



Serviço Nacional de Aprendizagem Industrial
PELO FUTURO DO TRABALHO

SÉRIE TI - SOFTWARE

LÓGICA DE PROGRAMAÇÃO

EXPRESSÕES LÓGICAS,
RELACIONAIS E ARITMÉTICAS



CONFEDERAÇÃO NACIONAL DA INDÚSTRIA – CNI

Robson Braga de Andrade
Presidente

GABINETE DA PRESIDÊNCIA

Teodomiro Braga da Silva
Chefe do Gabinete - Diretor

DIRETORIA DE EDUCAÇÃO E TECNOLOGIA - DIRET

Rafael Esmeraldo Lucchesi Ramacciotti
Diretor de Educação e Tecnologia

SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL - SENAI

Robson Braga de Andrade
Presidente do Conselho Nacional

SENAI – Departamento Nacional

Rafael Esmeraldo Lucchesi Ramacciotti
Diretor-Geral

Julio Sergio de Maya Pedrosa Moreira
Diretor-Adjunto

Gustavo Leal Sales Filho
Diretor de Operações



Serviço Nacional de Aprendizagem Industrial

PELO FUTURO DO TRABALHO

SÉRIE TI - SOFTWARE

LÓGICA DE PROGRAMAÇÃO

**EXPRESSÕES LÓGICAS,
RELACIONAIS E ARITMÉTICAS**



© 2020. SENAI – Departamento Nacional

© 2020. SENAI – Departamento Regional de Santa Catarina

A reprodução total ou parcial desta publicação por quaisquer meios, seja eletrônico, mecânico, fotocópia, de gravação ou outros, somente será permitida com prévia autorização, por escrito, do SENAI.

Esta publicação foi elaborada pela equipe de Educação a Distância do SENAI de Santa Catarina, com a coordenação do SENAI Departamento Nacional, para ser utilizada por todos os Departamentos Regionais do SENAI nos cursos presenciais e a distância.

SENAI Departamento Nacional

Unidade de Educação Profissional e Tecnológica - UNIEP

SENAI Departamento Regional de Santa Catarina

Gerência de Educação

SENAI

Serviço Nacional de
Aprendizagem Industrial
Departamento Nacional

Sede

Setor Bancário Norte • Quadra 1 • Bloco C • Edifício Roberto
Simonsen • 70040-903 • Brasília – DF • Tel.: (0xx61) 3317-
9001 Fax: (0xx61) 3317-9190 • <http://www.senai.br>

Lista de Ilustrações

Figura 1 - Tela do VisualG – Exemplo de Execução com Operadores Aritméticos	12
Figura 2 - Tela da IDE Eclipse – Exemplo de Execução com Operadores Aritméticos.....	14
Figura 3 - Tela do VisualG – Exemplo de Execução com Operadores Relacionais.....	16
Figura 4 - Tela da IDE Eclipse – Exemplo de Execução com Operadores Relacionais	18
Figura 5 - Tela do VisualG – Exemplo de Operador Lógico E em Portugol.....	20
Figura 6 - Tela da IDE Eclipse – Exemplo de Operador Lógico OU em Java.....	21
Figura 7 - Protótipo do Monitor dos Canos.....	22
Figura 8 - Tela do VisualG – Proposta Monitor Líquidos com Operadores.....	22
Figura 9 - Tela da IDE Eclipse – Proposta Monitor Líquidos com Operadores	23
Figura 10 - Tela do VisualG – Calcular IMC em Portugol.....	24
Figura 11 - Tela da IDE Eclipse – Calcular IMC em Java	24
Tabela 1 - Principais operadores aritméticos em Portugol e Java	10
Tabela 2 - Principais operadores relacionais em Portugol e Java	10
Tabela 3 - Principais operadores lógicos em Portugol e Java.....	11
Tabela 4 - Interpretação das linhas de código com operadores aritméticos em Portugol.....	13
Tabela 5 - Interpretação das linhas de código com operadores aritméticos em Java	15
Tabela 6 - Interpretação das linhas de código com operadores relacionais em Portugol	17
Tabela 7 - Interpretação das linhas de código com operadores relacionais em Java.....	19
Tabela 8 - Explicação do Operador Lógico E.....	20
Tabela 9 - Explicação do Operador Lógico E com alteração do valor de Idade	20
Tabela 10 - Explicação do Operador Lógico OU	21

Sumário

Editor de Planilhas Eletrônicas	9
Apresentação	9
Definição	10
Estrutura	10
Aplicação na indústria	32
Exemplos	32
Linha do tempo.....	32
Palavra do docente	35
Referências.....	37

Expressões Lógicas, Relacionais e Aritméticas



APRESENTAÇÃO

Olá! Agora aqui você vai poder explorar questões relacionadas as expressões lógicas, relacionais e aritméticas.

