



O SENHOR DOS ALGORITMOS

Domine várias linguagens de programação

RESUMO

Já aprendeu fluxograma e/ou Portugol, mas sente dificuldade de migrar para C# ou Java? Seja como o Senhor dos Algoritmos que entende os fundamentos e, para aprender uma nova linguagem de programação, procura conhecer sua sintaxe. Este e-book contém tabelas para te ajudar na transição de uma linguagem para outra. Bons estudos!

Ramon Chiara

Entenda Programação de uma vez por todas



Sumário

Introdução	4
Como Usar este e-book?	4
Linguagens Usadas.....	8
Fundamentos.....	9
Estrutura Básica	9
Criação de Variáveis	10
Criação de Variáveis do Tipo Inteiro.....	10
Criação de Variáveis do Tipo Real.....	11
Criação de Variáveis do Tipo Lógico.....	11
Criação de Variáveis do Tipo Texto	12
Criação de Variáveis do Tipo Vetor	12
Criação de Variáveis do Tipo Matriz	13
Literais (Valores).....	14
Literais do Tipo Inteiro	14
Literais do Tipo Real	14
Literais do Tipo Lógico	15
Literais do Tipo Texto.....	15
Estruturas Sequenciais	16
Entrada de Dados	16
Ler Valor do Tipo Inteiro	16
Ler Valor do Tipo Real	17
Ler Valor do Tipo Lógico	17
Ler Valor do Tipo Texto	18
Ler Valor do Tipo Vetor.....	18
Ler Valor do Tipo Matriz.....	19
Saída de Dados.....	20
Escrever.....	20
Escrever, Pulando Linha	20
Escrever com Parâmetros.....	21
Escrever com Parâmetros, Pulando Linha	21
Limpar Tela	22
Operações	23
Operação de Atribuição.....	23
Operações com Texto	23
Operações Aritméticas.....	24
Operações com Vetores	25
Operações com Matrizes	26
Operações Relacionais	27
Operações Lógicas	28
Outras Operações	29
Estruturas de Decisão	30
Se-Então.....	30
Se-Então-Senão	31
Se-Então-Senão Encadeado	32
Escolha-Caso	33
Estruturas de Repetição.....	35
Enquanto.....	35
Para	36
Repita-Até.....	37
Faça-Enquanto	38
Interrupção da Repetição	39

Procedimentos	40
Criação de Procedimento.....	40
Criação de Função	41
Execução de Procedimento.....	42
Execução de Função	42
Considerações Finais	43

Introdução

Olá! Meu nome é Ramon Chiara e este e-book é um presente para você do site [Entenda Programação de um vez por todas](#). Ele tem por objetivo facilitar a transição do aprendizado entre linguagens de programação. É muito comum, nos cursos universitários, ter a linguagem Portugol como sendo a primeira a ser aprendida. Logo após, aprende-se alguma linguagem comercial como C# ou Java. Notei, durante os últimos anos lecionando algoritmos, lógica de programação e outras disciplinas, que alguns alunos até conseguem desenvolver algoritmos em fluxograma e/ou Portugol, mas sentem dificuldade na hora de migrar para as linguagens comerciais. Daí, nasceu este e-book como uma forma de ajudá-los nessa transição. Até porque você já deve ter ouvido falar que o importante é saber algoritmos, pois aprender uma nova linguagem é uma questão de aprender a sua sintaxe. Eu também acredito nisso! O nome deste e-book vem desta ideia, ou seja, se você for o **Senhor dos Algoritmos**, dominará qualquer linguagem! ☺

Como Usar este e-book?

Para usar este e-book, você pode desenvolver normalmente o algoritmo sendo estudado usando o que você sabe: fluxograma ou Portugol. Logo após, você traduz para a nova linguagem usando as tabelas fornecidas nas próximas seções que são:

- Fundamentos, onde estão as estruturas básicas dos programas nas diversas linguagens, bem como a criação de variáveis.
- Estruturas Sequenciais, de Decisão e de Repetição, onde você encontra os comandos de cada uma dessas estruturas.
- Funções, onde é mostrado a criação bem como a chamada ou execução das mesmas.

Exemplo: suponha que você tenha desenvolvido o seguinte algoritmo em Portugol do Portugol Studio:

```
programa {
    funcao inicio() {
        inteiro n, fatorial

        escreva("Entre com o valor de n: ")
        leia(n)

        fatorial = 1
        para (inteiro i = 1; i <= n; i = i + 1) {
            fatorial = fatorial * i
        }

        escreva("O fatorial de " + n + " é " + fatorial)
    }
}
```

Para traduzí-lo para Java, basta procurar os comandos equivalentes e ir substituindo linha a linha. Começando pela estrutura básica, iremos ver na tabela correspondente que a estrutura do Java é a que está abaixo, em vermelho, substituindo a estrutura básica do Portugol (que você também pode ver na mesma tabela):

```
import java.util.Scanner;

public class Program {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);

        inteiro n, fatorial

        escreva("Entre com o valor de n: ")
        leia(n)

        fatorial = 1
        para (inteiro i = 1; i <= n; i = i + 1) {
            fatorial = fatorial * i
        }

        escreva("O fatorial de " + n + " é " + fatorial)
    }
}
```

Agora, vamos substituir a declaração das variáveis. Não podemos nos esquecer de colocar o ponto-e-vírgula no final de comandos Java:

```
import java.util.Scanner;

public class Program {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);

        int n, fatorial;

        escreva("Entre com o valor de n: ")
        leia(n)

        fatorial = 1
        para (inteiro i = 1; i <= n; i = i + 1) {
            fatorial = fatorial * i
        }

        escreva("O fatorial de " + n + " é " + fatorial)
    }
}
```

Continuando, traduzimos o comando de impressão na tela:

```
import java.util.Scanner;

public class Program {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);

        int n, fatorial;

        System.out.print("Entre com o valor de n: ");
        leia(n)

        fatorial = 1
        para (inteiro i = 1; i <= n; i = i + 1) {
            fatorial = fatorial * i
        }

        escreva("O fatorial de " + n + " é " + fatorial)
    }
}
```

E o comando que lê a variável inteira "n":

```
import java.util.Scanner;

public class Program {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);

        int n, fatorial;

        System.out.print("Entre com o valor de n: ");
        n = Integer.parseInt(console.nextLine());

        fatorial = 1
        para (inteiro i = 1; i <= n; i = i + 1) {
            fatorial = fatorial * i
        }

        escreva("O fatorial de " + n + " é " + fatorial)
    }
}
```

