

Curso Técnico em Desenvolvimento de Sistemas Online

LÓGICA E TÉCNICAS DE PROGRAMAÇÃO EM JAVA

GEEaD - Grupo de Estudo de Educação a Distância

Centro de Educação Tecnológica Paula Souza

Expediente

GEEaD – CETEC
GOVERNO DO ESTADO DE SÃO PAULO
EIXO TECNOLÓGICO DE INFORMAÇÃO E COMUNICAÇÃO
CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS
FUNDAMENTOS DE INFORMÁTICA

Autores:

*Eliana Cristina Nogueira Barion
Marcelo Fernando Iguchi
Paulo Henrique Mendes Carvalho
Rute Akie Utida*

Revisão Técnica:

Lilian Aparecida Bertini

Revisão Gramatical:

Juçara Maria Montenegro Simonsen Santos

Editoração e Diagramação:

Flávio Biazim

São Paulo – SP, 2019

AGENDA 5

ESTRUTURAS DE DECISÃO II





MERGULHANDO NO TEMA...

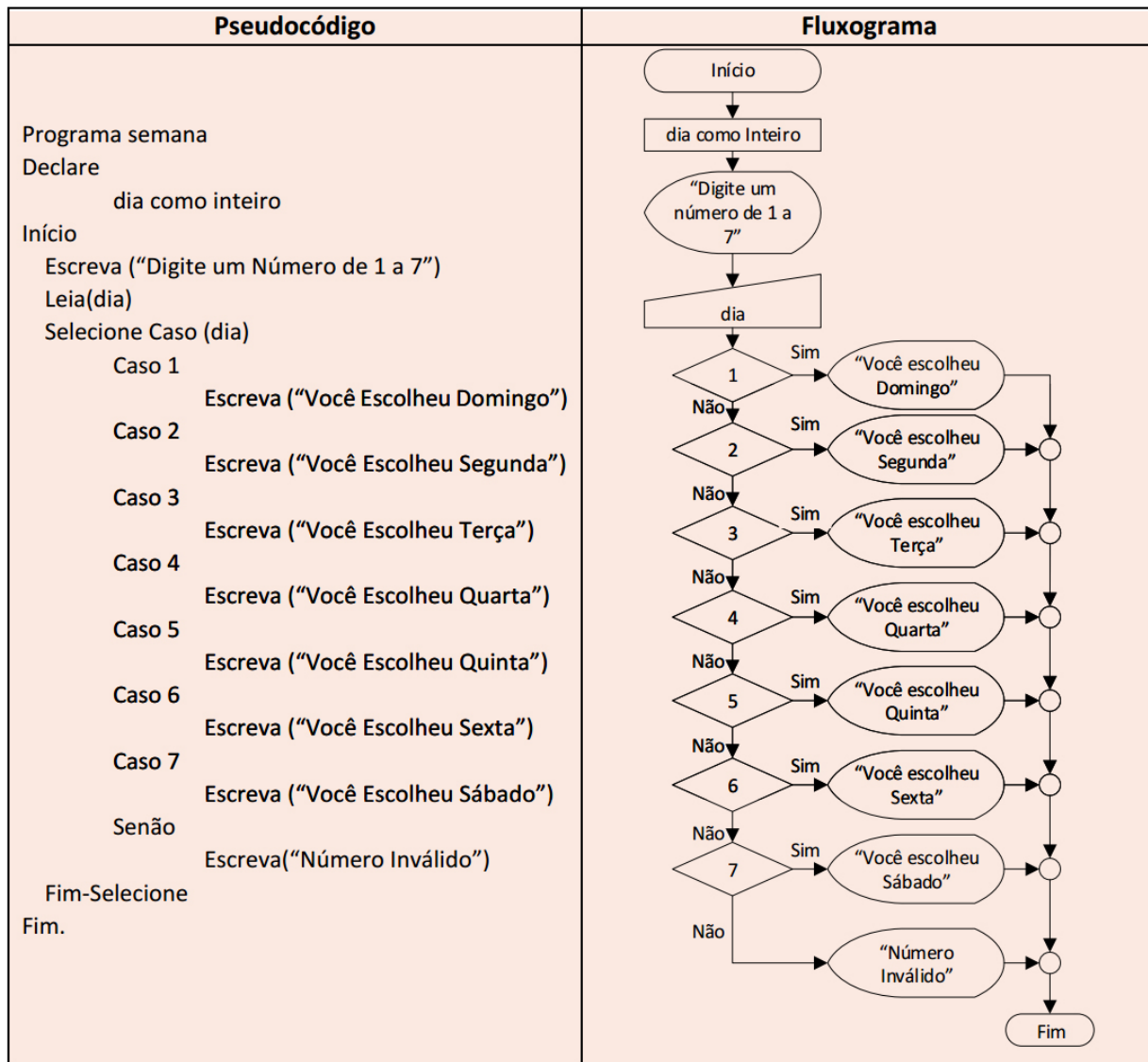
Se recapitularmos o último exemplo de Estrutura de decisão aninhada da agenda anterior, no qual o programa exibia o dia da semana de acordo com o número digitado pelo usuário, notamos que o código do programa fica relativamente confuso devido à grande quantidade de comandos de seleção (Se). A tendência dessa confusão é aumentar conforme o número de comandos de decisão aninhados for crescendo. Isso, é claro, levando em conta que estamos realizando a comparação sempre com a mesma variável.

