

R

Introdução ao
Mundo da
Computação e
Lógica de
Programação

Lógica de Programação

Lógica de Programação

A lógica de programação é uma técnica de desenvolvimento de sequências lógicas, com o objetivo de alcançar um determinado resultado. Desta forma, a lógica de programação é como se escreve um algoritmo.

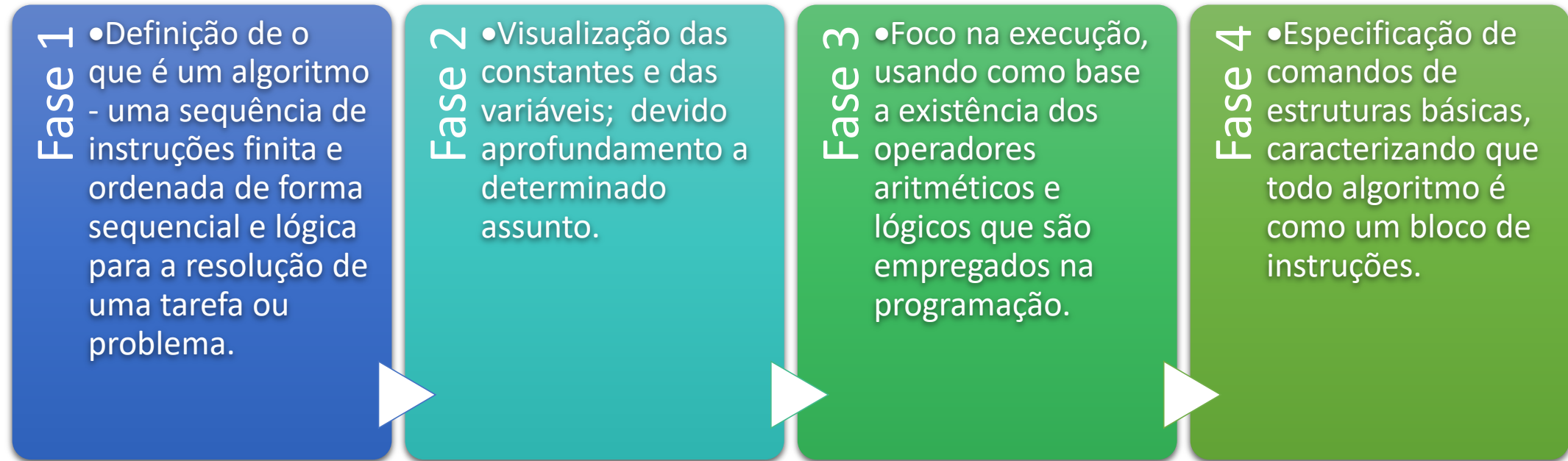
Algoritmo

É algoritmo é uma sequência de etapas para se executar uma função, ou seja, um conjunto de passos finitos e organizados que, quando executados, resolvem um determinado problema. Importante destacar que as suas etapas sempre possuem um início e um fim.



Fases da Lógica de Programação

A referida metodologia possui 4 fases:



Fluxograma

Criado com o objetivo de eliminar as ambiguidades dos algoritmos, um fluxograma é constituído de símbolos gráficos, onde cada símbolo é representado por uma forma geométrica que implica uma ação específica, instrução ou comando a ser realizado. Podemos verificar no diagrama a seguir que há uma denotação de um cenário intermediário à descrição narrativa e ao pseudocódigo (Portugol).

