

Programação no R - Aula 2

Disciplina: Lógica da programação (de computadores) e
análise de dados no R

Prof. Maurício Garcia de Camargo. IO-FURG.

2025-09-05



Algoritmos

Algoritmo é uma série de passos lógicos para resolver um problema.

Exemplos: receita de um bolo, trocar um pneu, somar dois números.

Formas de representação de algoritmos:

1. Textual
2. Fluxograma
3. Linguagem de programação



1. Representação Textual de algoritmos

Tarefa: somar dois números.

Passo a passo:

1. (INPUT) Pegue os dois números (x_1 e x_2)
2. (OPERAÇÃO) Realize a soma ($x_t = x_1 + x_2$)
3. (OUTPUT) Devolva o resultado (x_t)



Dica

Dica: ao escrever em texto, procure usar **verbos no imperativo** (pegue, calcule, devolva) ao nomear inputs/outputs**.

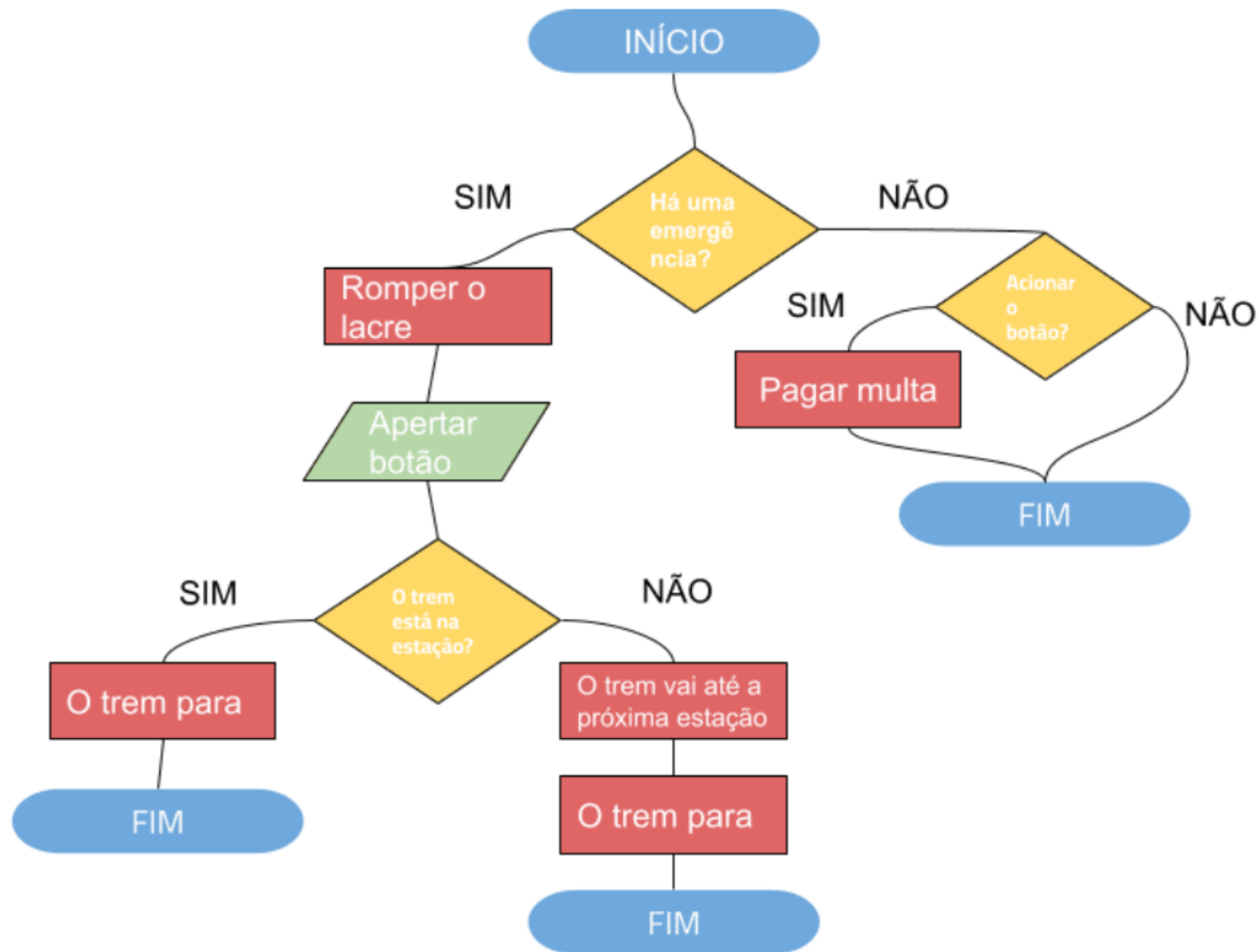


2. Representação de algoritmos por Fluxogramas

O fluxograma do nosso programa possui *entradas* → *operações* → *saída* de forma visual.