Para estas situações podemos utilizar a estrutura “selecione caso...senão...fim_selecione” do pseudocódigo ou a estrutura switch-case do Java.

Pseudocódigo	Fluxograma	Java
Selecione Caso {variável} Caso condição 01 {comando(s)} Caso condição 02 {comando(s)} Caso condição 03 {comando(s)} ... Senão {comando(s)} Fim-Selezione	<pre> graph TD Start(()) --> Cond1{condição} Cond1 -- Sim --> Right1(()) Cond1 -- Não --> Cond2{condição} Cond2 -- Sim --> Right2(()) Cond2 -- Não --> End(()) </pre>	switch (variável){ case condição 01: {comando(s)}; break ; case condição 02: {comando(s)}; break ; case condição 03: {comando(s)}; break ; ... default : {comando(s)}; }

Essas estruturas permitem que a seleção correta seja feita a partir da comparação do valor do conteúdo da variável a uma lista definida durante a programação. Ao encontrar a correspondência correta, os respectivos comandos são executados e as demais opções ignoradas. Tanto as sintaxes em pseudocódigo quanto em Java são semelhantes como temos a seguir:

O Funcionamento é muito simples. Vamos ver um exemplo para entender o funcionamento.



Você acha que já viu o programa em algum lugar? Não é impressão, não! É exatamente o mesmo programa de exemplo de Estrutura de decisão aninhada da agenda anterior. Com essa nova estrutura que foi apresentada, o código foi escrito de maneira muito mais elegante. O fluxograma permaneceu o mesmo sem alteração nenhuma.

Vamos agora ao código em Java:

```
switchCase.java
1 import javax.swing.JOptionPane;
2
3 public class switchCase {
4
5     public static void main(String[] args) {
6         //declaração de variáveis
7         int dia; // variável para armazenamento da semana
8
9         //entrada de dados com conversão de tipos juntas
10        dia = Integer.parseInt(JOptionPane.showInputDialog("Entre com um número de 1 a 7"));
11
12        switch (dia) {
13
14            case 1:
15                JOptionPane.showMessageDialog(null, "Você escolheu Domingo");
16                break;
17            case 2:
18                JOptionPane.showMessageDialog(null, "Você escolheu Segunda");
19                break;
20            case 3:
21                JOptionPane.showMessageDialog(null, "Você escolheu Terça");
22                break;
23            case 4:
24                JOptionPane.showMessageDialog(null, "Você escolheu Quarta");
25                break;
26            case 5:
27                JOptionPane.showMessageDialog(null, "Você escolheu Quinta");
28                break;
29            case 6:
30                JOptionPane.showMessageDialog(null, "Você escolheu Sexta");
31                break;
32            case 7:
33                JOptionPane.showMessageDialog(null, "Você escolheu Sábado");
34                break;
35
36            default:
37                JOptionPane.showMessageDialog(null, "Número Inválido");
38                break;
39        }
40    } // fim do método main
41
42 } // fim da classe
43
```

Analogamente ao pseudocódigo, em Java, após fazermos a entrada do dado pelo usuário, o comando switch (dia) irá comparar o valor armazenado na variável dia com um valor de 1 a 7 em cada comando case.