As expressões lógicas (com seus operadores lógicos e operadores relacionais) e as expressões aritméticas (com todos os seus conjuntos de operadores, tais como adição, subtração, multiplicação e divisão) fornecem uma experiência muito interessante para os programas, complementando os estudos sobre como declarar as variáveis, definir seus tipos e até atribuir alguns valores. Aqui você vai ter a possibilidade de adicionar comportamentos mais complexos para esses programas. Com a apresentação das expressões lógicas e aritméticas, você será capaz de resolver questões mais elaboradas em programas.

Bons estudos!

DEFINIÇÃO

Primeiramente, é preciso definir bem o que são expressões lógicas e o que são expressões aritméticas. Destaca-se que as expressões aritméticas fazem parte do cotidiano formativo das escolas, principalmente na disciplina de matemática.

As expressões aritméticas são as operações aritméticas fundamentais que são utilizados nos cálculos básicos de matemática, tais como: adição, subtração, multiplicação, divisão e resto.

Confira, na tabela, a seguir, como são identificados os operadores aritméticos tanto na linguagem Portugol como no Java.

OPERADOR ARITMÉTICO	PORTUGOL	JAVA
adição	+	+
subtração	-	-
multiplicação	*	*
divisão	/	/
resto	%	%

Tabela 1 - Principais operadores aritméticos em Portugol e Java
Fonte: do Autor (2020)

Fica fácil perceber que existe muita semelhança no que se refere aos operadores aritméticos dessas duas linguagens, principalmente nos operadores aritméticos mais básicos, justamente pela simplicidade de suas aplicações e padronizações na maioria das linguagens de programação.

E o que são operadores relacionais? São os operadores utilizados para promover a relação e comparação entre variáveis e/ou valores de um mesmo tipo e para promover como saída uma relação de comparação verdadeira ou falsa. Assim como os operadores aritméticos, os operadores relacionais, na aplicação da matemática, são empregados na resolução de diversas comparações de determinados valores.

A seguir, serão apresentados os operadores relacionais básicos utilizados na lógica de programação. Veja, na tabela, quais são e suas respectivas representações nas linguagens Portugol e Java.

OPERADOR RELACIONAL	PORTUGOL	JAVA
Igualdade	=	==
Maior que	>	>
Menor que	<	<
Diferente de	<>	!=
Maior ou Igual a	>=	>=
Menor ou Igual a	<=	<=

Tabela 2 - Principais operadores relacionais em Portugol e Java
Fonte: do Autor (2020)

Novamente é perceptível o quão são próximas as definições básicas dos operadores relacionais nas duas linguagens.

Por fim, ainda nas expressões lógicas, além dos operadores relacionais, há também os operadores lógicos, que são as estruturas utilizadas para encadear diversos valores lógicos ou expressões relacionais para, no fim, gerar um resultado verdadeiro ou falso. Então, pode-se considerar que os operadores lógicos servem para combinar resultados de expressões que geram determinados valores lógicos.

OPERADOR LÓGICO	PORTUGOL	JAVA
E	E	&&
OU	OU	
NÃO	NAO	!

Tabela 3 - Principais operadores lógicos em Portugol e Java
Fonte: do Autor (2020)

Já no caso dos operadores lógicos, há uma grande diferença na forma de declarar esses operadores. Porém, o resultado obtido é idêntico em todas as linguagens, pois o que se promoverá, nas maiorias das linguagens, é o comportamento de conectar expressões lógicas.

ESTRUTURA

Acompanhe agora como são declaradas as estruturas dessas expressões lógicas e de relacionamento, levando em consideração as particularidades das linguagens Portugol e do Java. Serão criadas, para ambos os casos, duas variáveis: X e Y, que serão do tipo numérico inteiro e mais uma variável R do tipo numérico real. Além disso, serão atribuídos depois os seguintes valores iniciais de X=3 e Y=2, sendo que a variável R será usada para guardar o resultado das operações.



```
1 Algoritmo "OperadoresAritmeticos"
2
3 Var
4   x : inteiro
5   y : inteiro
6   z : real
7 Inicio
8   //Atribuição de valor inicial
9   x <- 3
10  y <- 2
11
12  //Adição
13  z <- x + y
14  escreva(z)
15
16  //Subtração
17  z <- x - y
18  escreva(z)
19
20  //Multiplicação
21  z <- x * y
22  escreva(z)
23
24  //Divisão
25  z <- x / y
26  escreva(z)
27
28  //Resto
29  z <- x % y
30  escreva(z)
31
32 Fimalgoritmo
```

Unknown (20-?)

