

```
1
2
3  Algoritmo 'Lógica de' {
4
5      [ Programação 1 ]
6
7
8
9
10
11
12 }
13
14
```

Unidade 02

Prof. Daniel Caixeta



```
1  'Sumário' {
2
3      </01> Lógica & Algoritmos [...]
4          < 1.1. Lógica: Conceitos iniciais. 1.2. A lógica na informática [...]. >
5
6      </02> Hardware & Software
7          < 2.1. Em busca por soluções [...]. >
8
9      </03> Algoritmizando
10         < 3.1. O desenvolvimento [...]. 3.2. Conceito de Algoritmo. 3.3. Como
11         construir um algoritmo. 3.4. Tipos de algoritmos. 3.5 Alguns exemplos. >
12
13     Referência
14 }
```

01 {

[Lógica & Algoritmos]

< Conceitos iniciais >

}

1.1. Lógica: Conceitos iniciais

- 1
2
3 • Aristóteles (384 a.C. – 322 a.C.) é considerado o criador da
4 lógica¹. Na época, a denominava “razão”. Esse termo só pas-
5 sou a ser utilizado séculos depois.
- 6 • De acordo com o dicionário Michaelis, lógica é a análise das
7 formas e leis do pensamento, mas não se preocupa com a pro-
8 dução do pensamento, [...], sim com a forma, i.e., como o
9 pensamento é organizado e como chegamos a uma conclusão.
- 10 • Já um argumento ou ideia pode ser composto por uma ou várias
11 premissas², as quais podem ser Verdadeiras ou Falsas.

12
13 1. Do grego *logos* significa linguagem racional.

14 2. Ideia que serve de base para a criação de um pensamento, opinião, ponto de vista, raciocínio;
axioma, pressuposto. (Fonte: <https://www.dicio.com.br/premissa/>)

- 1
2 • No argumento a seguir, temos as premissas 1 e 2, e em 3 a
3 conclusão:

4
5 </1> Sandra é mais velha que Ricardo.

6 </2> Ricardo é mais velho que Pedro.

7 </3> Logo, Sandra é mais velha que Pedro.
8

9
10 </1> Pukki, meu cachorro, tem duas orelhas.

11 </2> Eu, seu dono, tenho duas orelhas.

12 </3> Logo, Eu não sou um cachorro.
13
14

1 • Os argumentos podem ser indutivos ou dedutivos (PUGA & RIS-

2 SETTI, 2013, grifo meu).

3

4 • Os argumentos indutivos:

5 São aqueles que, com base em dados, chega-se a uma resposta

6 por meio de uma analogia, *i.e.*, pela comparação com algo co-

7 nhecido. Esse tipo de raciocínio, contudo, não oferece certe-

8 za de que a resposta será de fato verdadeira. É necessário

9 conhecer os fatos ou as situações para que se possa fazer a

10 comparação (*ibidem*, grifo meu). Por exemplo:

11 </1> Ontem não havia nuvens no céu e não choveu.

12 </2> Hoje não há nuvens no céu.

13 </3> Portanto, hoje não vai chover.

14

1 • Já os argumentos dedutivos:

2 São aqueles cuja conclusão é obtida como consequência das
3 premissas, *i.e.*, por meio da análise das situações ou fatos,
4 pode-se obter a resposta. Trabalha-se com a forma das senten-
5 ças, sem que haja necessidade do conhecimento prévio das si-
6 tuações ou fatos, ou seja, a conclusão é obtida em decorrên-
7 cia das premissas (*ibidem*, grifo meu). Por exemplo:

8 </1> Joana D'Arc era uma mulher.

9 </2> As mulheres são seres dominantes.

10 </3> Logo, Joana D'Arc era um ser dominante.



De modo geral:

</1> A é verdade de B.

</2> B é verdade de C.

</3> Portanto, A é verdade de C.

1.2. A lógica na informática [...]

- 1
2 • Na informática/computação, aplica-se a lógica em pratica-
3 mente todas as áreas. Desde da construção ao pleno funci-
4 onamento de *hardwares* e *softwares*. Principalmente, na cons-
5 trução de circuitos integrados usando conceitos de portas
6 lógicas, i.e., para verificar a passagem ou não de corrente
7 elétrica, estabelecendo comunicação entre os componentes.
- 8 • Já na construção de *software*, por meio do raciocínio lógi-
9 co, construímos algoritmos que podem ser transformados em
10 programas capazes de solucionar problemas dos mais simples
11 aos mais complexos.