Logo em seguida, temos uma operação de atribuição, que em Java é a mesma coisa, com exceção que, de novo, não podemos esquecer de colocar o ponto-e-vírgula no final da linha:

```
import java.util.Scanner;

public class Program {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);

        int n, fatorial;

        System.out.print("Entre com o valor de n: ");
        n = Integer.parseInt(console.nextLine());

        fatorial = 1;
        para (inteiro i = 1; i <= n; i = i + 1) {
            fatorial = fatorial * i
        }

        escreva("O fatorial de " + n + " é " + fatorial)
    }
}
```

Depois, temos o laço do tipo "para" e, também, o comando que está dentro dele. Nesse caso, temos uma operação de multiplicação e uma de atribuição que também são iguais nas duas linguagens:

```
import java.util.Scanner;

public class Program {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);

        int n, fatorial;

        System.out.print("Entre com o valor de n: ");
        n = Integer.parseInt(console.nextLine());

        fatorial = 1;
        for (int i = 1; i <= n; i = i + 1) {
            fatorial = fatorial * i;
        }

        escreva("O fatorial de " + n + " é " + fatorial)
    }
}
```

Por último, mais um comando de impressão na tela:

```
import java.util.Scanner;

public class Program {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);

        int n, fatorial;

        System.out.print("Entre com o valor de n: ");
        n = Integer.parseInt(console.nextLine());

        fatorial = 1;
        for (int i = 1; i <= n; i = i + 1) {
            fatorial = fatorial * i;
        }

        System.out.println("O fatorial de " + n + " é " + fatorial);
    }
}
```

Linguagens Usadas

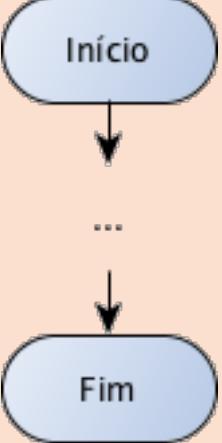
Na presente versão deste texto, você encontrará tabelas para a transição entre as seguintes linguagens:

- Fluxograma
- Portugol do VisuAlg
- Portugol do Portugol Studio
- C#
- Java

Estarei sempre atualizando este e-book para que ele conte cole mais linguagens! Por isso, é importante que você **cadastre o seu e-mail no site** [Entenda Programação de uma vez por todas.](#) Assim, quando sair uma versão mais atual ou alguma novidade, eu te mando um e-mail, combinado? Então vamos às tabelas!

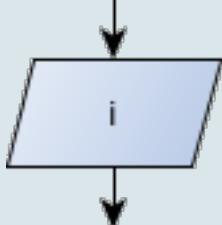
Fundamentos

Estrutura Básica

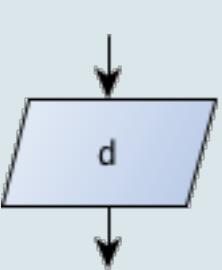
Linguagem	Comando ou Exemplo	Observações
Fluxograma	 <pre> graph TD Inicio([Início]) --> ...1[...] ...1 --> ...2[...] ...2 --> Fim([Fim]) </pre>	
Portugol do VisuAlg	<pre> algoritmo "meu programa" var ... inicio ... fimalgoritmo </pre>	
Portugol do Portugol Studio	<pre> programa { inclua biblioteca Matematica funcao inicio() { ... } } </pre>	← Essa linha só é necessária se você for usar as funções potencia ou raiz.
C#	<pre> using System; public class Program { public static void Main(string[] args) { ... } } </pre>	
Java	<pre> import java.util.Scanner; public class Program { public static void main(String[] args) { Scanner console = new Scanner(System.in); ... } } </pre>	

Criação de Variáveis

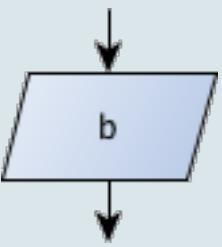
Criação de Variáveis do Tipo Inteiro

Linguagem	Comando ou Exemplo	Observações
Fluxograma		Nas minhas aulas eu não vejo a necessidade dessa informação no fluxograma. Mas, alguns autores usam o símbolo de "dados" (paralelogramo).
Portugol do VisuAlg	<code>var i: inteiro</code>	
Portugol do Portugol Studio	<code>inteiro i</code>	
C#	<code>sbyte i; short i; int i; long i; byte i; ushort i; uint i; ulong i;</code>	O C# possui 8 tipos inteiros. Normalmente usamos o tipo int ou long.
Java	<code>byte i; short i; int i; long i;</code>	O Java possui 4 tipos inteiros. Normalmente usamos o tipo int ou long.

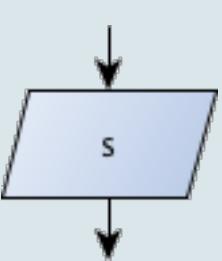
Criação de Variáveis do Tipo Real

Linguagem	Comando ou Exemplo	Observações
Fluxograma		Nas minhas aulas eu não vejo a necessidade dessa informação no fluxograma. Mas, alguns autores usam o símbolo de "dados" (paralelogramo).
Portugol do VisuAlg	<code>var d: real</code>	
Portugol do Portugol Studio	<code>real d</code>	
C#	<code>float d; double d; decimal d;</code>	O C# possui 3 tipos reais. Normalmente usamos o tipo double. Quando for para tratar quantias monetárias, usamos o decimal.
Java	<code>float d; double d;</code>	O Java possui 2 tipos reais. Normalmente usamos o tipo double. Quando for para tratar quantias monetárias, usamos objetos da classe BigDecimal (não se preocupe com isso durante o aprendizado).