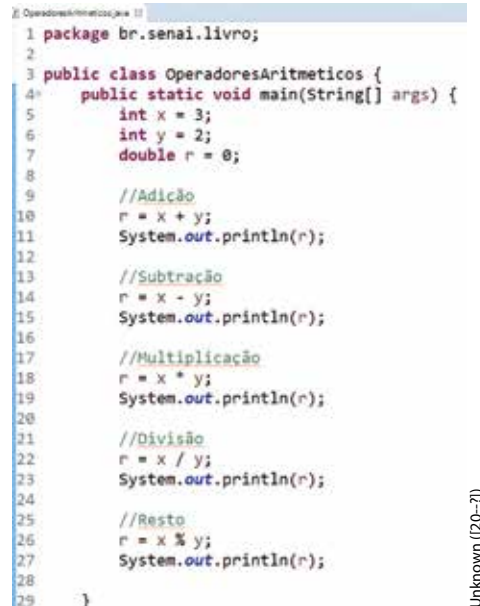
Figura 1 - Tela do VisualG – Exemplo de Execução com Operadores Aritméticos
Fonte: do Autor (2020)

Na figura, que trata da execução de operadores aritméticos no VisualG, é possível fazer a seguinte interpretação de algumas linhas de código (observe bem o número das linhas apresentadas na imagem anterior e seu resultado):

LINHA	RESULTADO NO PORTUGOL - VISUALG
4	Criada a variável x do tipo numérico inteiro.
5	Criada a variável y do tipo numérico inteiro.
6	Criada a variável r do tipo numérico real.
9	A variável x começa a valer o valor 3.
10	A variável y começa a valer o valor 2.
13	Acontece a operação de adição de x+y e atribui o resultado desta operação na variável r . Ou seja: $r \leftarrow x + y$ $r \leftarrow 3 + 2$ $r \leftarrow 5$
14	O sistema apresenta o valor da variável r naquele momento: 5.
17	Acontece a operação de subtração de x-y e atribui o resultado desta operação na variável r . Ou seja: $r \leftarrow x - y$ $r \leftarrow 3 - 2$ $r \leftarrow 1$
18	O sistema apresenta o valor da variável r naquele momento: 1.
21	Acontece a operação de multiplicação de x*y e atribui o resultado desta operação na variável r . Ou seja: $r \leftarrow x * y$ $r \leftarrow 3 * 2$ $r \leftarrow 6$
22	O sistema apresenta o valor da variável r naquele momento: 6.
25	Acontece a operação de divisão de x/y e atribui o resultado desta operação na variável r . Ou seja: $r \leftarrow x / y$ $r \leftarrow 3 / 2$ $r \leftarrow 1.5$
26	O sistema apresenta o valor da variável r naquele momento: 1.5.
29	Acontece a operação de resto de x%y e atribui o resultado desta operação na variável r . Ou seja: $r \leftarrow x \% y$ $r \leftarrow 3 \% 2$ $r \leftarrow 1$
30	O sistema apresenta o valor da variável r naquele momento: 1.

Tabela 4 - Interpretação das linhas de código com operadores aritméticos em Portugol
 Fonte: do Autor (2020)

Agora observe as mesmas operações aritméticas feitas em Java.

The image shows a screenshot of the Eclipse IDE with a Java file named 'OperadoresAritmeticos.java'. The code defines a package 'br.senai.livro', a public class 'OperadoresAritmeticos', and a static void main method. Inside the main method, variables x (int, 3), y (int, 2), and r (double, 0) are declared. The program then performs five arithmetic operations: addition (r = x + y), subtraction (r = x - y), multiplication (r = x * y), division (r = x / y), and modulo (r = x % y). Each operation is preceded by a comment in Portuguese: '//Adição', '//Subtração', '//Multiplicação', '//Divisão', and '//Resto'. The results are printed to the console using 'System.out.println(r);'. The code is enclosed in curly braces and line numbers 1 through 29 are visible on the left margin. On the right margin, there is a vertical text 'Unknown ((20--?))'.

```
1 package br.senai.livro;
2
3 public class OperadoresAritmeticos {
4     public static void main(String[] args) {
5         int x = 3;
6         int y = 2;
7         double r = 0;
8
9         //Adição
10        r = x + y;
11        System.out.println(r);
12
13        //Subtração
14        r = x - y;
15        System.out.println(r);
16
17        //Multiplicação
18        r = x * y;
19        System.out.println(r);
20
21        //Divisão
22        r = x / y;
23        System.out.println(r);
24
25        //Resto
26        r = x % y;
27        System.out.println(r);
28
29    }
```

Figura 2 - Tela da IDE Eclipse – Exemplo de Execução com Operadores Aritméticos
Fonte: do Autor (2020)

Mas, será que essas operações aritméticas executadas no Java terão os mesmos resultados que no Português? A resposta é não. Veja que, no caso da divisão, ocorre um resultado um pouco diferente.

