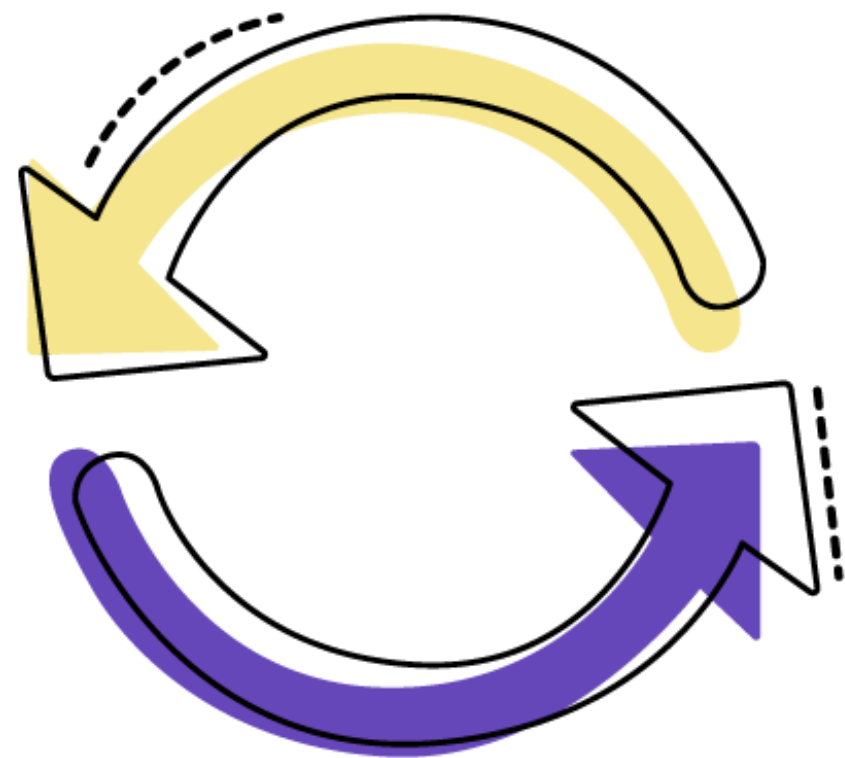




ESTRUTURAS DE REPETIÇÃO II



Este texto-base aborda os seguintes temas:

<u>Conhecendo dois novos Laços de Repetição</u>	<u>Laço de Repetição “enquanto... fim-enquanto”</u>	<u>Laço de Repetição “repita... até que”</u>	<u>Utilizando comandos de repetição em conjunto com comando condicional if</u>
-------------------------------------------------	-----------------------------------------------------	----------------------------------------------	--------------------------------------------------------------------------------



Conhecendo dois novos Laços de Repetição

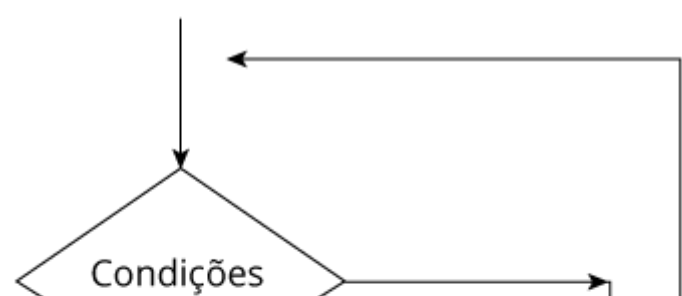
Na unidade anterior, você conheceu o comando “para...fim-para”, utilizado em situações em que você sabe exatamente o número vezes que as instruções deverão ser repetidas. Como exemplo, você estudou programas que realizam instruções repetidamente para exibir aos usuários os números de 0 a 9.