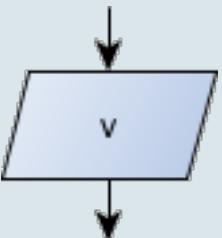
Criação de Variáveis do Tipo Lógico

Linguagem	Comando ou Exemplo	Observações
Fluxograma		Nas minhas aulas eu não vejo a necessidade dessa informação no fluxograma. Mas, alguns autores usam o símbolo de "dados" (paralelogramo).
Portugol do VisuAlg	<code>var b: logico</code>	
Portugol do Portugol Studio	<code>logico b</code>	
C#	<code>bool b;</code>	
Java	<code>boolean b;</code>	

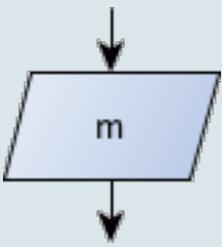
Criação de Variáveis do Tipo Texto

Linguagem	Comando ou Exemplo	Observações
Fluxograma		Nas minhas aulas eu não vejo a necessidade dessa informação no fluxograma. Mas, alguns autores usam o símbolo de "dados" (paralelogramo).
Portugol do VisuAlg	var s: caractere	
Portugol do Portugol Studio	caracter s cadeia s	O tipo caracter serve para guardar apenas um caractere.
C#	char s; string s;	O tipo char serve para guardar apenas um caractere.
Java	char s; String s;	O tipo char serve para guardar apenas um caractere.

Criação de Variáveis do Tipo Vetor

Linguagem	Comando ou Exemplo	Observações
Fluxograma		Nas minhas aulas eu não vejo a necessidade dessa informação no fluxograma. Mas, alguns autores usam o símbolo de "dados" (paralelogramo).
Portugol do VisuAlg	var v: vetor[linha1..linhaN] de tipo	
Portugol do Portugol Studio	tipo v[linhas]	
C#	tipo[] v = new tipo[linhas];	
Java	tipo[] v = new tipo[linhas];	

Criação de Variáveis do Tipo Matriz

Linguagem	Comando ou Exemplo	Observações
Fluxograma		Nas minhas aulas eu não vejo a necessidade dessa informação no fluxograma. Mas, alguns autores usam o símbolo de "dados" (paralelogramo).
Portugol do VisuAlg	var m: vetor[linha1..linhaN, coluna1..colunaN] de tipo	
Portugol do Portugol Studio	tipo v[[linhas][colunas]	
C#	tipo[,] m = new tipo[[linhas, colunas];	
Java	tipo[][] m = new tipo[[linhas][colunas];	

Literais (Valores)

Literais do Tipo Inteiro

Linguagem	Comando ou Exemplo	Observações
Fluxograma	3	
Portugol do VisuAlg	3	
Portugol do Portugol Studio	3	
C#	3 3L 3U 3UL	int long unsigned int unsigned long
Java	3 3L	int long

Literais do Tipo Real

Linguagem	Comando ou Exemplo	Observações
Fluxograma	3.14	
Portugol do VisuAlg	3.14	
Portugol do Portugol Studio	3.14	
C#	3.14F 3.14 3.14M	float double decimal
Java	3.14F 3.14	float double

Literais do Tipo Lógico

Linguagem	Comando ou Exemplo	Observações
Fluxograma	F V	
Portugol do VisuAlg	FALSO VERDADEIRO	
Portugol do Portugol Studio	falso verdadeiro	
C#	false true	
Java	false true	

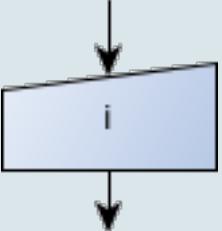
Literais do Tipo Texto

Linguagem	Comando ou Exemplo	Observações
Fluxograma	"texto entre aspas"	
Portugol do VisuAlg	"texto entre aspas"	
Portugol do Portugol Studio	'C' "texto entre aspas"	caracter cadeia
C#	'C' "texto entre aspas"	char string
Java	'C' "texto entre aspas"	char String

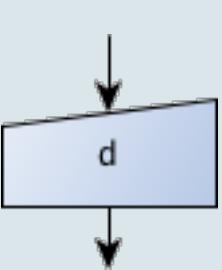
Estruturas Sequenciais

Entrada de Dados

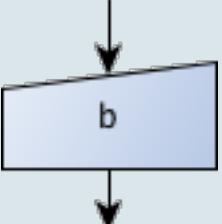
Ler Valor do Tipo Inteiro

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 A fluxograma mostra um retângulo de entrada com uma seta apontando para cima e outra saída da caixa.	Para lembrar, costumo comparar o símbolo com um teclado.
Portugol do VisuAlg	leia(i)	
Portugol do Portugol Studio	leia(i)	
C#	i = sbyte.Parse(Console.ReadLine()); i = short.Parse(Console.ReadLine()); i = int.Parse(Console.ReadLine()); i = long.Parse(Console.ReadLine()); i = byte.Parse(Console.ReadLine()); i = ushort.Parse(Console.ReadLine()); i = uint.Parse(Console.ReadLine()); i = ulong.Parse(Console.ReadLine());	
Java	i = Byte.parseByte(console.nextLine()); i = Short.parseShort(console.nextLine()); i = Integer.parseInt(console.nextLine()); i = Long.parseLong(console.nextLine());	

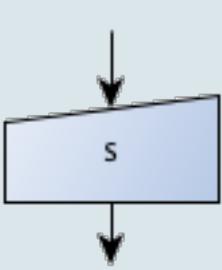
Ler Valor do Tipo Real

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 A fluxograma mostra uma caixa de entrada com a letra 'd' dentro, com uma seta apontando para baixo.	Para lembrar, costumo comparar o símbolo com um teclado.
Portugol do VisuAlg	leia(d)	
Portugol do Portugol Studio	leia(d)	
C#	d = float.Parse(Console.ReadLine()); d = double.Parse(Console.ReadLine()); d = decimal.Parse(Console.ReadLine());	
Java	d = Float.parseFloat(console.nextLine()); d = Double.parseDouble(console.nextLine());	

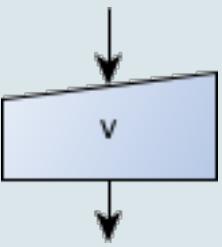
Ler Valor do Tipo Lógico

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 A fluxograma mostra uma caixa de entrada com a letra 'b' dentro, com uma seta apontando para baixo.	Para lembrar, costumo comparar o símbolo com um teclado.
Portugol do VisuAlg	leia(b)	
Portugol do Portugol Studio	leia(b)	
C#	b = bool.Parse(Console.ReadLine());	
Java	b = Boolean.parseBoolean(console.nextLine());	