LINHA	RESULTADO NO JAVA - IDE ECLIPSE
5	Criada a variável x do tipo numérico inteiro e atribuído o valor 3.
6	Criada a variável y do tipo numérico inteiro e atribuído o valor 2.
7	Criada a variável r do tipo numérico real e atribuído o valor 0.
10	Acontece a operação de adição de x+y e é atribuído o resultado desta operação na variável r . Ou seja: $r = x + y$ $r = 3 + 2$ $r = 5$
11	O sistema apresenta o valor da variável r naquele momento: 5.
14	Acontece a operação de subtração de x-y e é atribuído o resultado desta operação na variável r . Ou seja: $r = x - y$ $r = 3 - 2$ $r = 1$
15	O sistema apresenta o valor da variável r naquele momento: 1.
18	Acontece a operação de multiplicação de x*y e é atribuído o resultado desta operação na variável r . Ou seja: $r = x * y$ $r = 3 * 2$ $r = 6$
19	O sistema apresenta o valor da variável r naquele momento: 6
22	Acontece a operação de divisão de x/y e é atribuído o resultado desta operação na variável r . Ou seja: $r = x / y$ $r = 3 / 2$ $r = 1$
23	O sistema apresenta o valor da variável r naquele momento: 1.
26	Acontece a operação de resto de x%y e é atribuído o resultado desta operação na variável r . Ou seja: $r = x \% y$ $r = 3 \% 2$ $r = 1$
27	O sistema apresenta o valor da variável r naquele momento: 1.

Tabela 5 - Interpretação das linhas de código com operadores aritméticos em Java
 Fonte: do Autor (2020)

Você notou a pequena diferença? Neste caso, o Portugol e o Java se comportaram de forma diferenciada, por causa da estratégia com que lidam com os tipos de dados. Para o Portugol, na linha 25, a variável **r** assume o valor de 1,5, enquanto no Java na linha 22 a variável **r** assume o resultado de 1. Isso acontece porque no Java, e em boa parte das linguagens, uma operação entre valores do tipo inteiro sempre resultará em um valor também do tipo inteiro. Assim, será feita uma operação de divisão inteira entre o valor da variável **x**, que contém o valor 3, e a variável **y**, que contém o valor 2, resultando no valor 1. Entretanto, é possível “resolver” facilmente esse “problema” alterando o tipo da variável **x** de **int** para **double**. Assim ela teria a capacidade de receber valores decimais.

Agora, analise as estruturas dos operadores relacionais, a partir dos quais é possível comparar determinadas variáveis e verificar se tais comparações são verdadeiras ou falsas.



```

1 Algoritmo "OperadoresRelacionais01"
2
3 Var
4   x : inteiro
5   y : inteiro
6   z : logico
7 Inicio
8
9   //Atribuição de valor inicial
10  x <- 3
11  y <- 2
12
13  z <- (x > y)
14  escreva(z)
15
16  z <- (x < y)
17  escreva(z)
18
19  z <- (x = y)
20  escreva(z)
21
22  z <- (x >= y)
23  escreva(z)
24
25  z <- (x <= y)
26  escreva(z)
27
28  z <- (x <> y)
29  escreva(z)
30
31 Fimalgoritmo
  
```

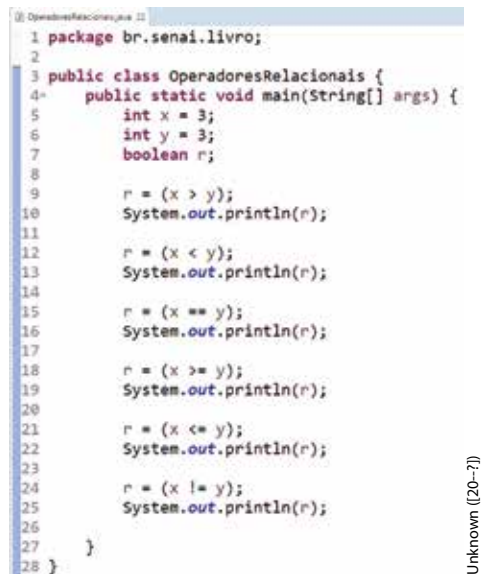
Figura 3 - Tela do VisualG – Exemplo de Execução com Operadores Relacionais
Fonte: do Autor (2020)

Verifique o que acontece em algumas linhas de comando na figura anterior, para compreender melhor o uso dos operadores relacionais.