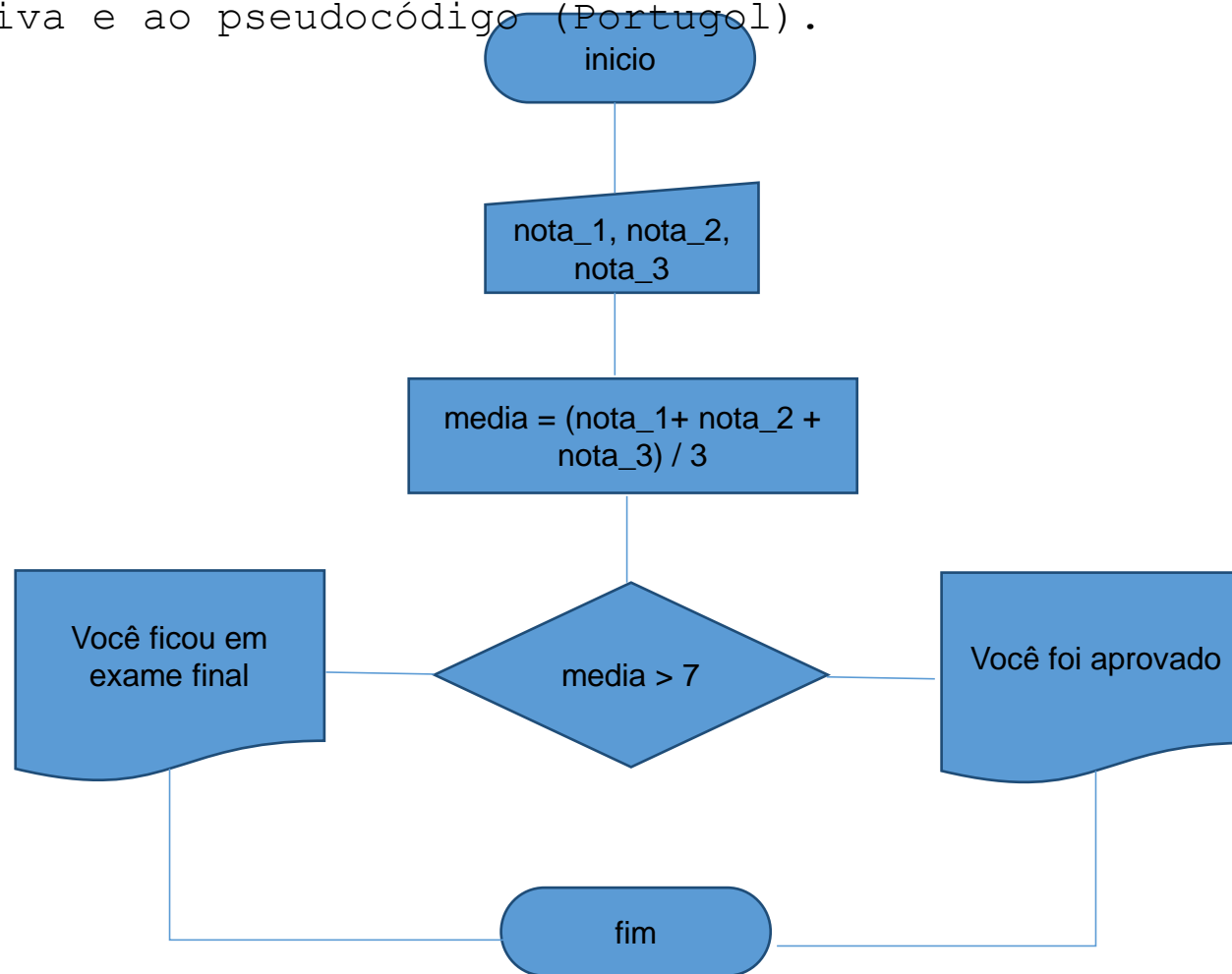
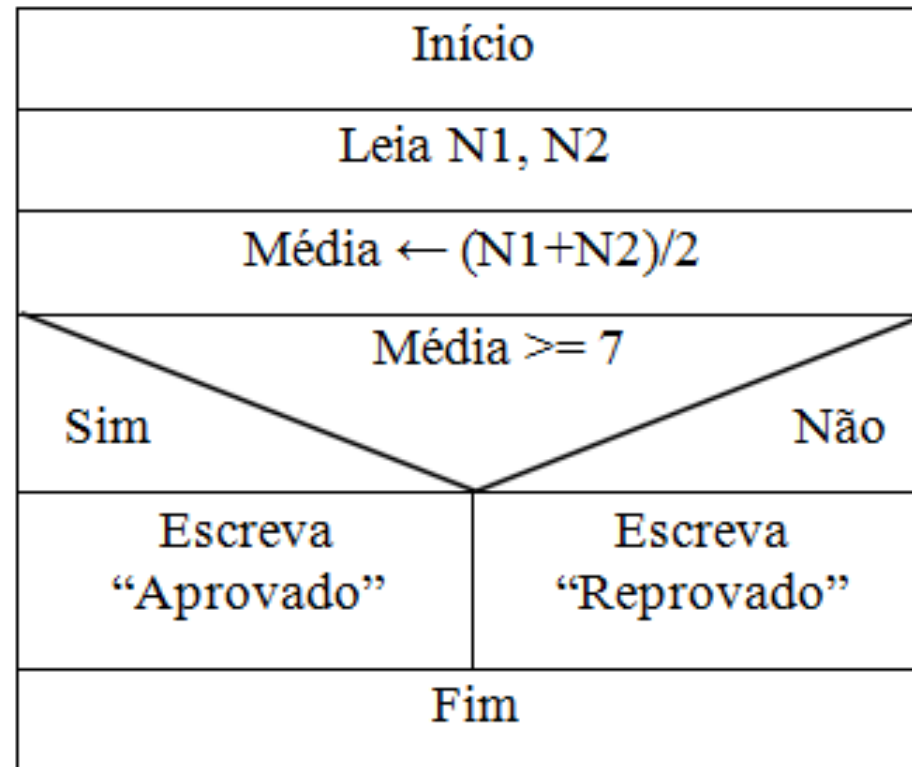