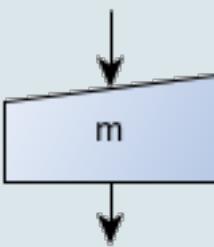
Ler Valor do Tipo Texto

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 A caixa de entrada de dados contém a letra 's'. Um ponto final está acima da caixa.	Para lembrar, costumo comparar o símbolo com um teclado.
Portugol do VisuAlg	leia(s)	
Portugol do Studio	leia(s)	
C#	s = char.Parse(Console.ReadLine()); s = Console.ReadLine();	
Java	s = console.nextLine();	

Ler Valor do Tipo Vetor

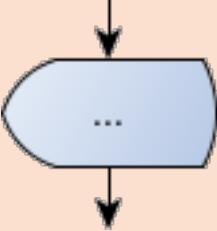
Linguagem	Comando ou Exemplo	Observações
Fluxograma	 A caixa de entrada de dados contém a letra 'v'. Um ponto final está acima da caixa.	Para lembrar, costumo comparar o símbolo com um teclado.
Portugol do VisuAlg	para i de linha1 ate linhaN faca leia(v[i]) fimpara	Para ler um vetor, temos que ler cada uma das variáveis individualmente.
Portugol do Studio	para (inteiro i = 0; i < linhas; i = i + 1) { leia(v[i]) }	Para ler um vetor, temos que ler cada uma das variáveis individualmente.
C#	for (int i = 0; i < v.Length; i++) { v[i] = comando de leitura }	Para ler um vetor, temos que ler cada uma das variáveis individualmente.
Java	for (int i = 0; i < v.length; i++) { v[i] = comando de leitura }	Para ler um vetor, temos que ler cada uma das variáveis individualmente.

Ler Valor do Tipo Matriz

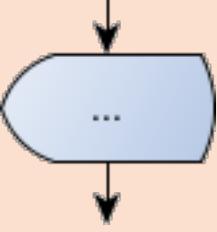
Linguagem	Comando ou Exemplo	Observações
Fluxograma		Para lembrar, costumo comparar o símbolo com um teclado.
Portugol do VisuAlg	<pre> para i de linha1 ate linhaN faca para j de coluna1 ate colunaN faca leia(m[i, j]) fimpara fimpara </pre>	Para ler uma matriz, temos que ler cada uma das variáveis individualmente.
Portugol do Portugol Studio	<pre> para (inteiro i = 0; i < linhas; i = i + 1) { para (inteiro j = 0; j < colunas; j = j + 1) { leia(m[i][j]) } } </pre>	Para ler uma matriz, temos que ler cada uma das variáveis individualmente.
C#	<pre> for (int i = 0; i < m.GetLength(0); i++) { for (int j = 0; j < m.GetLength(1); j++) { m[i, j] = comando de leitura } } </pre>	Para ler uma matriz, temos que ler cada uma das variáveis individualmente.
Java	<pre> for (int i = 0; i < m.length; i++) { for (int j = 0; j < m[i].length; j++) { m[i][j] = comando de leitura } } </pre>	Para ler uma matriz, temos que ler cada uma das variáveis individualmente.

Saída de Dados

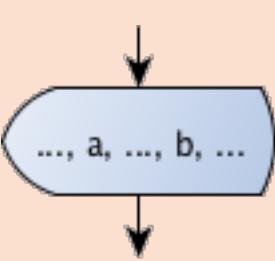
Escrever

Linguagem	Comando ou Exemplo	Observações
Fluxograma		Para lembrar, costumo comparar o símbolo com um monitor (antigo), visto de lado.
Portugol do VisuAlg	escreva(...)	
Portugol do Portugol Studio	escreva(...)	
C#	Console.WriteLine(...);	
Java	System.out.println(...);	

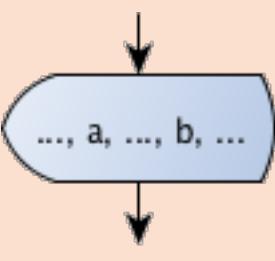
Escrever, Pulando Linha

Linguagem	Comando ou Exemplo	Observações
Fluxograma		Para lembrar, costumo comparar o símbolo com um monitor (antigo), visto de lado.
Portugol do VisuAlg	escreval(...)	
Portugol do Portugol Studio	escreva(..., "\n")	
C#	Console.WriteLine(...);	
Java	System.out.println(...);	

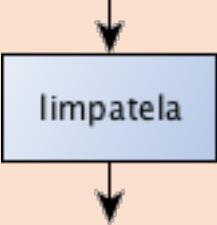
Escrever com Parâmetros

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 A imagem mostra um fluxograma com uma saída de dados. Um retângulo azul contém os símbolos ..., a, ..., b, ... e é conectado por uma seta para baixo a um triângulo de saída.	Para lembrar, costumo comparar o símbolo com um monitor (antigo), visto de lado.
Portugol do VisuAlg	escreva(..., a, ..., b, ...)	
Portugol do Portugol Studio	escreva(..., a, ..., b, ...)	
C#	Console.WriteLine("...{0}...{1}...", a, b, ...);	Para saber o que colocar no lugar do {n}, consulte a documentação do C# .
Java	System.out.printf("...%t...%t...", a, b, ...);	Para saber o que colocar no lugar do t, consulte a documentação do Java .

Escrever com Parâmetros, Pulando Linha

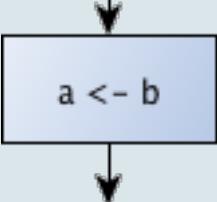
Linguagem	Comando ou Exemplo	Observações
Fluxograma	 A imagem mostra um fluxograma com uma saída de dados. Um retângulo azul contém os símbolos ..., a, ..., b, ... e é conectado por uma seta para baixo a um triângulo de saída.	Para lembrar, costumo comparar o símbolo com um monitor (antigo), visto de lado.
Portugol do VisuAlg	escreval(..., a, ..., b, ...)	
Portugol do Portugol Studio	escreva(..., a, ..., b, ... ,"\n")	
C#	Console.WriteLine("...{0}...{1}...", a, b, ...);	Para saber o que colocar no lugar do {n}, consulte a documentação do C# .
Java	System.out.printf("...%t...%t...%n", a, b, ...);	Para saber o que colocar no lugar do t, consulte a documentação do Java .

Limpar Tela

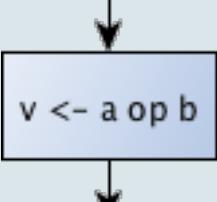
Linguagem	Comando ou Exemplo	Observações
Fluxograma		Não tem comando específico. Nesse caso, pode-se usar o símbolo de processamento para executar um comando de limpar tela.
Portugol do VisuAlg	limpatela	
Portugol do Portugol Studio	limpa()	
C#	Console.Clear();	
Java	Não tem comando específico.	