LINHA	RESULTADO NO PORTUGOL - VISUALG
4	Criada a variável x do tipo numérico inteiro.
5	Criada a variável y do tipo numérico inteiro.
6	Criada a variável r do tipo lógico.
10	A variável x é inicializada com o valor 3.
11	A variável y é inicializada com o valor 2.
13	A variável lógica r receberá o resultado da operação relacional: se x for maior que y , recebe verdadeiro , caso contrário recebe falso .
11	Valor da variável r naquele momento: verdadeiro . Pois 3 é maior do que 2 .
16	A variável lógica r receberá o resultado da operação relacional: se x for menor que y , recebe verdadeiro , caso contrário recebe falso .
17	Valor da variável r naquele momento: falso . Pois 3 não é menor do que 2 .
19	A variável lógica r receberá o resultado da operação relacional: se x for igual ao valor de y , recebe verdadeiro , caso contrário recebe falso .
20	Valor da variável r naquele momento: falso . Pois 3 não é igual a 2 .
22	A variável lógica r receberá o resultado da operação relacional: se x for maior ou igual que y , recebe verdadeiro , caso contrário recebe falso .
23	Valor da variável r naquele momento: verdadeiro . Pois 3 é maior ou igual do que 2 .
25	A variável lógica r receberá o resultado da operação relacional: se x for menor ou igual que y , recebe verdadeiro , caso contrário recebe falso .
26	Valor da variável r naquele momento: falso . Pois 3 não é menor ou igual do que 2 .
28	A variável lógica r receberá o resultado da operação relacional: se x for diferente de y , recebe verdadeiro , caso contrário recebe falso .
29	Valor da variável r naquele momento: verdadeiro . Pois 3 é diferente do que 2 .

Tabela 6 - Interpretação das linhas de código com operadores relacionais em Portugol
Fonte: do Autor (2020)

Em Java, tem-se a seguinte forma de aplicar esses mesmos operadores relacionais, porém dessa vez será alterado o valor da variável **y**.



```
1 package br.senai.livro;
2
3 public class OperadoresRelacionais {
4     public static void main(String[] args) {
5         int x = 3;
6         int y = 3;
7         boolean r;
8
9         r = (x > y);
10        System.out.println(r);
11
12        r = (x < y);
13        System.out.println(r);
14
15        r = (x == y);
16        System.out.println(r);
17
18        r = (x >= y);
19        System.out.println(r);
20
21        r = (x <= y);
22        System.out.println(r);
23
24        r = (x != y);
25        System.out.println(r);
26    }
27 }
28 }
```

Unknown (20-?)

Figura 4 - Tela da IDE Eclipse – Exemplo de Execução com Operadores Relacionais
Fonte: do Autor (2020)

LINHA	RESULTADO NO JAVA - IDE ECLIPSE
5	Criada a variável x do tipo numérico inteiro e valor inicial 3.
6	Criada a variável y do tipo numérico inteiro e valor inicial 3.
7	Criada a variável r do tipo lógico.
9	A variável lógica r receberá o resultado da operação relacional: se x for maior que y , recebe verdadeiro , caso contrário recebe falso .
10	Valor da variável r naquele momento: falso . Pois 3 não é menor do que 3 .
12	A variável lógica r receberá o resultado da operação relacional: se x for menor que y , recebe verdadeiro , caso contrário recebe falso .
13	Valor da variável r naquele momento: falso . Pois 3 não é menor do que 3 .
15	A variável lógica r receberá o resultado da operação relacional: se x for igual ao valor de y , recebe verdadeiro , caso contrário recebe falso .
16	Valor da variável r naquele momento: verdadeiro . Pois 3 é igual a 3 .
18	A variável lógica r receberá o resultado da operação relacional: se x for maior ou igual que y , recebe verdadeiro , caso contrário recebe falso .
19	Valor da variável r naquele momento: verdadeiro . Pois 3 é maior ou igual do que 3 .
21	A variável lógica r receberá o resultado da operação relacional: se x for menor ou igual que y , recebe verdadeiro , caso contrário recebe falso .
22	Valor da variável r naquele momento: verdadeiro . Pois 3 é menor ou igual do que 3 .
24	A variável lógica r receberá o resultado da operação relacional: se x for diferente de y , recebe verdadeiro , caso contrário recebe falso .
25	Valor da variável r naquele momento: falso . Pois 3 é não diferente do que 3 .