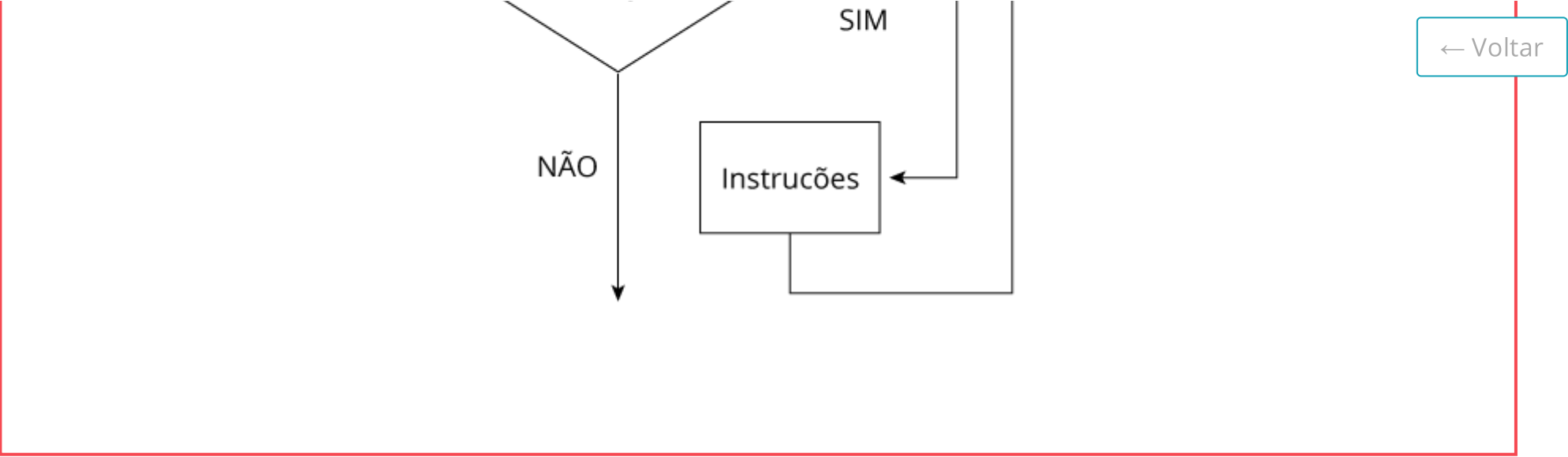
Mas, como fazer em situações nas quais não é possível definir quantas vezes a instrução deverá ser repetida?

Os Laços de Repetição abordados nessa unidade são utilizados nesses contextos. São comandos que só encerram suas repetições quando a condição é satisfeita. Dê uma olhada:

1. Enquanto... fim-enquanto

Essa estrutura, também conhecida por **enquanto** (*while*), avalia a condição e só então executa as instruções indicadas. Veja o fluxograma que representa o comando.





VOLTAR

PRÓXIMO

Agora, você conhecerá cada Estrutura de Repetição com maior profundidade.

Laço de Repetição “enquanto... fim-enquanto”

Este Laço de Repetição, como apresentado anteriormente, trabalha enquanto a condição for verdadeira e vai executando as instruções. Porém, a condição sendo falsa, ele sai do loop e vai para o próximo comando na programação.

PSEUDOCÓDIGO

Enquanto <condição> **faça**

<Comando>

<Comando>

<Comando>

Fim enquanto

FLUXOGRAMA

JAVA

```
while (condição){  
    instrução  
}
```

Agora que você conheceu a definição e a estrutura do laço, veja o exemplo a seguir.

Elabore um Algoritmo, um Fluxograma e um Programa em Java que mostre todos os números menores que 10.