Operações

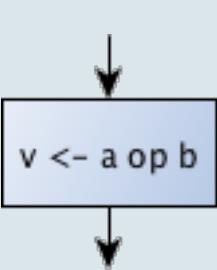
Operação de Atribuição

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 <pre>graph TD; A[a <- b]</pre>	
Portugol do VisuAlg	<code>a <- b</code>	
Portugol do Portugol Studio	<code>a = b</code>	
C#	<code>a = b;</code>	
Java	<code>a = b;</code>	

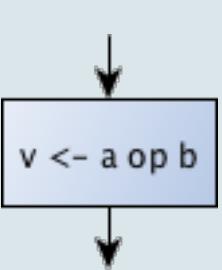
Operações com Texto

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 <pre>graph TD; A[v <- a op b]</pre>	<p><i>op</i> é um dos operadores mostrados abaixo. Coloquei a operação de atribuição porque, normalmente, quando criamos alguma expressão, fórmula, etc. para ser calculada, queremos armazenar o valor obtido.</p>
Portugol do VisuAlg	<code>"texto" + "texto"</code> <code>"texto" + a</code> <code>a + "texto"</code>	Casos onde ocorrem concatenação, ou seja, a junção de dois textos ou de um texto com algum outro valor.
Portugol do Portugol Studio	<code>"texto" + "texto"</code> <code>"texto" + a</code> <code>a + "texto"</code>	Casos onde ocorrem concatenação, ou seja, a junção de dois textos ou de um texto com algum outro valor.
C#	<code>"texto" + "texto"</code> <code>"texto" + a</code> <code>a + "texto"</code>	Casos onde ocorrem concatenação, ou seja, a junção de dois textos ou de um texto com algum outro valor.
Java	<code>"texto" + "texto"</code> <code>"texto" + a</code> <code>a + "texto"</code>	Casos onde ocorrem concatenação, ou seja, a junção de dois textos ou de um texto com algum outro valor.

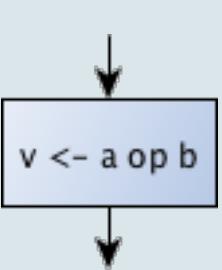
Operações Aritméticas

Linguagem	Comando ou Exemplo	Observações
Fluxograma		op é um dos operadores mostrados abaixo. Coloquei a operação de atribuição porque, normalmente, quando criamos alguma expressão, fórmula, etc. para ser calculada, queremos armazenar o valor obtido.
Portugol do VisuAlg	$a + b$ $a - b$ $a * b$ a / b $a \ b$ $a \% b$ ou $a \text{ mod } b$ $a ^ b$ $\text{RaizQ}(a)$	Adição Subtração Multiplicação Divisão Divisão Inteira Resto da Divisão Potenciação Raiz Quadrada
Portugol do Portugol Studio	$a + b$ $a - b$ $a * b$ a / b $a \ b$ $a \% b$ $\text{Matematica.potencia}(a, b)$ $\text{Matematica.raiz}(a, 2)$	Adição Subtração Multiplicação Divisão Divisão Inteira (a e b devem ser inteiros) Resto da Divisão Potenciação Raiz Quadrada
C#	$a + b$ $a - b$ $a * b$ a / b $a \ b$ $a \% b$ $\text{Math.Pow}(a, b)$ $\text{Math.Sqrt}(a)$	Adição Subtração Multiplicação Divisão Divisão Inteira (a e b devem ser inteiros) Resto da Divisão Potenciação Raiz Quadrada
Java	$a + b$ $a - b$ $a * b$ a / b $a \ b$ $a \% b$ $\text{Math.pow}(a, b)$ $\text{Math.sqrt}(a)$	Adição Subtração Multiplicação Divisão Divisão Inteira (a e b devem ser inteiros) Resto da Divisão Potenciação Raiz Quadrada

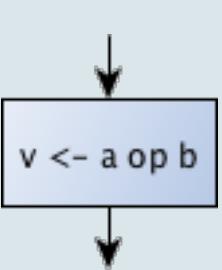
Operações com Vetores

Linguagem	Comando ou Exemplo	Observações
Fluxograma		op é um dos operadores mostrados abaixo. Coloquei a operação de atribuição porque, normalmente, quando criamos alguma expressão, fórmula, etc. para ser calculada, queremos armazenar o valor obtido.
Portugol do VisuAlg	v[índice da linha]	Acesso à uma variável ou valor do vetor
	Não tem comando específico.	Tamanho do vetor
Portugol do Portugol Studio	v[índice da linha]	Acesso à uma variável ou valor do vetor
	Util.numero_elementos(v)	Tamanho do vetor → é necessário incluir a biblioteca Util
C#	v[índice da linha]	Acesso à uma variável ou valor do vetor (índice começa em 0 e termina em tamanho - 1)
	v.Length	Tamanho do vetor
Java	v[índice da linha]	Acesso à uma variável ou valor do vetor (índice começa em 0 e termina em tamanho - 1)
	v.length	Tamanho do vetor

