

Etapa 1 — Introdução a Algoritmos e Pseudocódigo (Pseudocode)

1. O que é um Algoritmo (Algorithm)

Um **algoritmo** é uma sequência **finita e ordenada de instruções** que descrevem **como resolver um problema** ou **executar uma tarefa**.

Ele é o coração da programação: todo programa de computador nada mais é do que a implementação de um algoritmo.

Exemplo do cotidiano:

Fazer café:

1. Ferver a água
2. Colocar o pó de café no filtro
3. Despejar a água quente
4. Servir o café

Mesmo sem perceber, seguimos **algoritmos diariamente**. A diferença é que, na computação, eles precisam ser descritos de forma **precisa e lógica**.

2. Características de um Algoritmo

Um bom algoritmo precisa ter:

1. **Finitude (Finiteness)** → deve terminar após um número limitado de passos.
2. **Definição (Definiteness)** → cada instrução deve ser clara e não ambígua.
3. **Entrada (Input)** → dados que o algoritmo recebe para processar.
4. **Saída (Output)** → resultado gerado após o processamento.
5. **Efetividade (Effectiveness)** → cada passo deve ser possível de executar.

3. Formas de Representação de um Algoritmo

Podemos representar algoritmos de várias maneiras:

- **Linguagem natural** (texto comum)
- **Fluxogramas (Flowcharts)**
- **Pseudocódigo (Pseudocode)**
- **Código fonte (Source code)** em linguagens como Python, C, Java, etc.

O **pseudocódigo** é o mais usado no ensino, pois é uma forma **independente de linguagem** e fácil de compreender.

4. O que é Pseudocódigo (Pseudocode)

O **pseudocódigo** é uma forma estruturada e textual de descrever algoritmos, usando **palavras-chave intuitivas e instruções simples**.

Ele serve como uma ponte entre o pensamento lógico e a programação real.

Estrutura básica:

início

leia A, B

soma \leftarrow A + B

escreva "Resultado:", soma

fim

Observe que ele **se parece com um código**, mas **não é executável** — serve apenas para expressar a lógica.

5. Vantagens do Pseudocódigo

- Fácil de entender mesmo para quem nunca programou.
- Ideal para planejar soluções antes da codificação.
- Foca na lógica, não na sintaxe.
- Facilita a comunicação entre programadores e analistas.
- É uma ferramenta essencial no aprendizado de lógica.

6. Estrutura Geral de um Algoritmo

Um algoritmo em pseudocódigo geralmente segue este formato:

```
algoritmo "nome_do_algoritmo"  
inicio  
    // Declaração de variáveis  
    // Processamento  
    // Saída de dados  
fim
```

Exemplo:

```
algoritmo "calcular_media"  
inicio  
    leia nota1, nota2  
    media ← (nota1 + nota2) / 2  
    escreva "Média =", media  
fim
```

7. Etapas para Construção de um Algoritmo

1. **Compreender o problema** — entenda o que se deseja resolver.
2. **Definir as entradas e saídas** — quais dados entram e quais resultados devem sair.
3. **Planejar a lógica** — pense nos passos necessários para transformar a entrada na saída.
4. **Escrever em pseudocódigo** — descreva as etapas de forma clara e sequencial.
5. **Testar** — use exemplos reais para verificar se o resultado está correto.
6. **Aprimorar** — simplifique e otimize, se possível.

8. Boas Práticas ao Escrever Pseudocódigos

- Use **nomes significativos** para variáveis.
- Utilize **indentação** (recuo) para melhorar a leitura.
- Faça **comentários** com `//` para explicar partes importantes.

- Evite passos muito longos ou confusos.
- Siga uma **estrutura lógica** clara.

Exemplo:

```
inicio
// Ler dois números e calcular a soma
leia x, y
 $z \leftarrow x + y$ 
escreva "Soma =", z
fim
```

9. Exemplo Completo de Algoritmo com Decisão

```
algoritmo "verificar_maioridade"
inicio
leia idade
se idade >= 18 entao
    escreva "Você é maior de idade."
senao
    escreva "Você é menor de idade."
fimse
fim
```

Este exemplo mostra o uso de uma **estrutura condicional (conditional structure)** com **se** e **senao**.

10. Conclusão

Nesta primeira etapa, aprendemos:

- O conceito e a importância dos **algoritmos (algorithms)**;
- O que é e para que serve o **pseudocódigo (pseudocode)**;
- Como construir e representar um algoritmo passo a passo;
- Boas práticas para escrever pseudocódigos claros e funcionais.

A partir daqui, você já está pronto para entender como **dados são armazenados e manipulados** dentro dos algoritmos — tema da **próxima etapa**!



Perguntas de Múltipla Escolha

1. O que é um algoritmo?

- a) Um programa pronto para execução
- b) Um conjunto de passos lógicos e finitos para resolver um problema
- c) Uma linguagem de programação
- d) Um fluxograma colorido

2. O que caracteriza o pseudocódigo?

- a) É executado por qualquer computador
- b) É usado apenas em linguagens de alto nível
- c) É uma forma textual e estruturada de descrever algoritmos
- d) É um tipo de fluxograma

3. Qual das opções abaixo não é uma característica essencial de um algoritmo?

- a) Finitude
- b) Entrada
- c) Aleatoriedade
- d) Saída

4. No pseudocódigo abaixo, qual é a saída se $A = 5$ e $B = 3$?

inicio

 soma $\leftarrow A + B$

 escreva soma

fim

- a) 8
- b) 53
- c) 2
- d) Erro de sintaxe

Gabarito

- 1. b
- 2. c
- 3. c
- 4. a