Exemplo:

se o usuário entrar com o valor 3, teremos dia = 3 e ao executarmos o comando case 3: (dia=3), como o resultado da comparação será verdadeiro, ele executará o comando JOptionPane.showMessageDialog(null, "Você escolheu Terça"); e o comando break; e por fim finalizará o programa.

O comando break tem a finalidade de parar a execução do comando switch, uma vez que já foi executada a ação necessária (comparação verdadeira) e não há a necessidade de se continuar com o comando switch.

Caso nenhuma comparação resulte em uma resposta verdadeira, o comando default: é executado gerando uma mensagem de "número inválido" para o usuário.

Programa de seleção de cores:

Programa cor

Declare

cor como caractere

Início

Escreva ("Digite o nome de uma cor")

Leia(cor)

Selecione Caso (cor)

Caso "Azul"

Escreva ("Você Escolheu a cor Azul")

Caso "Vermelho"

Escreva ("Você Escolheu a cor Vermelha")

Caso "Branco"

Escreva ("Você Escolheu a cor Branca")

Caso "Preto"

Escreva ("Você Escolheu a cor Preta")

Caso "Verde"

Escreva ("Você Escolheu a cor Verde")

Caso "Amarelo"

Escreva ("Você Escolheu a cor Amarela")

Senão

Escreva ("Cor Inválida")

Fim-Seleção

Fim.



VOCÊ NO COMANDO

Exercícios sobre estrutura de seleção:

Os exercícios de 1 a 4 devem ser desenvolvidos elaborando o pseudocódigo, o fluxograma e a linguagem Java:

1. Vanessa precisa desenvolver um programa em que ao se digitar o código de um produto cadastrado da papelaria Lápis Colorido este retorne o nome do produto. Se o código do produto não estiver cadastrado, o programa deve exibir a mensagem produto não cadastrado. A tabela a seguir descreve os códigos e os produtos cadastrados:

Como Vanessa resolveria essa questão?

Código	Produto
100	Lápis preto N.2
150	Borracha branca
200	Caneta Azul
230	Caneta Vermelha
256	Giz de Cera 12un.
300	Cartolina Branca
310	Resma de Sulfite branco A4
400	Estojo Escolar Verde
470	Caderno universitário 100fls.
500	Caderno brochura 50 fls.

2. Eliberto deseja fazer um menu de seleção de um programa utilizando o comando de seleção múltipla. O menu consiste nas opções de:

1. Cadastrar usuário
2. Alterar dados
3. Excluir Usuário

Como Eliberto faria esse programa? Elabore uma mensagem diferente para ser exibida para o usuário ao selecionar cada um dos itens dos menus.

3. Além do menu apresentado na questão anterior, Eliberto precisa fazer alguns submenus com a seguinte estrutura:

1. Cadastrar usuário
2. Alterar dados
 1. Alterar nome
 2. Alterar Endereço
 3. Alterar telefone
 4. Alterar RG.
3. Excluir usuário

Altere o programa do exercício anterior para incluir este submenu e exiba uma mensagem correspondente de acordo com o menu escolhido pelo usuário.

4. Crie um programa para uma calculadora com cinco operações básicas (soma, subtração, multiplicação, divisão e resto da divisão) em que dois números reais devem ser inseridos e, posteriormente, em um menu, o usuário escolha qual operação deve ser realizada. Após a execução da operação selecionada o resultado deve ser apresentado na tela do computador.

Exercícios para serem resolvidos somente em linguagem Java (5 e 6):

5. Elabore um programa em que o usuário deva inserir um número do tipo double e a seguir o programa exiba este número.

6. Denis decidiu, por curiosidade, elaborar um programa que realiza a conversão de milhas para quilômetros. Sabe-se que uma milha tem 1,609km. Como Denis resolveria este problema?