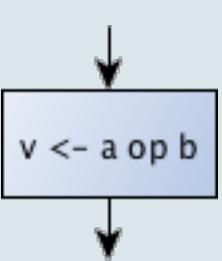
Operações com Matrizes

Linguagem	Comando ou Exemplo	Observações
Fluxograma		op é um dos operadores mostrados abaixo. Coloquei a operação de atribuição porque, normalmente, quando criamos alguma expressão, fórmula, etc. para ser calculada, queremos armazenar o valor obtido.
Portugol do VisuAlg	<p><code>m[índice da linha, índice da coluna]</code></p> <p>Não tem comando específico.</p> <p>Não tem comando específico.</p> <p>Não tem comando específico.</p>	<p>Acesso à uma variável ou valor da matriz (índices começam em 0 e terminam em tamanho - 1)</p> <p>Tamanho da matriz</p> <p>Quantidade de linhas</p> <p>Quantidade de colunas de uma linha</p>
Portugol do Portugol Studio	<p><code>m[índice da linha][índice da coluna]</code></p> <p>Não tem comando específico.</p> <p><code>Util.numero_linhas(m)</code></p> <p><code>Util.numero_colunas(m)</code></p>	<p>Acesso à uma variável ou valor da matriz (índices começam em 0 e terminam em tamanho - 1)</p> <p>Tamanho da matriz</p> <p>Quantidade de linhas</p> <p>Quantidade de colunas</p> <p>→ é necessário incluir a biblioteca Util</p>
C#	<p><code>m[índice da linha, índice da coluna]</code></p> <p><code>m.Length</code></p> <p><code>m.GetLength(0)</code></p> <p><code>m.GetLength(1)</code></p>	<p>Acesso à uma variável ou valor da matriz (índices começam em 0 e terminam em tamanho - 1)</p> <p>Tamanho da matriz</p> <p>Quantidade de linhas</p> <p>Quantidade de colunas</p>
Java	<p><code>m[índice da linha][índice da coluna]</code></p> <p>Não tem comando específico.</p> <p><code>m.length</code></p> <p><code>m[índice da linha].length</code></p>	<p>Acesso à uma variável ou valor da matriz (índices começam em 0 e terminam em tamanho - 1)</p> <p>Tamanho da matriz</p> <p>Quantidade de linhas</p> <p>Quantidade de colunas de uma linha</p>

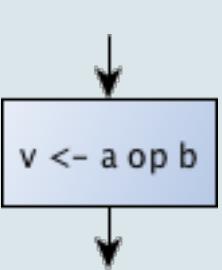
Operações Relacionais

Linguagem	Comando ou Exemplo	Observações
Fluxograma		op é um dos operadores mostrados abaixo. Coloquei a operação de atribuição porque, normalmente, quando criamos alguma expressão, fórmula, etc. para ser calculada, queremos armazenar o valor obtido.
Portugol do VisuAlg	$a < b$ $a \leq b$ $a > b$ $a \geq b$ $a = b$ $a \neq b$	Menor Menor ou Igual Maior Maior ou Igual Igual Diferente
Portugol do Portugol Studio	$a < b$ $a \leq b$ $a > b$ $a \geq b$ $a == b$ $a != b$	Menor Menor ou Igual Maior Maior ou Igual Igual Diferente
C#	$a < b$ $a \leq b$ $a > b$ $a \geq b$ $a == b$ $a != b$	Menor Menor ou Igual Maior Maior ou Igual Igual Diferente
Java	$a < b$ $a \leq b$ $a > b$ $a \geq b$ $a == b$ $a != b$	Menor Menor ou Igual Maior Maior ou Igual Igual Diferente

Operações Lógicas

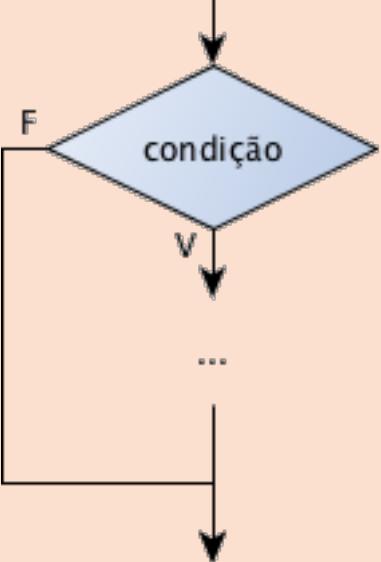
Linguagem	Comando ou Exemplo	Observações
Fluxograma		op é um dos operadores mostrados abaixo. Coloquei a operação de atribuição porque, normalmente, quando criamos alguma expressão, fórmula, etc. para ser calculada, queremos armazenar o valor obtido.
Portugol do VisuAlg	$a \wedge b$ $a \vee b$ $a \vee\wedge b$ $\neg a$	E OU OU EXCLUSIVO NEGAÇÃO
Portugol do Portugol Studio	$a \wedge b$ $a \vee b$ Não tem comando específico. $\neg a$	E OU OU EXCLUSIVO NEGAÇÃO
C#	$a \&& b$ $a \mid\mid b$ Não tem comando específico. $\neg a$	E OU OU EXCLUSIVO NEGAÇÃO
Java	$a \&& b$ $a \mid\mid b$ Não tem comando específico. $\neg a$	E OU OU EXCLUSIVO NEGAÇÃO

Outras Operações

Linguagem	Comando ou Exemplo	Observações
Fluxograma		op é um dos operadores mostrados abaixo. Coloquei a operação de atribuição porque, normalmente, quando criamos alguma expressão, fórmula, etc. para ser calculada, queremos armazenar o valor obtido.
Portugol do VisuAlg	Não tem comando específico. Não tem comando específico. Não tem comando específico.	Incremento ($a <- a + 1$) Decremento ($a <- a - 1$) Operação com Atribuição ($a = a \text{ op } b$)
Portugol do Portugol Studio	$a++$ $a--$ $a \text{ op} = b$	Incremento ($a <- a + 1$) Decremento ($a <- a - 1$) Operação com Atribuição ($a = a \text{ op } b$)
C#	$a++$ ou $++a$ $a--$ ou $--a$ $a \text{ op} = b$	Incremento ($a <- a + 1$) Decremento ($a <- a - 1$) Operação com Atribuição ($a = a \text{ op } b$)
Java	$a++$ ou $++a$ $a--$ ou $--a$ $a \text{ op} = b$	Incremento ($a <- a + 1$) Decremento ($a <- a - 1$) Operação com Atribuição ($a = a \text{ op } b$)

Estruturas de Decisão

Se-Então

Linguagem	Comando ou Exemplo	Observações
Fluxograma		
Portugol do VisuAlg	se condição entao ... fimse	
Portugol do Portugol Studio	se (condição) { ... }	
C#	if (condição) { ... }	
Java	if (condição) { ... }	

Se-Então-Senão

Linguagem	Comando ou Exemplo	Observações
Fluxograma	<pre> graph TD Start(()) --> Cond{condição} Cond -- F --> Left[...] Cond -- V --> Right[...] Left --> Join(()) Right --> Join Join --> End(()) </pre>	
Portugol do VisuAlg	<pre> se condição entao ... senao ... fimse </pre>	
Portugol do Portugol Studio	<pre> se (condição) { ... } senao { ... } </pre>	
C#	<pre> if (condição) { ... } else { ... } </pre>	
Java	<pre> if (condição) { ... } else { ... } </pre>	