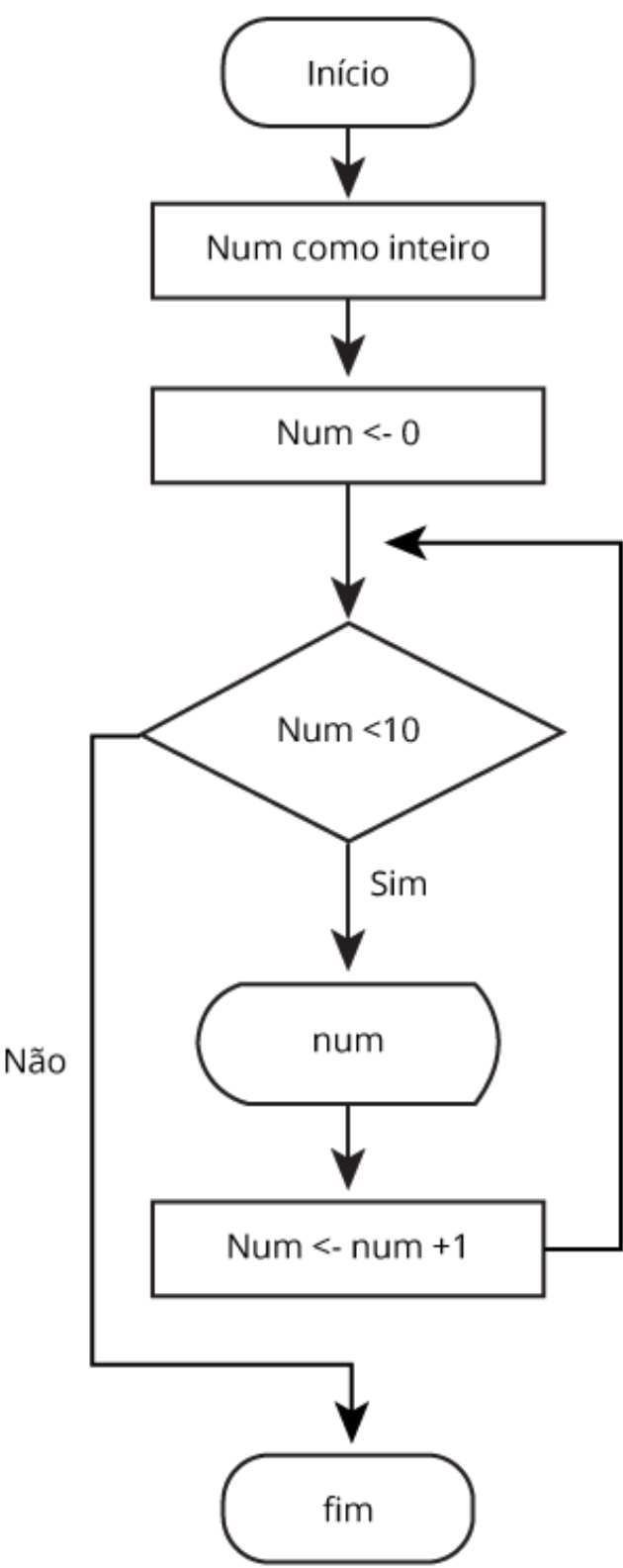
PSEUDOCÓDIGO

Programa ex

Declare
 num como inteiro

Início
 Num <- 0
 Enquanto (num < 10) faça
 escreva num
 num <- num+1
 fim-enquanto

FLUXOGRAMA



Agora, veja o programa em Java.

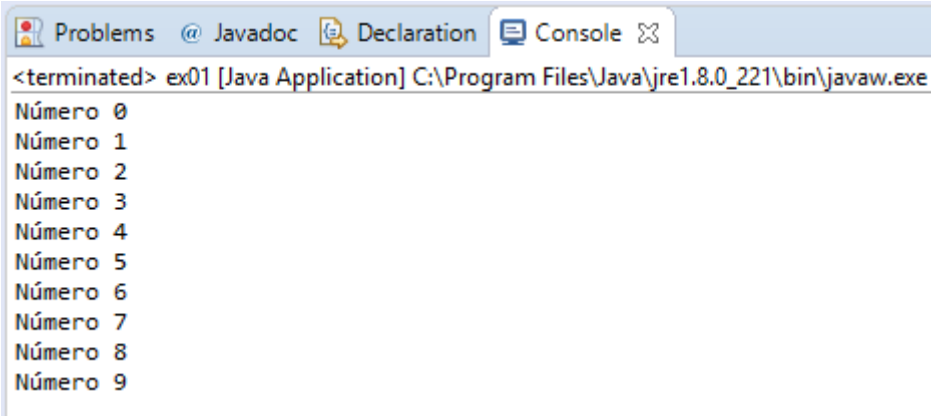
```
1 public class ex01 {
2
3     public static void main(String[]args) {
4         int num =0;
5         while (num < 10) {
6             System.out.println("Número " + num);
7             num++;
8         }
9     }
10 }
```

Copiar

Observe que na linha 5 temos a estrutura de repetição **enquanto** while tendo como condição a situação que o exercício colocou (números menores que 10). As linhas 6 e 7 apresentam os comandos que são executados dentro desta estrutura.