Tabela 7 - Interpretação das linhas de código com operadores relacionais em Java
Fonte: do Autor (2020)

Observe que os comportamentos dos operadores relacionais se comportam da mesma maneira nas duas linguagens. Ocorreu que foi realizada apenas a troca dos valores das variáveis para produzir um resultado diferente nas duas aplicações, porém, se tivessem o mesmo valor de variáveis, teriam sempre os mesmos resultados de comparação relacional.

Agora acompanhe a aplicação dos operadores lógicos.

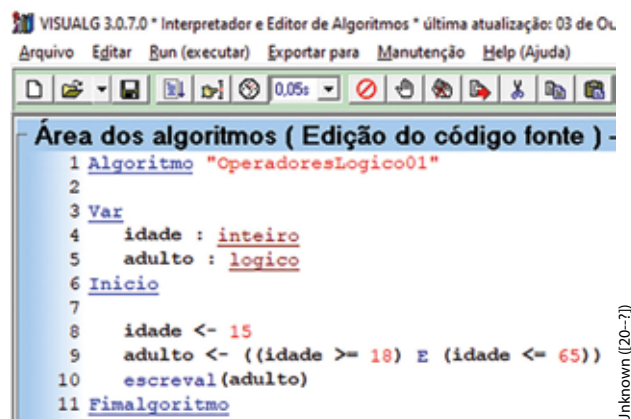


Figura 5 - Tela do VisualG – Exemplo de Operador Lógico E em Português
Fonte: do Autor (2020)

No exemplo anterior, pode ser percebido que uma variável inteira chamada idade será comparada em duas condições, e para que possa ser considerada verdadeira, ela terá que ser verdadeira em ambas as condições; se em qualquer uma das duas condições a comparação não for satisfatória, o resultado final será falso.

PRIMEIRA CONDIÇÃO	SEGUNDA CONDIÇÃO	RESULTADO FINAL
15 >= 18 Falso, porque 15 não é maior ou igual a 18.	15 <= 65 Verdadeiro, porque 15 é menor ou igual a 65.	Falso, porque, com a condição lógica E, todas condições deveriam ser verdadeiras.

Tabela 8 - Explicação do Operador Lógico E
Fonte: do Autor (2020)

Caso o valor da variável idade fosse, por exemplo, alterada para 19, tem-se outro resultado.

PRIMEIRA CONDIÇÃO	SEGUNDA CONDIÇÃO	RESULTADO FINAL
19 >= 18 Verdadeiro, porque 19 é maior ou igual a 18.	19 <= 65 Verdadeiro, porque 19 é menor ou igual a 65.	Verdadeiro Porque todas condições são verdadeiras.

Tabela 9 - Explicação do Operador Lógico E com alteração do valor de Idade
Fonte: do Autor (2020)

```

1 package br.senai.livro;
2
3 public class OperadorLogico {
4     public static void main(String[] args) {
5         boolean fome = true;
6         int diaSemana = 2;
7         boolean pedirPizza;
8         pedirPizza = ((fome==true) || (diaSemana >5));
9         System.out.println(pedirPizza);
10    }
11 }

```

Figura 6 - Tela da IDE Eclipse – Exemplo de Operador Lógico OU em Java
Fonte: do Autor (2020)

Agora em Java (figura anterior) estão em teste as seguintes condições: caso o usuário esteja com fome (*true*) ou o dia da semana seja posterior à sexta (maior que 5), será feito o pedido de uma pizza. Então, nesse caso, bastaria uma dessas condições ser verdadeira para que seja feito o pedido de uma pizza.

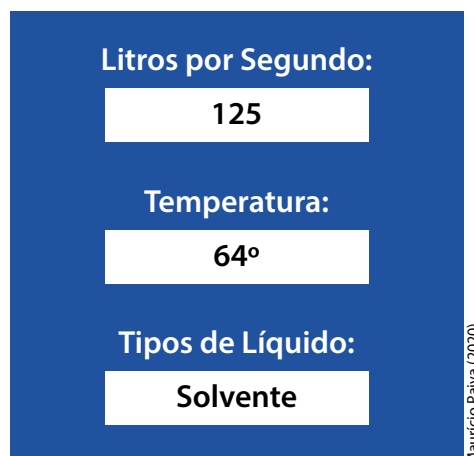
PRIMEIRA CONDIÇÃO	SEGUNDA CONDIÇÃO	RESULTADO FINAL
<code>true == true</code> Verdadeiro, porque fome está <i>true</i> , e <i>true</i> é igual a <i>true</i> .	<code>2 > 5</code> Falso, porque 2 é menor do que 5.	Verdadeiro, porque no operador lógico OU basta uma condição ser verdadeira.