12
13
14

02 {

[*Hardware & Software*]

< 0 físico e o lógico >

}

2.1. Em busca por soluções [...]

- Desde o advento da era tecnológica, a busca por soluções que simplifiquem tarefas e otimizem o tempo tem sido uma constante na história humana. Nesse contexto, o computador destacou-se como uma ferramenta excepcionalmente versátil, rápida e segura.
- Portanto, a função essencial é processar dados, que envolve receber, manipular e armazenar informações. Isso é alcançado por meio de programas projetados especificamente para executar tarefas em processamento de dados.

Coleta de dados



Processamento de
informações



Transmissão



Armazenagem
(*data storage*)



03 {

[Algoritmizando]

< Análises descritivas e Códigos >

}

3.1. 0 desenvolvimento [...]

- Quando desenvolvemos um *software*, precisamos escrever códigos que podem ser compostos por várias partes “interligadas”.
- Entretanto, para que o computador compreenda e execute esse programa, devemos escrevê-lo usando uma linguagem que tanto o computador quanto o desenvolvedor entendam. Essa linguagem é chamada linguagem de programação.

```
Welcome C helloworld.c X
C helloworld.c > main()
1  #include <stdio.h>
2
3  int main(){
4      printf("Hello World");
5
6      return 0;
7  }
```



- As etapas para o desenvolvimento de um programa são:

Componentes do 'desenvolvimento' {

- Análise** • Estuda-se o problema para definir os dados de entrada, processamento e dados de saída.
- Algoritmo** • Ferramentas para descrição narrativa ou fluxograma são utilizadas para descrever o problema e possíveis soluções.
- Codificação** • O algoritmo é transformado em códigos da linguagem de programação.

}

- Portanto, um programa é a codificação de um algoritmo em uma linguagem de programação (ASCENCIO, 1999).

3.2. Conceito de algoritmo

```
1  O que é Algoritmo? {
2
3  < Ascencio & Campos (2014), apresentam alguns conceitos
4  de algoritmos. >
5
6  < /1 > Algoritmo é uma sequência finita de instruções ou o-
7  perações cuja execução, em tempo finito, resolve um
8  problema computacional, qualquer que seja sua instân-
9  cia" (SALVETTI, 1999).
10
11  < /2 > É a descrição de uma sequência de passos que deve
12  ser seguida para a realização de uma tarefa"
13  (ASCENCIO, 1999).
14  < /3 > É uma sequência de passos que visa atingir um
    objetivo bem definido" (FORBELLONE, 1999).
}
```

- Por exemplo:

Algoritmo 1 'Trocar uma lâmpada' {

— Passo 1 – Pegar uma lâmpada nova.

— Passo 2 – Pegar uma escada.

— Passo 3 – Posicionar a escada embaixo da lâmpada queimada.

— Passo 4 – Subir na escada com a lâmpada nova na mão.

— Passo 5 – Retirar a lâmpada queimada.

— Passo 6 – Colocar a lâmpada nova.

— Passo 7 – Descer da escada.

— Passo 8 – Testar o interruptor.

— Passo 9 – Guardar a escada.

— Passo 10 – Descartar a lâmpada queimada.

}

```
1  Algoritmo 2 'Sacar dinheiro no banco 24 horas' {
2
3      — Passo 1 – Ir até o banco 24 horas.
4      — Passo 2 – Colocar o cartão.
5      — Passo 3 – Digitar a senha.
6      — Passo 4 – Solicitar a quantia desejada.
7      — Passo 5 – Se o saldo for maior ou igual à quantia desejada sacar;
8                    caso contrário, mostrar mensagem de indisponibilidade de
9                    saque.
10     — Passo 6 – Retirar o cartão.
11     — Passo 7 – Sair do banco 24 horas.
12
13 }
14
```


3.3. Como construir um algoritmo

- 1 • Para construirmos um algoritmo é necessário seguir alguns
- 2 passos:
- 3 </1> Compreender o problema, destacando os pontos mais impor-
- 4 tantes.
- 5
- 6 </2> Definir os dados de entrada.
- 7 </3> Definir o processamento³.
- 8 </4> Definir os dados de saída.
- 9 </5> Construir o algoritmo utilizando um dos tipos descritos
- 10 (próxima seção).
- 11 </6> Testar o algoritmo realizando simulações.
- 12
- 13
- 14 3. O processamento é responsável pela transformação dos dados de entrada em dados de saída.

3.4. Tipos de algoritmos

- 1 • Os três tipos mais utilizados são:

2 Descrição narrativa </1> {

3
4 < Consiste em analisar o enunciado do problema e escrever, u-
5 tilizando uma linguagem natural (*e.g.*, Português), os passos a
6 serem seguidos para a sua resolução. >

7 Vantagem

- 8 • Não é necessário aprender nenhum novo conceito.