Como resultado, o computador apresenta:

← Voltar

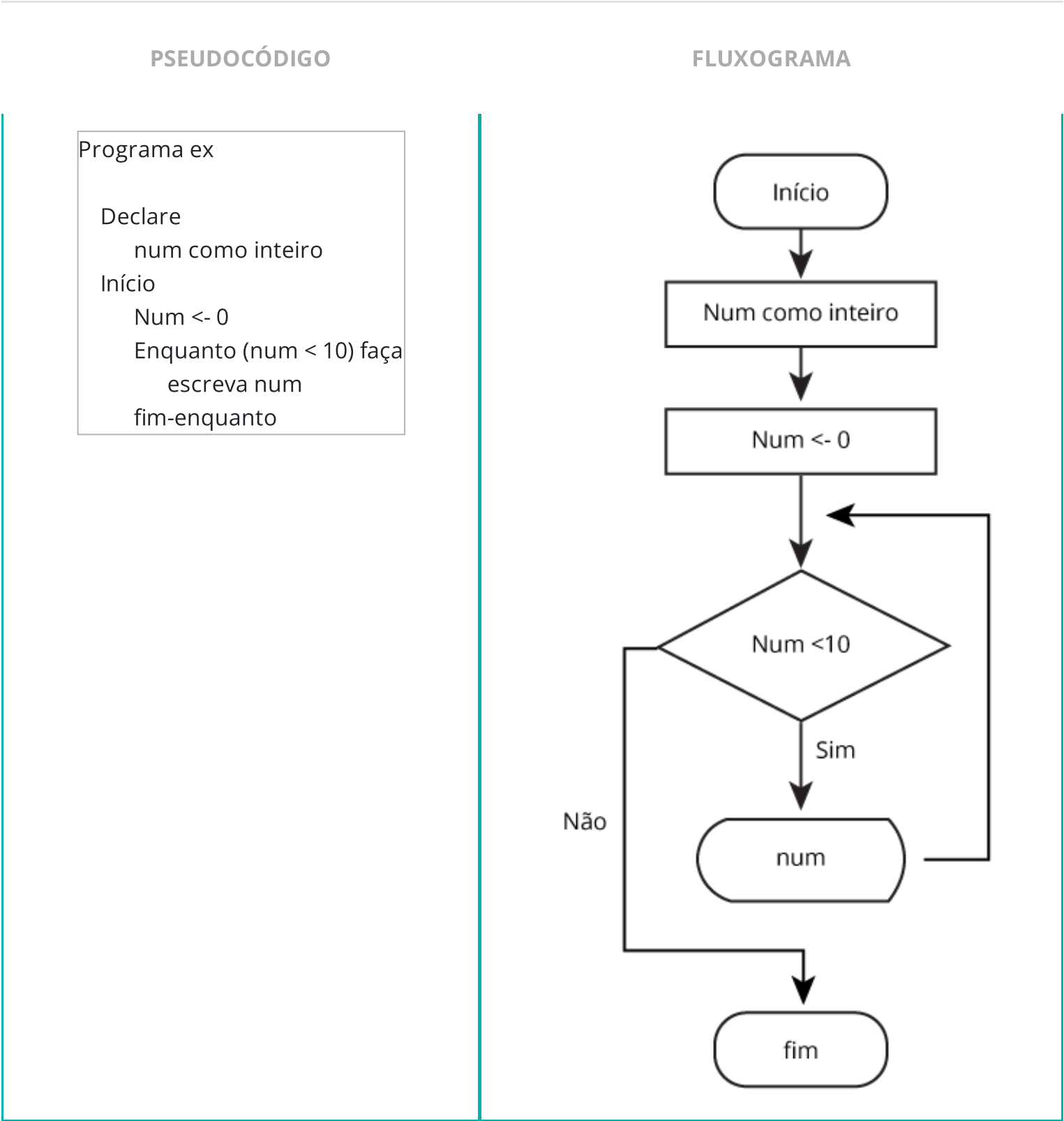


Você percebeu que a condição colocada acima foi “num < 10”, por isso o resultado apresentado são números de 0 até o 9.

Lembre-se sempre: o incremento não é automático

O Laço de repetição Enquanto...Fim-Enquanto **não possui incremento automático** como o laço Para...Fim...Para.