Diagrama de Chapin

Também conhecido como **diagrama Chapin**, o **diagrama** de Nassi-Shneiderman é uma ferramenta de representação gráfica que oferece grande clareza na representação de sequenciação, seleção e repetição de um algoritmo



Teste de Mesa

São mais usados para propósitos didáticos, ou quando não se dispõe de um computador enquanto se está criando um algoritmo e deseja-se testar o algoritmo, geralmente com valores de input diferentes. Então, fazendo o teste de mesa, fica mais fácil entender o que o algoritmo faz.

| Desk Check | | | |
|-------------|-----------|----------------|----------------|
| Line Number | x | Conditions | Input / Output |
| 1 | | | |
| 2 | | | |
| 3 | 1 | | |
| 4 | | 1 < 3 is True | |
| 5 | | | Hello |
| 6 | 1 + 1 = 2 | | |
| 4 | | 2 < 2 is True | |
| 5 | | | Hello |
| 6 | 2 + 1 = 3 | | |
| 4 | | 3 < 3 is False | |
| 8 | | | |

A importância da estruturação do raciocínio lógico

Neste vídeo será mostrado como uma falta de ordenação ou de linha de raciocínio pode fazer com algo possa ser desenvolvido de várias formas.

Vídeo: <https://www.youtube.com/watch?v=pdhqwbUWf4U>



Variáveis e Constantes

Variáveis

Caracterizadas como informações que serão armazenadas em uma variável e alocadas em um local de memória definido pelo próprio programa.

Constantes

Informações que serão armazenadas em uma variável e alocada em um determinado espaço de memória - contudo, a informação **NÃO** mudará durante a execução do programa.

Operador de Atribuição

O operador de atribuição é muito simples e serve para atribuir um valor a uma variável (em qualquer momento), ou uma constante no momento de sua criação.

| Símbolo | Função |
|---------|----------------|
| = | Atribuir Valor |

Exemplos:

```
X = 3;
```

```
T = 25;
```

Operadores Aritméticos

| Aritmético | Operação | Prioridade |
|------------|------------------|------------|
| + | Adição | 3 |
| - | Subtração | 3 |
| % | Resto da divisão | 2 |
| * | Multiplicação | 1 |
| / | Divisão | 1 |

Operadores Aritméticos

Exemplos:

`A = 3 + 5;`

`B = 7 - 4;`

`C = 2 * 3;`

`D = 5 / 1;`

Expressões:

`X = 3 + 4 * 2 + 1;`

Ou

`X = 3 + (4 * 2) + 1;`

Importante o uso dos parênteses nos cálculos aritméticos.