Se-Então-Senão Encadeado

Linguagem	Comando ou Exemplo	Observações
Fluxograma	<pre> graph TD Start(()) --> Cond1{condição1} Cond1 -- V --> ...1[...] Cond1 -- F --> Cond2{condição2} Cond2 -- V --> ...2[...] Cond2 -- F --> End(()) ...1 --> End ...2 --> End </pre>	
Portugol do VisuAlg	<pre> se condição1 entao ... senao se condição2 entao ... senao ... fimse fimse </pre>	
Portugol do Portugol Studio	<pre> se (condição1) { ... } senao se (condição2) { ... } senao { ... } </pre>	
C#	<pre> if (condição1) { ... } else if (condição2) { ... } else { ... } </pre>	
Java	<pre> if (condição1) { ... } else if (condição2) { ... } else { ... } </pre>	

Escolha-Caso

Linguagem	Comando ou Exemplo	Observações
Fluxograma	<pre> graph TD Start(()) --> Cond1{v = valor1} Cond1 -- V --> Block1[...] Cond1 -- F --> Cond2{v = valor2 ou v = valor3} Cond2 -- V --> Block2[...] Cond2 -- F --> Block3[...] Block3 --> Fim(()) Block1 --> Fim Block2 --> Fim </pre>	Alguns autores utilizam apenas um losango, com vários fluxos saindo dele, cada um com um dos casos previstos.
Portugol do VisuAlg	<pre> escolha variável caso valor1 ... caso valor2, valor3 ... outrocaso ... fimescolha </pre>	
Portugol do Portugol Studio	<pre> escolha (variável) { caso valor1: ... pare caso valor2: caso valor3: ... pare caso contrario: ... } </pre>	
C#	<pre> switch (variável) { case valor1: ... break; case valor2: case valor3: ... break; default: ... break; } </pre>	

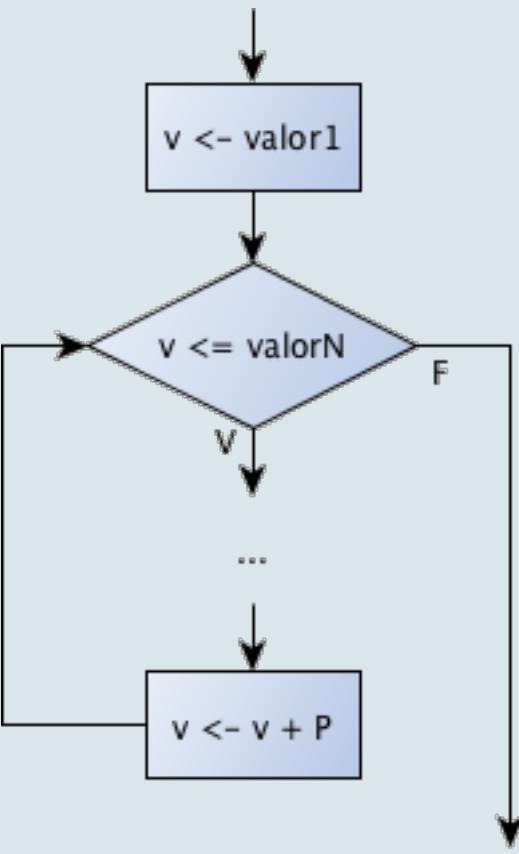
Java

```
switch (variável) {
    case valor1:
        ...
        break;
    case valor2:
    case valor3:
        ...
        break;
    default:
        ...
        break;
}
```

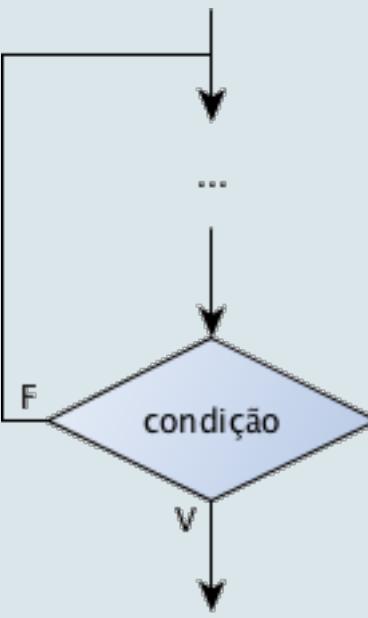
Estruturas de Repetição

Enquanto

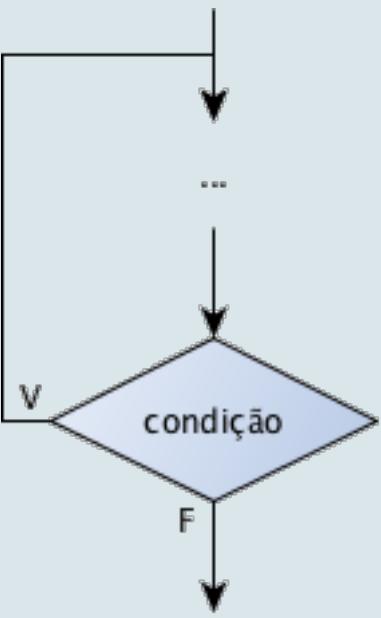
Linguagem Fluxograma	Comando ou Exemplo	Observações
Portugol do VisuAlg	enquanto condição faça ... fim enquanto	
Portugol do Portugol Studio	enquanto (condição) { ... }	
C#	while (condição) { ... }	
Java	while (condição) { ... }	

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 <pre> v <- valor1 v <= valorN? S: ... v += P loop volta para v <= valorN? F: loop termina </pre>	Alguns autores utilizam apenas um hexágono alongado ao invés do losango; dentro dele, colocam o valor inicial, o valor final e o passo.
Portugol do VisuAlg	<pre> para variável de valor1 ate valorN passo P faca ... fimpara </pre>	
Portugol do Portugol Studio	<pre> para (variável = valor1; variável <= valorN; variável += P) { ... } </pre>	
C#	<pre> for (variável = valor1; variável <= valorN; variável += P) { ... } </pre>	
Java	<pre> for (variável = valor1; variável <= valorN; variável += P) { ... } </pre>	

Repita-Até

Linguagem	Comando ou Exemplo	Observações
Fluxograma		
Portugol do VisuAlg	repita ... ate condição	Termina quando a condição for verdadeira.
Portugol do Portugol Studio	Não tem comando específico.	Mas, é possível usar a estrutura "enquanto" com a condição invertida.
C#	Não tem comando específico.	Mas, é possível usar a estrutura "enquanto" com a condição invertida.
Java	Não tem comando específico.	Mas, é possível usar a estrutura "enquanto" com a condição invertida.