Respostas:

Comando de Seleção de Múltipla

1.

Pseudocódigo	Fluxograma
<p>Programa ex1</p> <p>Declare</p> <p>opcao como inteiro</p> <p>Início</p> <p>Escreva ("Entre com o código do produto")</p> <p>Leia (opcao)</p> <p>Selecione Caso (opcao)</p> <p>Caso condição 100</p> <p>Escreva ("Lápis preto nº2")</p> <p>Caso condição 150</p> <p>Escreva ("borracha branca")</p> <p>Caso condição 200</p> <p>Escreva ("caneta azul")</p> <p>Caso condição 230</p> <p>Escreva ("Caneta vermelha")</p> <p>Caso condição 256</p> <p>Escreva ("giz de cera 12 und.")</p> <p>Caso condição 300</p> <p>Escreva ("cartolina branca")</p> <p>Caso condição 310</p> <p>Escreva ("resma de sulfite A4")</p> <p>Caso condição 400</p> <p>Escreva ("estojo escolar verde")</p> <p>Caso condição 470</p> <p>Escreva ("caderno universitário 100fls.")</p> <p>Caso condição 500</p> <p>Escreva ("caderno brochura 50fls.")</p> <p>Senão</p> <p>Escreva ("Produto não cadastrado")</p> <p>Fim-selecione</p> <p>Fim.</p>	<pre> graph TD Inicio([Início]) --> opcao_int[opcao como inteiro] opcao_int --> entrada[/Entre com o código do Produto/] entrada --> opcao{opcao} opcao -- Sim --> d1{100} opcao -- Não --> d1 d1 -- Sim --> o1([Lapis preto nº2]) d1 -- Não --> d2{150} d2 -- Sim --> o2([borracha branca]) d2 -- Não --> d3{200} d3 -- Sim --> o3([caneta azul]) d3 -- Não --> d4{230} d4 -- Sim --> o4([Caneta vermelha]) d4 -- Não --> d5{256} d5 -- Sim --> o5([giz de cera 12 und.]) d5 -- Não --> d6{300} d6 -- Sim --> o6([cartolina branca]) d6 -- Não --> d7{310} d7 -- Sim --> o7([resma de sulfite A4]) d7 -- Não --> d8{400} d8 -- Sim --> o8([estojo escolar verde]) d8 -- Não --> d9{470} d9 -- Sim --> o9([caderno universitário 100fls.]) d9 -- Não --> d10{500} d10 -- Sim --> o10([caderno brochura 50fls.]) d10 -- Não --> o11([Produto não cadastrado]) o1 --> join1(()) o2 --> join1 o3 --> join2(()) o4 --> join2 o5 --> join3(()) o6 --> join3 o7 --> join4(()) o8 --> join4 o9 --> join5(()) o10 --> join5 o11 --> join6(()) join1 --> join2 join2 --> join3 join3 --> join4 join4 --> join5 join5 --> join6 join6 --> Fim([Fim]) </pre>