9 Desvantagem

- 10 • A linguagem natural abre espaços para várias interpre-
11 tações, o que pode dificultar a transcrição das infor-
12 mações para o algoritmo.

13
14 }

Fluxograma </2> {

< Consiste em analisar o enunciado do problema e escrever, utilizando símbolos gráficos, predefinindo os passos a serem seguidos para sua resolução. >

Vantagem

- O entendimento de elementos gráficos é mais simples que o entendimento de textos.


Desvantagem

- Torna-se necessário aprender a simbologia, além disso, o algoritmo resultante não apresenta muitos detalhes, dificultando sua transcrição para um programa.


}

- 1 • Abaixo, os símbolos fundamentais mais utilizados na constru-
2 ção de um fluxograma.


3

4  • Indica o início e o fim do algoritmo.


5

6  • Indica o sentido do fluxo de dados.
• Conecta exclusivamente os símbolos ou blocos existentes.


7

8  • Representa a entrada de dados.

9

10  • Representa a saída de dados.

11

12  • Indica tomada de decisão.
• Aponta possibilidade de desvios.

13

14  • Indica o sentido do fluxo de dados.
• Serve exclusivamente para conectar os símbolos ou blocos existentes.

15

Pseudocódigo ou Portugol </3> {

< Consiste em analisar o enunciado do problema e escrever, por meio de regras predefinidas, os passos a serem seguidos para sua resolução. >

Vantagem

- A passagem do algoritmo para qualquer linguagem de programação é quase imediata, bastando conhecer as palavras reservadas da linguagem.

Desvantagem

- É necessário aprender as regras do pseudocódigo.

}

3.5. Alguns exemplos

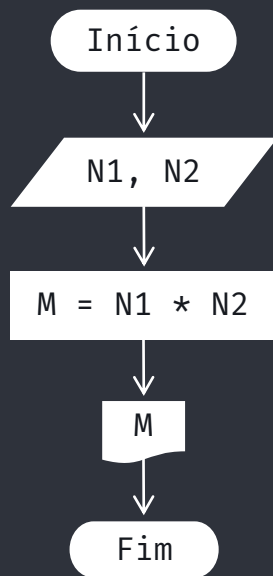
- 1 • A seguir, alguns algoritmos desenvolvidos a partir dos três
2 tipos citados anteriormente.

3
4 </1> Faça um algoritmo que mostre o resultado da multipli-
5 cação de dois números.
6

7
8 Algoritmo em descrição narrativa:

9 Exemplo 1 'Multiplicar dois números' {
10 |
11 | — Passo 1 – Receber dois números que serão multiplicados.
12 | — Passo 2 – Multiplicar os números.
13 | — Passo 3 – Mostrar o resultado obtido na multiplicação.
14 | }

Fluxograma



Pseudocódigo

ALGORITMO

DECLARE N1, N2, M NUMÉRICO

ESCREVA "Digite dois números"

LEIA N1, N2

$M \leftarrow N1 * N2$

ESCREVA "Multiplicação = ", M

FIM_ALGORITMO.

1 </2> Faça um algoritmo que calcule a média aritmética entre
2 duas notas de um aluno e mostre sua situação, que pode ser
3 aprovado ou reprovado.
4

5
6 Algoritmo em descrição narrativa:

7 Exemplo 2 'Calcular média de duas notas [...]' {

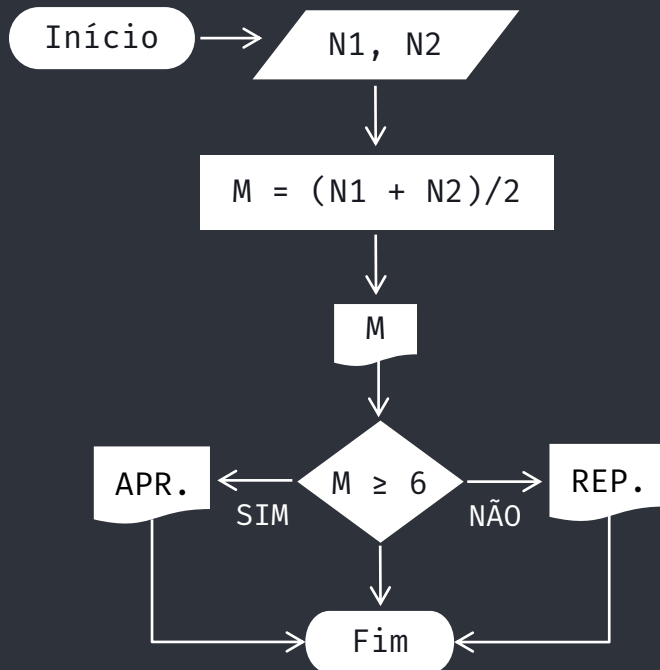
8 — Passo 1 – Receber as duas notas.