Faça-Enquanto

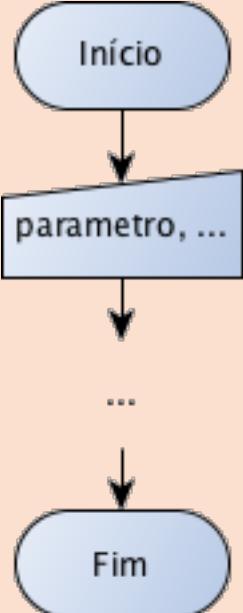
Linguagem	Comando ou Exemplo	Observações
Fluxograma		
Portugol do VisuAlg	Não tem comando específico.	Mas, é possível usar a estrutura "repita-até" com a condição invertida.
Portugol do Portugol Studio	faca { ... } enquanto (condição)	
C#	do { ... } while (condição);	
Java	do { ... } while (condição);	

Interrupção da Repetição

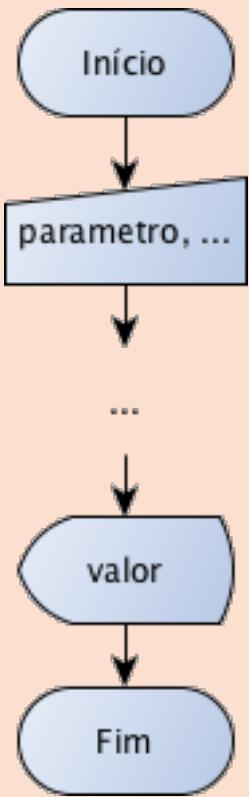
Linguagem	Comando ou Exemplo	Observações
Fluxograma		<p>A interrupção da repetição é feita de forma normal, com uma seta de fluxo saindo do laço (é comum que haja alguma condição dentro do laço que cause essa interrupção).</p> <p>No exemplo ao lado, é mostrada a interrupção em um laço do tipo “faça-enquanto”, mas a interrupção pode ser usada em qualquer estrutura de repetição.</p>
Portugol do VisuAlg Portugol do Portugol Studio	interrompa	Não tem comando específico.
C#	break;	
Java	break;	

Procedimentos

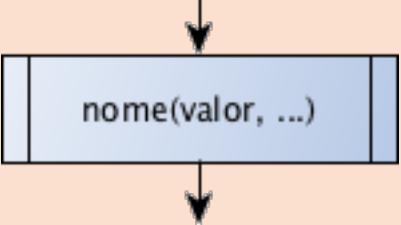
Criação de Procedimento

Linguagem	Comando ou Exemplo	Observações
Fluxograma		Aqui, você irá fazer outro fluxograma, anotando o nome dele. Pode-se usar o comando de entrada de dados para indicar a leitura dos parâmetros.
Portugol do VisuAlg	<pre>procedimento nome(parâmetro: tipo; ...) var ... inicio ... fimprocedimento</pre>	
Portugol do Portugol Studio	<pre>funcao vazio nome(parâmetro: tipo; ...) { ... }</pre>	
C#	<pre>static void nome(tipo parâmetro, ...) { ... }</pre>	
Java	<pre>static void nome(tipo parâmetro, ...) { ... }</pre>	

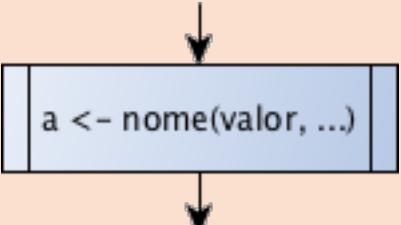
Criação de Função

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 <pre> graph TD Inicio([Início]) --> Parametro[/parametro, ...] Parametro --> Ellipsis1[...] Ellipsis1 --> Valor([valor]) Valor --> Fim([Fim]) </pre>	Aqui, você irá fazer outro fluxograma, anotando o nome dele. Pode-se usar o comando de entrada de dados para indicar a leitura dos parâmetros. O retorno do valor pode ser indicado por um comando de saída.
Portugol do VisuAlg	<pre> funcao nome(parâmetro: tipo; ...): tipo var ... inicio ... retorne valor fimfuncao </pre>	
Portugol do Portugol Studio	<pre> funcao tipo nome(parâmetro: tipo; ...) { ... retorne valor } </pre>	
C#	<pre> static tipo nome(tipo parâmetro, ...) { ... return valor; } </pre>	
Java	<pre> static tipo nome(tipo parâmetro, ...) { ... return valor; } </pre>	

Execução de Procedimento

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 A rectangular flowchart node with a downward-pointing arrow above it. Inside the node, the text "nome(valor, ...)" is written in a light blue area.	
Portugol do VisuAlg	<code>nome(valor, ...)</code>	
Portugol do Portugol Studio	<code>nome(valor, ...)</code>	
C#	<code>nome(valor, ...);</code>	
Java	<code>nome(valor, ...);</code>	

Execução de Função

Linguagem	Comando ou Exemplo	Observações
Fluxograma	 A rectangular flowchart node with a downward-pointing arrow above it. Inside the node, the text "a <- nome(valor, ...)" is written in a light blue area.	
Portugol do VisuAlg	<code>a <- nome(valor, ...)</code>	
Portugol do Portugol Studio	<code>a = nome(valor, ...)</code>	
C#	<code>a = nome(valor, ...);</code>	
Java	<code>a = nome(valor, ...);</code>	

Considerações Finais

O projeto [Entenda Programação de uma vez por todas](#) é um curso gratuito de algoritmos e lógica de programação. Este e-book pode te ajudar a fazer a transição do Fluxograma ou Portugol para C# ou Java, ou mesmo entre essas duas linguagens. Espero, sinceramente, que ele tenha conseguido atingir este objetivo. Se você encontrar algum erro, tiver alguma crítica ou sugestão, por favor, entre em contato.

Além disso, muitas novidades estão por vir. Uma delas é que eu vou avançar nas bibliotecas do Portugol Studio, que permitem acesso a arquivos, tela gráfica, acesso ao mouse, entre outros. Por isso, volto a pedir que você **cadastre o seu e-mail no site** de forma que eu possa te avisar sobre qualquer novidade.

Bons estudos!



www.entendaprogramacao.com.br