Operadores Relacionais

| Símbolo | Função |
|-----------------------------------|-----------------------------------------|
| <code>!=</code> | Diferente |
| <code>></code> | Maior |
| <code><</code> | Menor |
| <code><=</code> | Menor igual |
| <code>>=</code> | Maior igual |
| <code>==</code> | Igual |
| <code>%</code> | Resto da divisão entre números inteiros |
| <code>x = (a > b) ? a:b</code> | Ternário |

Linguagem algorítmica (pseudocódigo ou portugol)

A linguagem algorítmica foi criada com o intuito de sanar as deficiências das outras representações.

É caracterizada como uma “pseudolinguagem de programação”, cujos comandos são escritos em português, mas já estão denotados em uma estrutura de uma linguagem de programação bem estruturada.

A pseudolinguagem se parece com um programa, ou seja, com é escrito em uma determinada linguagem.



Pseudocódigo

Escrever com pseudocódigo requer que você conheça alguns comandos básicos.

algoritmo = comando para indicar o início do programa;

var = comando para declarar variáveis;

inicio = comando para iniciar o programa principal;

escreva (" ") = comando usado para imprimir uma mensagem na tela;

leia () = comando usado para ler valores digitados no teclado;

<- = comando de atribuição;



Pseudocódigo

Tipos de dados utilizados:

inteiro = Valor numérico sem casa decimal;

real = Valor numérico com casa decimal;

caractere = Entrada de dados com letra;

Pseudocódigo

```
algoritmo  
  
var  
    n1, n2 : inteiro  
  
inicio  
    n1 <- 5  
    n2 <- 9  
  
fimalgoritmo
```

- linha 01: define o início do algoritmo;
- linha 02: definição de variáveis. Aqui, as variáveis n1 e n2 do tipo inteiro são criadas para, posteriormente, armazenar os valores que serão inseridos;
- linhas 03 e 04: atribuição de valores com o operador <- ;
- linha 05: aqui, informamos o fim do nosso algoritmo.

Pseudocódigo

```
algoritmo "soma"

var
    n1, n2, result :

inicio
    leia (n1)
    leia (n2)
    result <- n1 +
    escreva (result)

finalgoritmo
```

- linha 01: define o início do algoritmo;
- linha 02: definição de variáveis. Aqui, as variáveis n1 e n2 do tipo inteiro são criadas para, posteriormente, armazenar os valores que serão inseridos pelo usuário nas linhas 3 e 4. Além disso, é criada a variável que receberá a soma deles, que decidimos chamar de result;
- linhas 03 e 04: leitura de variáveis. Aqui, o código lê os valores informados pelo usuário e que serão armazenados nas variáveis n1 e n2. Essa é a entrada de dados do algoritmo;
- linha 05: atribuição de valores com o operador <-. Esse operador define que o resultado da soma $n1 + n2$ seja armazenado na variável result, que definimos na linha 2. É aqui que acontece o processamento dos dados;
- linha 06: saída de dados do algoritmo. Após realizar a operação de soma, o resultado só vai aparecer se isso for ordenado, e é isso que fazemos na linha 6. No Portugol, utilizamos o verbo "escreva" seguido da variável que queremos exibir, entre parênteses;

- linha 07: aqui, informamos o fim do nosso algoritmo



Pseudocódigo

```
1  algoritmo "Calcmedia"
2  var
3      nota_1, nota_2, nota_3, media :
4  inicio
5      leia (nota_1, nota_2, nota_3)
6      media ← (nota_1 + nota_2 +
7      se media >= 7 então
8          escreva("Você foi aprovado")
9      senão
10         escreva("Você ficou de
11     fimse
12 fimalgoritmo
```

VisuAlg

Hora de praticar!