Fique atento a isso! O exemplo a seguir apresenta uma repetição infinita e demonstra as implicações da ausência de contador:



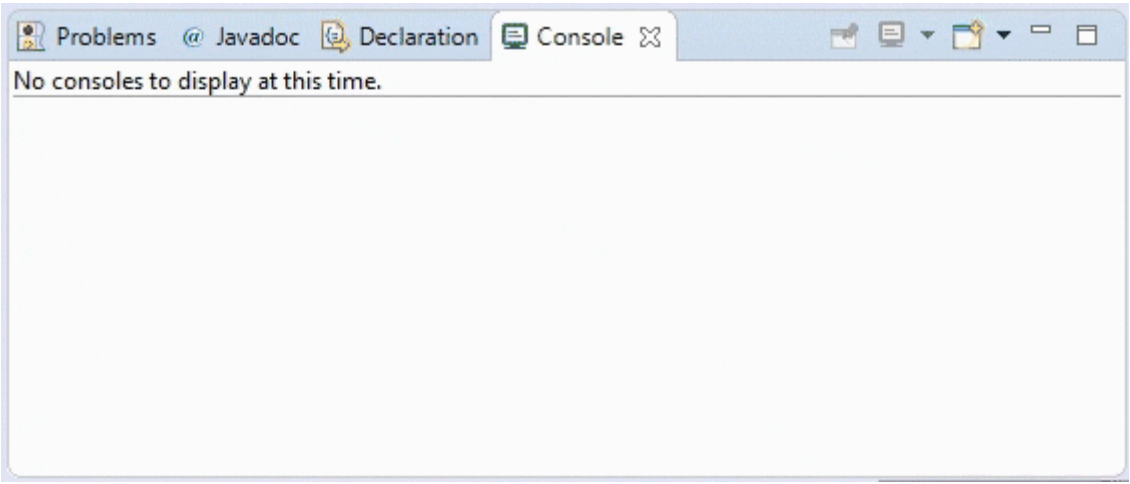
No Java:

```
1 public class exemploLoopInfinito {
2     public static void main(String[] args) {
3         int num = 0;
4         while (num < 10) {
5             System.out.println("Número " + num);
6         }
7     }
8 }
```

Copiar

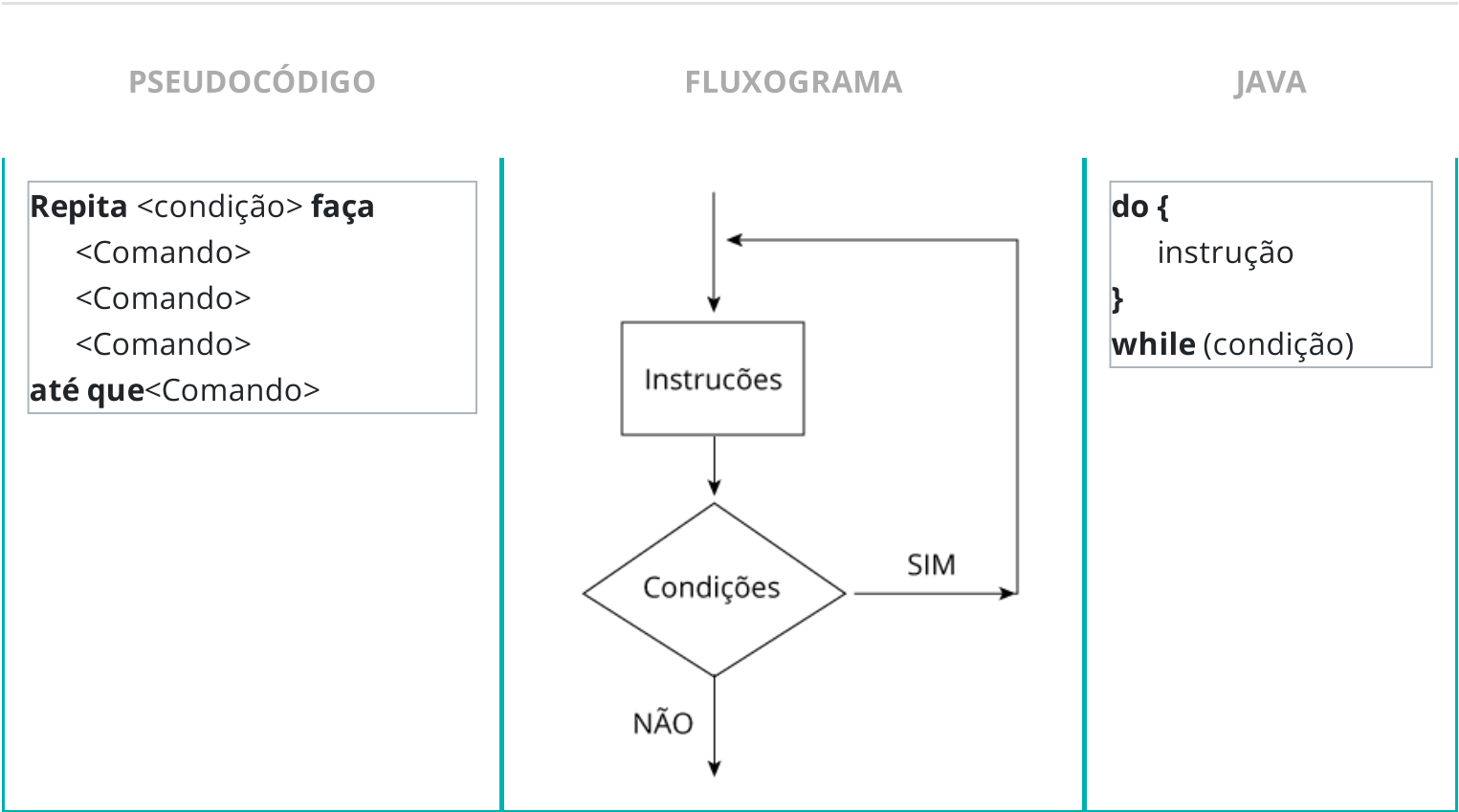
Note que o programa é idêntico ao anterior com a exceção de que não temos o incremento do contador **num** (num <- num+1). Com isso, o contador fica eternamente com o valor de 0 (zero) e o programa nunca terá um fim.