Tabela 10 - Explicação do Operador Lógico OU
Fonte: do Autor (2020)

APLICAÇÃO NA INDÚSTRIA

Acompanhe agora a análise do cenário da indústria química que solicitou um sistema de monitoração de algumas informações dos canos que escoam determinados líquidos utilizados em seus preparos químicos. Neste caso, foi concebido um equipamento acoplado aos canos que era capaz de captar três informações a partir de alguns sensores:

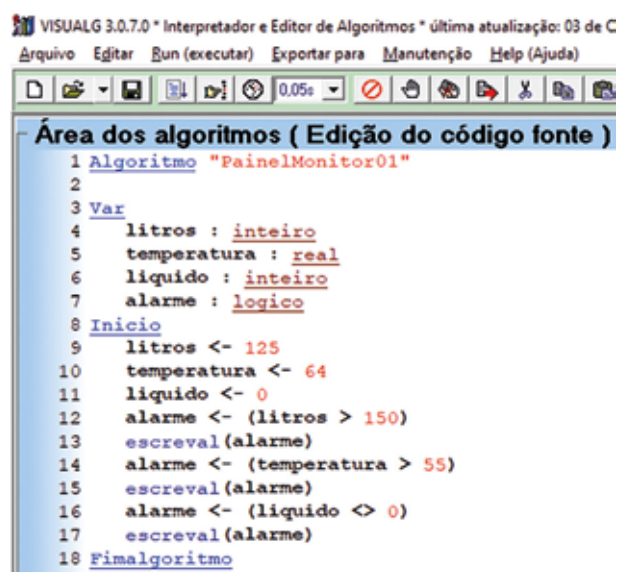
- Vazão em litros por segundo;
- Temperatura;
- Tipo do líquido.



Maurício Paiva (2020)

Figura 7 - Protótipo do Monitor dos Canos
Fonte: do Autor (2020)

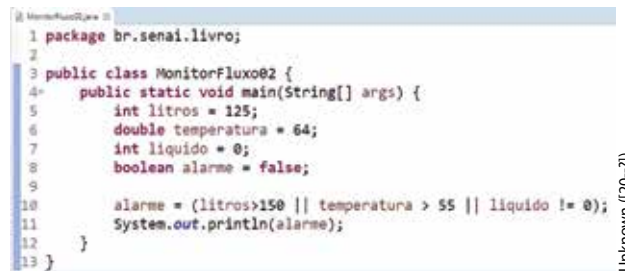
Agora poderiam ser adicionados alguns operadores relacionais e lógicos no comportamento do programa, para que ele fosse capaz de emitir determinados alertas.



Unknown (20--?)

Figura 8 - Tela do VisualG – Proposta Monitor Líquidos com Operadores
Fonte: do Autor (2020)

Neste caso, percebe-se que, pelo fato de se inserir algumas verificações, a monitoração acionará um alarme, pois, na linha 14, uma das condições testadas definirá a variável alarme como verdadeiro, visto que a temperatura naquele momento de medição será de 64° e a condição define que será verdadeiro caso o valor comparado seja maior que 55°. Portanto, como 64 é maior que 55, o resultado atribuído à variável alarme será verdadeiro.



```
1 package br.senai.livro;
2
3 public class MonitorFluxo02 {
4     public static void main(String[] args) {
5         int litros = 125;
6         double temperatura = 64;
7         int liquido = 0;
8         boolean alarme = false;
9
10        alarme = (litros > 150 || temperatura > 55 || liquido != 0);
11        System.out.println(alarme);
12    }
13 }
```

Figura 9 - Tela da IDE Eclipse – Proposta Monitor Líquidos com Operadores
Fonte: do Autor (2020)

Em uma versão também em Java, verifica-se uma nova forma de implementar o algoritmo de verificação do alarme, porém agora foram encadeadas as expressões relacionais (com os operadores $>$ e $!=$) com os operadores lógicos, no caso o OU (em Java, $||$). Com isso, basta uma das expressões testadas resultar em verdadeiro, para que o resultado final seja verdadeiro.