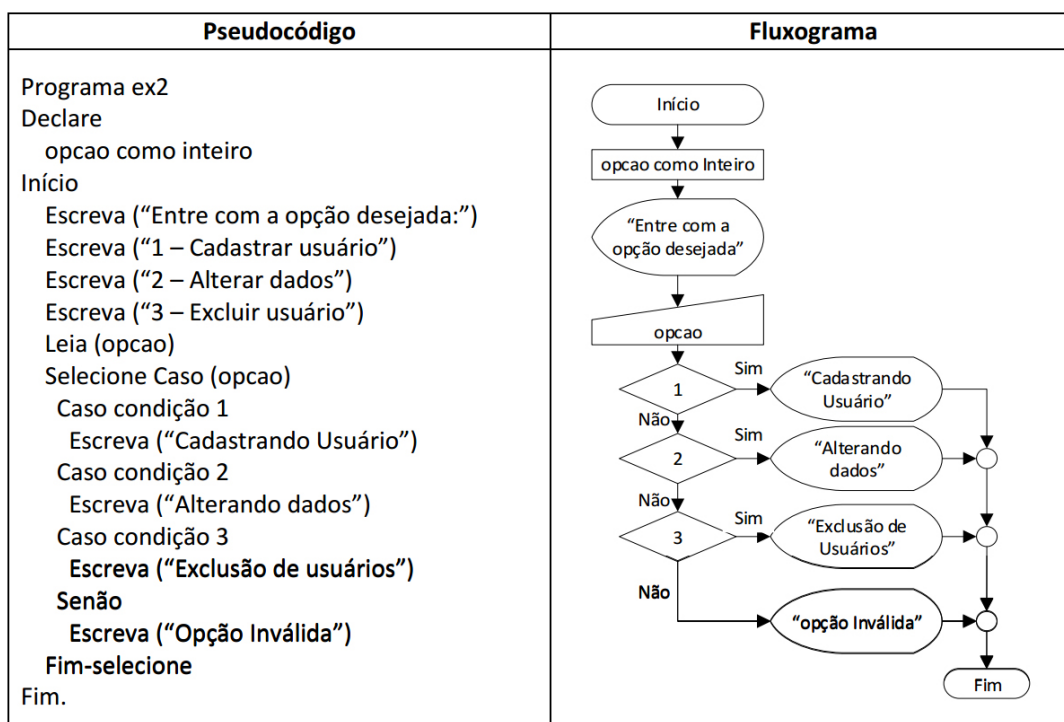
Em Java

```

Switch_ex1.java
1 import javax.swing.JOptionPane;
2
3 public class Switch_ex1 {
4
5     public static void main(String[] args) {
6         // Exercício 1
7
8         //declaração de variáveis
9         int opcao;
10
11         //entrada de dados
12         opcao = Integer.parseInt(JOptionPane.showInputDialog("Entre com o código "
13             + "do produto"));
14
15         //processamento e saída
16         switch (opcao) {
17             case 100:
18                 JOptionPane.showMessageDialog(null, "O produto selecionado é Lapis preto n.2");
19                 break;
20
21             case 150:
22                 JOptionPane.showMessageDialog(null, "O produto selecionado é Borracha branca");
23                 break;
24
25             case 200:
26                 JOptionPane.showMessageDialog(null, "O produto selecionado é Caneta Azul");
27                 break;
28
29             case 230:
30                 JOptionPane.showMessageDialog(null, "O produto selecionado é Caneta Vermelha");
31                 break;
32
33             case 256:
34                 JOptionPane.showMessageDialog(null, "O produto selecionado é Giz de cera 12und.");
35                 break;
36
37             case 300:
38                 JOptionPane.showMessageDialog(null, "O produto selecionado é Cartolina Branca");
39                 break;
40
41             case 310:
42                 JOptionPane.showMessageDialog(null, "O produto selecionado é Resma de "
43                     + "sulfite branco A4");
44                 break;
45
46             case 400:
47                 JOptionPane.showMessageDialog(null, "O produto selecionado é Estojo "
48                     + "escolar verde");
49                 break;
50             ---

```

2.