Resultado:



Laço de Repetição “repita... até que”

Como você já sabe, esta estrutura é parecida com a while, que você acabou de conhecer. A única diferença é que a condição dela é executada por último. Vamos aos detalhes para entender melhor?



Agora que você conheceu um pouco mais esta estrutura, vamos voltar ao exemplo anterior.

← Voltar



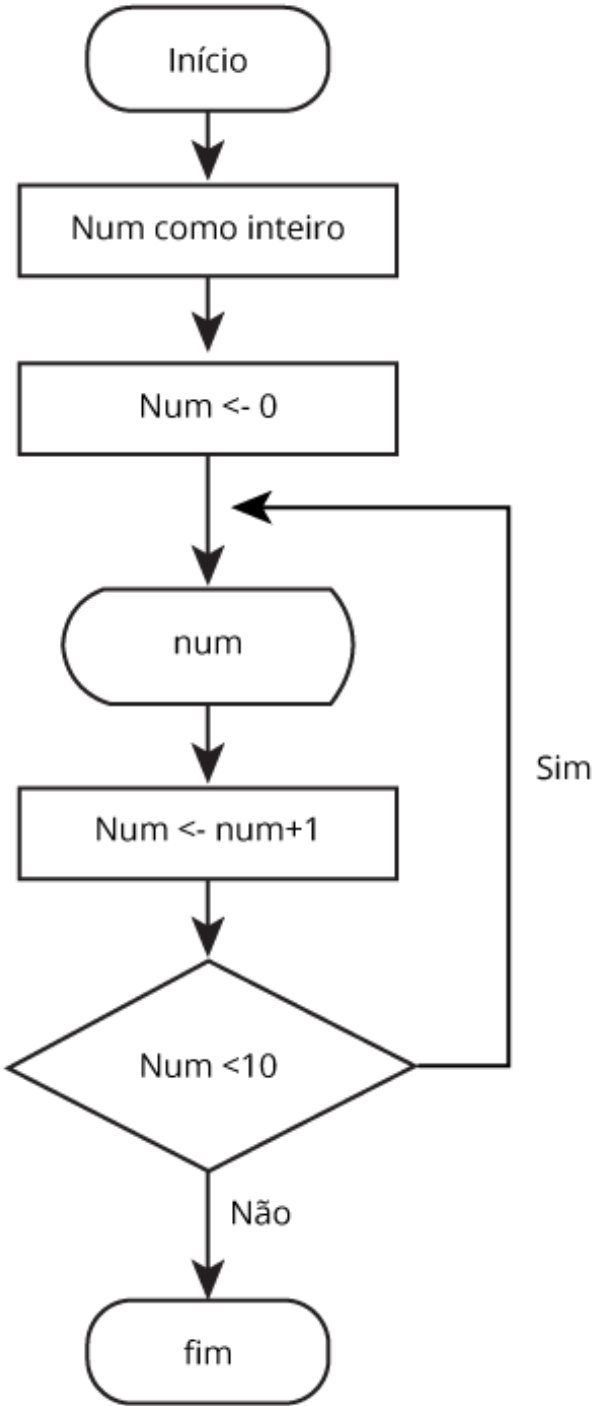
Elabore um Algoritmo, um Fluxograma e um Programa em Java que mostre todos os números menores que 10.