Download:

<https://sourceforge.net/projects/visualg30/files/latest/download>



Exercício Básico de Pseudocódigo com o VisuAlg

- 1) Escreva um **algoritmo** para que seja dado boas-vindas para uma determinada pessoa. É preciso fazer a atribuição de um nome. Fazer usando o operador \leftarrow .
- 2) Escreva um algoritmo para ler duas variáveis numéricas e fazer a subtração dos valores e por ultimo, exibir o resultado da subtração;
- 3) Escreva um algoritmo que leia possa fazer a resolução da seguinte expressão matemática: $x = (a + (b * c))$. Exibir o resultado da expressão.
- 4) Escreva um algoritmo que leia o nome de uma pessoa, leia uma nota, leia uma segunda nota, faça a média aritmética das duas notas e exiba uma mensagem com a referida media. Cálculo da média: $md = (n1 + n2) / 2$.



Exercício Avançado de Pseudocódigo com o VisuAlg

- 1) Escreva um **algoritmo** que armazene o valor 10 em uma variável A e o valor 20 em uma variável B. A seguir (utilizando apenas atribuições entre variáveis) troque os seus conteúdos fazendo com que o valor que está em A passe para B e vice-versa. Ao final, escrever os valores que ficaram armazenados nas variáveis.
- 2) Escreva um algoritmo para ler as dimensões de um triângulo (base e altura), calcular e escrever a área do triângulo. Sabendo que para calcular a área devemos usar a fórmula a seguir: $area = (base * altura) / 2;$
- 3) Escreva um algoritmo que leia a idade de uma pessoa expressa em anos, meses e dias e escreva a idade dessa pessoa expressa apenas em dias. Considerar ano com 365 dias e mês com 30 dias.



Gabarito Exercício 1

Algoritmo "Troca de Valores"

Var a, b, intermediario: inteiro // crie duas variáveis do tipo inteiro.

Início // Seção de Comandos, procedimento, funções, operadores, etc...

escreva ("Olá, digite um valor para A:")

leia (a)

escreva ("Olá de novo, digite agora outro valor para B:")

leia (b) // Exibe para o usuário os valores digitados.

escreva ("Muito bem! Você digitou os valores para A:", a, " e B:", b) //

Fazemos a troca dos valores das variáveis, // precisamos de uma variável intermediária para fazer a troca

intermediario <- a

a <- b

b <- intermediario

escreva ("Se invertermos os valores de A e B teremos A:", a, " e B:", b)

Fimalgoritmo



Gabarito Exercício 2

Algoritmo "Triângulo"

Var base, altura, area: real // crie duas variáveis do tipo inteiro.

Inicio // Seção de Comandos, procedimento, funções, operadores, etc...

escreva ("***** TRIANGULO *****")

escreval () // Linha em branco

escreva ("Olá, informe a base do triângulo:")

leia (base)

escreva ("Olá, informe a altura do triângulo:")

leia (altura)

area <- (base * altura) / 2

escreva ("Esse triângulo tem uma área de ", area)

Fimalgoritmo

Gabarito Exercício 3

```
Algoritmo "IDADE"
```

```
Var diaNascimento, mesNascimento, anoNascimento, calculaAno, calculaMes, calculaDias: inteiro
```

```
Inicio // Seção de Comandos, procedimento, funções, operadores, etc...
```

```
Escreval ("***** IDADE ***** ")
```

```
Escreval ("Digite o dia em que nasceu: ")
```

```
Leia (diaNascimento)
```

```
Escreval ("Digite o mes em que nasceu (digite o mês em número, de 1 à 12): ")
```

```
Leia (mesNascimento)
```

```
Escreval ("Digite o ano em que nasceu: ")
```

```
Leia (anoNascimento)
```

```
Escreval ("***** ")
```

```
calculaAno <- (2017 - anoNascimento) * 365
```

```
calculaMes <- (mesNascimento * 30) - (30 - diaNascimento)
```

```
calculaDias <- calculaAno + calculaMes
```

```
Escreva ("Você possui", calculaDias, " dias de vida")
```

```
Fimalgoritmo
```





f i t @rederecode | y @recoderede

<https://recode.org.br>