É interessante ressaltar que as expressões têm prioridades, o que é chamado de precedência de operadores. Os operadores aritméticos têm precedência sobre os relacionais e estes têm precedência sobre os operadores lógicos. Por isso, foi possível encadear expressões relacionais com operadores lógicos no exemplo anterior, uma vez que as expressões relacionais são resolvidas antes das expressões lógicas. Se fosse necessário, por algum motivo, executar uma expressão com um operador fora da ordem natural de precedência, esta deve ser colocada entre parênteses.

EXEMPLOS

Para ilustrar esse tema, será apresentado um exemplo que calcula o Índice de Massa Corporal de uma pessoa, o famoso IMC, e informe ao usuário se ele está ou não de acordo com o peso ideal. O cálculo do IMC é: peso dividido (em quilogramas) pela altura (em metros) ao quadrado. Caso o resultado for maior ou igual a 18.5 e menor que 25, a pessoa será considerada dentro do peso ideal para um adulto. Assim, tem-se em Portugal:

```

1 Algoritmo "CalcularIMC"
2
3 Var
4   altura : real
5   peso : real
6   imc : real
7   pesoIdeal : logico
8 Inicio
9   altura <- 1.77
10  peso <- 81.5
11  imc <- (peso / (altura*altura))
12  escreval(imc)
13  pesoIdeal <- ((imc >= 18.5) E (imc < 25))
14  escreval(pesoIdeal)
15 Finalgoritmo
  
```

Figura 10 - Tela do VisualG – Calcular IMC em Portugol
Fonte: do Autor (2020)

Nesse exemplo, o resultado do IMC foi de 26.01, ou seja, um pouco acima do ideal, atribuindo então o valor da variável lógica pesoIdeal igual a falso.

```

1 package br.senai.livro;
2
3 public class CalcularIMC {
4     public static void main(String[] args) {
5         double altura = 1.77;
6         double peso = 77.5;
7         double imc = 0.0;
8         boolean pesoIdeal;
9         imc = peso / (altura*altura);
10        pesoIdeal = (imc >= 18.5 && imc < 25);
11        System.out.println(pesoIdeal);
12    }
13 }
  
```

Figura 11 - Tela da IDE Eclipse – Calcular IMC em Java
Fonte: do Autor (2020)

Já na implementação do cálculo de IMC em Java, atribuiu-se um novo peso à pessoa, fazendo-a emagrecer 4 quilos, o que demonstro que ela ficasse com um valor final de IMC igual a 24.73, ou seja, maior que 18.5 e inferior a 25, ficando assim na proposta de um peso ideal.

**SAIBA
MAIS**

Para ampliar seus conhecimentos, pesquise e leia mais sobre os encadeamentos dos operadores lógicos, pois é comum encontrar testes lógicos encadeados na resolução de diversas soluções computacionais para verificar determinadas situações que devem ser controladas.

Imagine, por exemplo, um carro automatizado para condução sem a intervenção do motorista. Neste caso, vários testes lógicos deveriam ser validados, tais como:

- a) As portas estão fechadas apropriadamente?
- b) As portas estão com as trancas acionadas?
- c) O motorista está com cinto de segurança colocado?
- d) O carro, antes de ser ligado, está no ponto morto?
- e) O carro possui combustível?
- f) O freio de mão do carro está desligado?
- e) Se for período noturno, os faróis estão acesos?

Esses são apenas alguns exemplos de como é possível encadear diversos testes de expressões lógicas e relacionais para determinar comportamentos de um sistema. Procure praticar ao máximo esse tema, pois ele normalmente é muito importante para um profissional de TI, que, com excelente raciocínio lógico, será muito valorizado nas empresas.



REFERÊNCIAS

SOUZA, Marco Antonio de; GOMES, Marcelo Marques; SOARES, Márcio José; CONCILIO, Ricardo. **Algoritmos e lógica de programação**. São Paulo (SP): Thomson Pioneira, 2005.

ARAÚJO, Everton Coimbra de. **Algoritmos**: fundamentos e pratica. Florianópolis: Visual Books, 2005.

MANZANO, José Augusto N. G.; OLIVEIRA, Jayr Figueiredo de. **Algoritmos**: lógica para desenvolvimento de programação de computadores. 24. ed. São Paulo (SP): Érica, 2011.

ANSELMO, Fernando. **Aplicando lógica orientada a objetos em Java**: da lógica à certificação. 3. ed. Florianópolis: Visual Books, 2013.



Serviço Nacional de Aprendizagem Industrial

PELO FUTURO DO TRABALHO