PSEUDOCÓDIGO

```
Programa ex01

Declare
  num como inteiro
Início
  repita
    escreva num
    num <- num + 1
  até que (num < 10)
fim.
```

FLUXOGRAMA



Agora, veja o programa em Java:

```
1
2 public class ex01 {
3
4     public static void main(String[] args) {
5         int num =0;
6         do{
7             System.out.println("Número " + num);
8             num++;
9         }
10        while (num < 10);
11    }
12}
```

Copiar

Observe que, como o programa responde ao mesmo exercício, e o resultado é o mesmo. Porém, o programa foi construído utilizando outro Laço de repetição. Note que as linhas 6 até 10 apresentam a sintaxe do comando **repita até**.

Na videoaula a seguir, o professor Sandro Valérios retoma os conteúdos estudados até o momento.



Utilizando comandos de repetição em conjunto com comando condicional if

Nas unidades anteriores você viu que é possível utilizar o comando **para** (for) em conjunto com o **se** (if) para atingir a solução de um problema na programação. Assim, como no caso do **para**, também podemos usar o **enquanto** (while) e o **repita até** (do while) em conjunto com o **se** (if) com o mesmo propósito.

Veja o exemplo a seguir, um programa de uma empresa financeira. Nele, o usuário informa seu cargo e salário e, a partir dessas informações, o programa informa quanto de empréstimo a pessoa pode obter.

Diretor = 30%

Gerente = 25%

Operacional = 20%

```
1  import java.util.Scanner;
2  public class ex03 {
3
4      public static void main(String[]args) {
5
6          String nome, cargo;
7          float sal, emprestimo=0;
8
9          System.out.println("Por favor informe seu nome");
10         nome = new Scanner(System.in).nextLine();
11         System.out.println("E agora informe seu cargo (Diretor, Gerente ou
Operacional)");
12
13
14         cargo = new Scanner(System.in).nextLine();
15
16         while (cargo == "Diretor" || cargo == "Gerente" || cargo == "Operacional") {
17             System.out.println("Cargo incorreto, por favor informe o cargo
novamente");
18
19             cargo = new scanner(System.in).nextLine();
20         }
21
22         System.out.println("Agora informe o seu salário");
23         sal = new scanner(System.in).nextFloat();
24
25         if (cargo.equals("Gerente"))
26             emprestimo = sal*25/100;
27         else if (cargo.equals("Diretor"))
28             emprestimo = sal*30/100;
29         else
30             emprestimo = sal*20/100;
31
32         System.out.println("Olá" + nome);
33         System.out.println("Seu cargo é" + cargo);
34         System.out.println("Seu salário é" + sal);
35         System.out.println("Olá" + nome);
36         System.out.println("E você tem direito a pegar R$" + emprestimo + "de
empréstimo");
    }
}
```

[Copiar](#)


Você notou que além dos usos do Laço de repetição **while** (linhas 15 até 18), da estrutura de seleção if (linhas 23 até 28), foi utilizado, também, o **scanner** (linhas 1, 10, 13, 17 e 21)? Pois é, neste momento do curso você já tem diversos comandos em seu repertório, e deve utilizá-los conjuntamente para solucionar os desafios propostos.

Para finalizar, assista à videoaula do professor Rogério Silva, que sintetiza o conteúdo da unidade.

← Voltar

DS - Estruturas de Repetição - Comando while / do ... while



Recurso Educacional Aberto (REA) desenvolvido pela [Univesp](#), disponível sob licença [MIT](#).
Você pode utilizá-lo, compartilhá-lo e modificá-lo. | [Créditos](#) | [GitHub](#) .

Avaliar este recurso

Reportar erro



Secretaria de
Desenvolvimento Econômico

Secretaria da
Educação