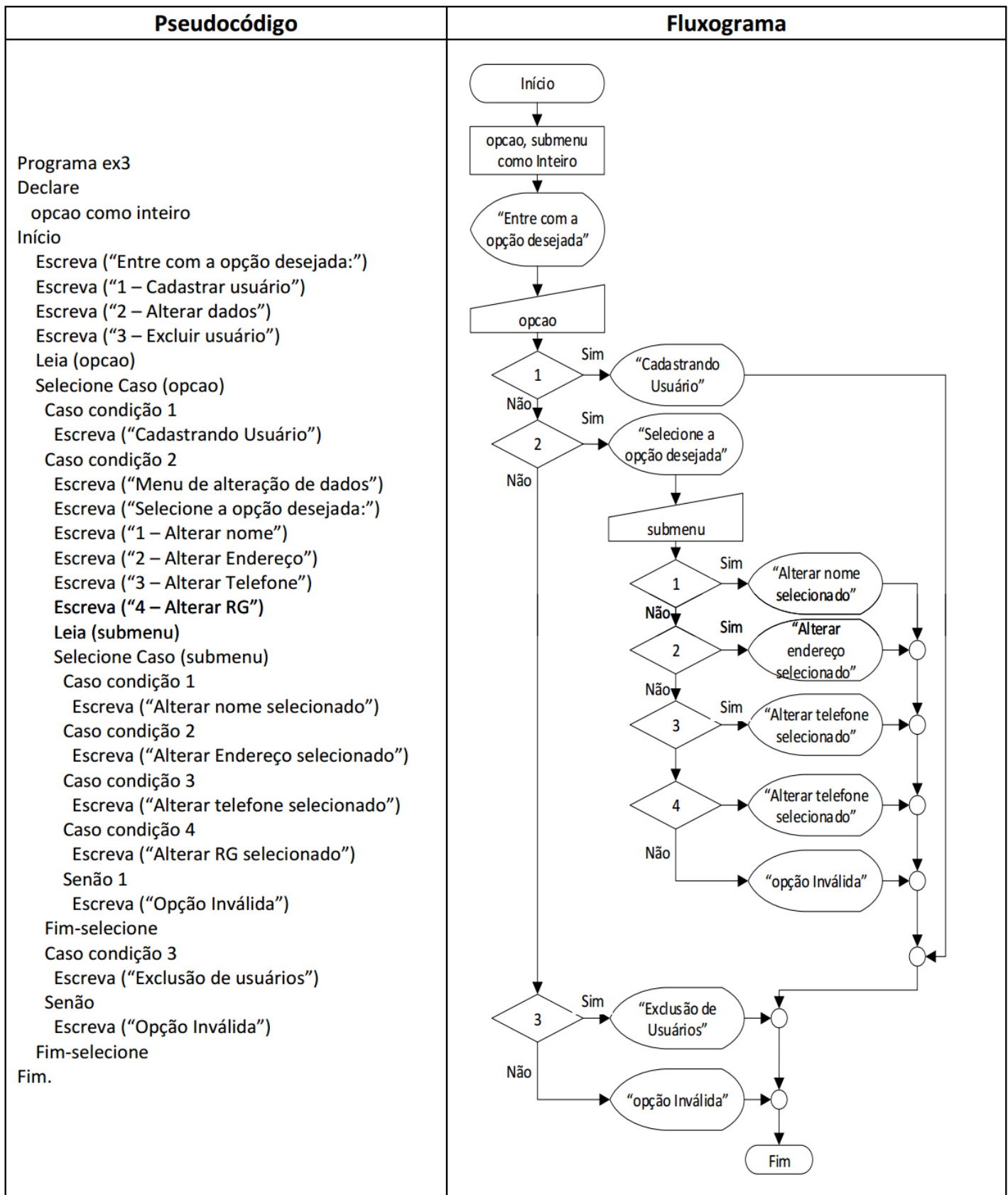
Em Java

```

1  import javax.swing.JOptionPane;
2
3  public class Switch_ex2 {
4
5      public static void main(String[] args) {
6          // Exercício 2
7
8          //declaração de variáveis
9          int opcao;
10
11         //entrada de dados
12         opcao = Integer.parseInt(JOptionPane.showInputDialog("Entre com a opção desejada:" +
13             "\n1 - Cadastrar usuário" +
14             "\n2 - Alterar dados"+
15             "\n3 - Excluir usuário"));
16
17
18         //processamento e saída
19         switch (opcao) {
20             case 1:
21                 JOptionPane.showMessageDialog(null, "Cadastrando usuário");
22                 break;
23
24             case 2:
25                 JOptionPane.showMessageDialog(null, "Alterando dados");
26                 break;
27
28             case 3:
29                 JOptionPane.showMessageDialog(null, "Exclusão de usuários");
30                 break;
31
32             default:
33                 JOptionPane.showMessageDialog(null, "Opção Inválida");
34                 break;
35         } //fim do switch-case
36
37     } // fim do método main
38
39 } // fim da classe
40

```

3.