9 — Passo 2 – Calcular a média aritmética.

10 — Passo 3 – Mostrar a média aritmética.

11 — Passo 4 – SE a média aritmética for ≥ 6 , ENTÃO aluno(a)
12 aprovado; SENÃO, aluno reprovado.
13 }
14

Fluxograma



Pseudocódigo

ALGORITMO

DECLARE N1, N2, M NUMÉRICO

ESCREVA "Digite as duas notas"

LEIA N1, N2

$M \leftarrow (N1 + N2)/2$

ESCREVA "Média =", M

SE $M \geq 6$

ENTÃO EScreva "Aprovado"

SENÃO EScreva "Reprovado"

FIM_ALGORITMO.

1 </3> Faça um algoritmo para calcular o novo salário de um
2 funcionário. Sabe-se que os funcionários que recebem atual-
3 mente salário de até R\$ 500 terão aumento de 20%; os demais
4 terão aumento de 10%.

5
6 Algoritmo em descrição narrativa:

7
8 Exemplo 3 'Calcular novo salário' {

9 — Passo 1 – Receber o salário atual do funcionário.

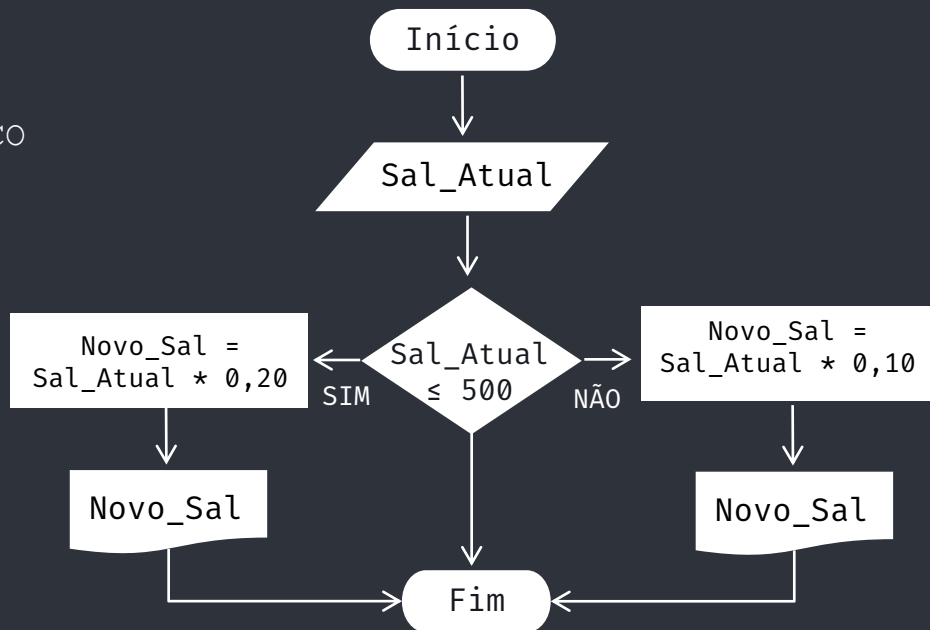
10
11 — Passo 2 – Se o salário atual do funcionário for de até R\$ 500,
12 calcular o novo salário com percentual de aumento de
13 20%; caso contrário, calcular o novo salário com per-
14 centual de aumento de 10%.

}

Pseudocódigo

```
1  
2  
3  
4 ALGORITMO  
5 DECLARE SAL_ATUAL, NOVO_SAL NUMÉRICO  
6 ESCREVA "Digite o salário atual do  
7 funcionário"  
8 LEIA SAL_ATUAL  
9 SE SAL_ATUAL ≤ 500  
10 ENTÃO NOVO_SAL = SAL_ATUAL * 0,20  
11 SENÃO NOVO_SAL = SAL_ATUAL * 0,10  
12 ESCREVA "Novo salário =", NOVO_SAL  
13 FIM_ALGORITMO.  
14
```

Fluxograma



1

2

3 </1> ASCENCIO, Ana Fernanda G. ARAÚJO, Grazielle. S. Estruturas de Dados:
4 algoritmos, análise da complexidade e implementações em Java e C/C++. São Paulo:
5 Pearson Prentice Hall, 2010. (Biblioteca virtual).

5

6 </2> PUGA, Sandra. RISSETTE, Gerson. Lógica de Programação e Estrutura de Dados.
7 2. ed. São Paulo? Pearson Prentice Hall, 2009.

7

8

9

10

11

12

13

14

A 'Obrigado' Is {

Algoritmo & Linguagem de Programação 1

|
}