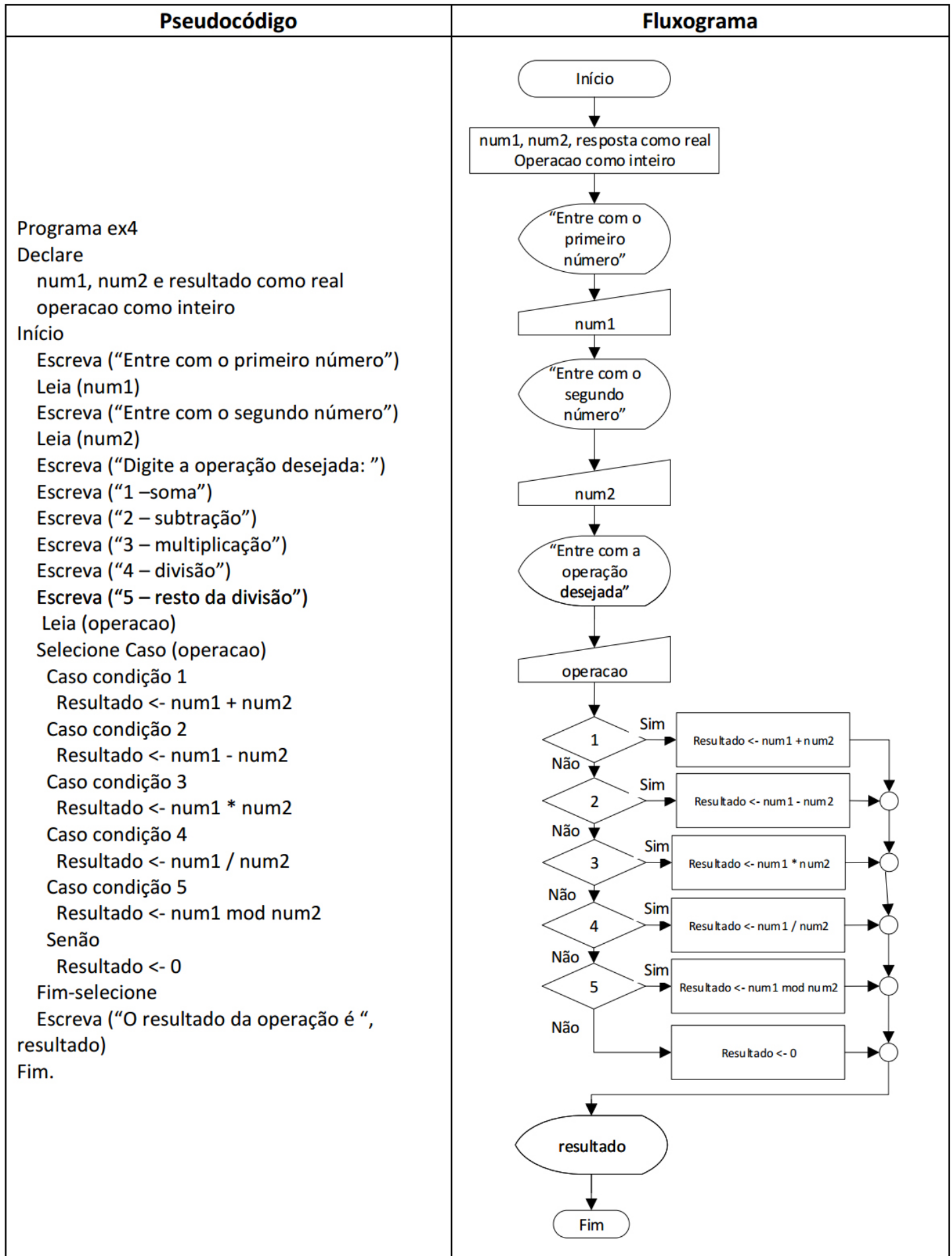
Em Java

```

Switch_ex3.java
1  import javax.swing.JOptionPane;
2
3  public class Switch_ex3 {
4
5      public static void main(String[] args) {
6          // Exercício 3
7
8          // declaração de variáveis
9          int opcao, submenu;
10
11         // entrada de dados
12         opcao = Integer.parseInt(JOptionPane.showInputDialog("Entre com a opção desejada:" +
13             "\n1 - Cadastrar usuário" +
14             "\n2 - Alterar dados" +
15             "\n3 - Excluir usuário"));
16
17         // processamento e saída
18         switch (opcao) {
19             case 1:
20                 JOptionPane.showMessageDialog(null, "Cadastrando usuário");
21                 break;
22
23             case 2:
24                 submenu = Integer.parseInt(JOptionPane.showInputDialog(
25                     "Menu de alteração de dados" +
26                     "\nSelecione a opção desejada:" +
27                     "\n1 - Alterar nome" +
28                     "\n2 - Alterar endereço" +
29                     "\n3 - Alterar Telefone" +
30                     "\n4 - Alterar RG"));
31                 // início do 2º switch
32                 switch (submenu) {
33                     case 1:
34                         JOptionPane.showMessageDialog(null, "Alterar nome selecionado");
35                         break;
36                     case 2:
37                         JOptionPane.showMessageDialog(null, "Alterar endereço selecionado");
38                         break;
39                     case 3:
40                         JOptionPane.showMessageDialog(null, "Alterar telefone selecionado");
41                         break;
42                     case 4:
43                         JOptionPane.showMessageDialog(null, "Alterar RG selecionado");
44                         break;
45                     default:
46                         JOptionPane.showMessageDialog(null, "Opção inválida");
47                         break;
48                 } // fim do 2º switch
49                 break;
50

```

4.



Em Java

```

1  import javax.swing.JOptionPane;
2
3  public class Switch_ex4 {
4
5      public static void main(String[] args) {
6          // Exercício 4
7
8          //declaração de variáveis
9          double num1, num2, resultado;
10         int operacao;
11
12         //entrada de dados
13         num1 = Double.parseDouble(JOptionPane.showInputDialog("Entre com o primeiro número"));
14         num2 = Double.parseDouble(JOptionPane.showInputDialog("Entre com o segundo número"));
15
16         operacao = Integer.parseInt(JOptionPane.showInputDialog(
17             "Digite a operação desejada: " +
18             "\n1 - Soma" +
19             "\n2 - Subtração" +
20             "\n3 - Multiplicação" +
21             "\n4 - Divisão" +
22             "\n5 - Resto da divisão"));
23
24         //processamento
25         switch (operacao) {
26             case 1: //soma
27                 resultado = num1 + num2;
28                 break;
29
30             case 2: //subtração
31                 resultado = num1 - num2;
32                 break;
33
34             case 3: //multiplicação
35                 resultado = num1 * num2;
36                 break;
37
38             case 4: //divisão
39                 resultado = num1 / num2;
40                 break;
41
42             case 5: //resto da divisão
43                 resultado = num1 % num2;
44                 break;
45
46             default:
47                 resultado = 0;
48                 break;
49         } //fim do switch-case
50
51         //saída de dados
52         JOptionPane.showMessageDialog(null, "O resultado da operação é " + resultado);
53     } // fim do método main
54
55 } // fim da classe
56

```

Observação: resolução dos exercícios somente em Java.

5.

```
TryCatch_Ex1.java
1 import javax.swing.JOptionPane;
2
3 public class TryCatch_Ex1 {
4
5     public static void main(String[] args) {
6         // Exercício 1
7
8         // declaração de variáveis
9         double num;
10
11        // entrada de dados
12        num = Double.parseDouble(JOptionPane.showInputDialog("Entre com um número"));
13        // saída de dados
14        JOptionPane.showMessageDialog(null, "O número digitado é " + num);
15    }
16 }
17
18 }
```

6.

```
*TryCatch_Ex3.java
1 import javax.swing.JOptionPane;
2
3 public class TryCatch_Ex3 {
4
5     public static void main(String[] args) {
6         // Exercício 3
7
8         // declaração de variáveis
9         double km, milha;
10
11        // entrada de dados
12        milha = Double.parseDouble(JOptionPane.showInputDialog("Entre com um número" +
13            " em milhas"));
14
15        // processamento
16        km = milha * 1.609;
17
18        // saída de dados
19        JOptionPane.showMessageDialog(null,
20            "O valor de " + milha + " milhas convertido para quilômetros é " + km +
21            " quilômetros");
22    }
23 }
24
25 }
```