Exemplo de fluxograma mais complexo



3. Algoritmos como Linguagens de programação

Pode ser implementado em diferentes linguagens: BASIC, Pascal, Java, Python, R, etc.

A lógica da programação (objetivo desta aula) é universal para todas as linguagens. Mudam apenas a sintaxe e a semântica.

Um algoritmo pode ser implementado de diferentes maneiras com a mesma linguagem, criando assim diferentes programas com os mesmos procedimentos.



Relembrando tipos de dados

- Numérico ($x = 12$): Operações: $+$ $-$ $*$ $/$ $^$
- String ($s = \text{'olá!'}$): Operações: `paste(s1,s2)`
- Lógico (TRUE ou FALSE). Pode ser abreviado (T ou F).
Operações: $<$ $>$ $<=$ $>=$ $==$.



Funções

Funções existem em todas as linguagens de programação e funcionam como um programa inteiro, podendo conter INPUTS, OPERAÇÕES e OUTPUTS.



A estrutura de uma função no R

```
minha_funcao = function(input1, input2) {  
  ... operações ...  
  return(output)  
}
```

- **Minha_funcao** é o nome da função.
- { } representam o início e o fim da função.
- **Function** e **return**: palavras reservadas do R.
- **Operações** são os cálculos que geram o output.



Exemplos de funções no R

Exemplo 1: Função para somar dois números.

```
1 soma2num = function(x1,x2) {  
2   x_tot = x1 + x2  
3   return(x_tot)  
4 }  
5 # Testando a função  
6 soma2num(12,14)
```

Exemplo 2: Função para concatenar duas strings.

```
1 alo_mundo = function(s1,s2) {  
2   s = (paste(s1, s2))  
3   return(s)  
4 }  
5 # Testando a função  
6 alo_mundo('Hello, ', 'world!')
```



Exercícios de funções no R

Exercício 1: Crie uma função que receba dois números e realize a multiplicação entre eles.

Exercício 2: Escreva uma função que receba três números e imprima a média deles.

Exercício 3: Crie uma função que receba dois números (**a** e **b**) e eleve o valor de **a** pelo de **b**.

Exercício 4: Crie uma função que receba o valor do raio e devolva o valor da circunferência de um círculo.



Mais exercícios de funções no R

Exercício 5: Crie uma função que receba o valor do raio e devolva o valor da circunferência e da área de um círculo.

Exercício 6: Crie uma função que receba do usuário a sua altura e o seu peso e mostre na tela o seu Índice de Massa Corporal (IMC):

$$IMC = \frac{Peso}{Altura^2}$$

/images/imc.png



Mais exercícios de funções no R

Exercício 7: Faça um programa que receba do usuário um valor de temperatura em graus Célsius e mostre na tela o valor transformado em graus Fahrenheit:

$$F = \left(C \times \frac{9}{5} \right) + 32$$

Exercício 8: Crie uma função que receba um valor em Dólar e converta o valor para Real, considerando que 1 Dólar = 5,7 reais.



Mais exercícios de funções no R

Exercício 9: A partir da fórmula quadrática, crie uma função que receba os valores de **a**, **b** e **c** e devolva as duas raízes quadráticas.

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$



Estruturas de Condição em R

Estruturas de condição permitem que o programa tome decisões com base em uma verificação lógica.

Em R, a verificação lógica resulta em **TRUE** (verdadeiro) ou **FALSE** (falso).

Exemplo:

```
1 x = 10
2 x > 5
3 # Resultado: TRUE
```



Estrutura básica if

Usada quando queremos executar um bloco de código apenas se uma condição for verdadeira.

Sintaxe:

```
1 if (condição) {  
2     # código a ser executado se a condição for verdadeira  
3 }
```

Exemplo:

```
1 x = 7  
2 if (x > 5) {  
3     print("x é maior que 5")  
4 }
```



Estrutura if...else

Permite executar um código quando a condição é verdadeira e outro quando é falsa.

Sintaxe:

```
1 if (condição) {  
2     # código se verdadeiro  
3 } else {  
4     # código se falso  
5 }
```

Exemplo:

```
1 idade = 16  
2 if (idade >= 18) {  
3     print("Maior de idade")  
4 } else {  
5     print("Menor de idade")  
6 }
```



Estrutura if...else if...else

Permite testar várias condições em sequência.

Sintaxe:

```
1 if (condição1) {  
2     # código se condição1 for verdadeira  
3 } else if (condição2) {  
4     # código se condição2 for verdadeira  
5 } else {  
6     # código se nenhuma condição for verdadeira  
7 }
```

Exemplo:

```
1 nota = 8  
2 if (nota >= 9) {  
3     print("Excelente")  
4 } else if (nota >= 7) {  
5     print("Bom")  
6 } else {
```



Exercícios de *if...else*

Exercício 10: Escreva uma função que receba do usuário um valor numérico e mostre na tela se é positivo ou negativo.

Exercício 11: Escreva um código que leia um número inteiro (variável num) e verifique:

- se é positivo, imprima “Número positivo”
- se é negativo, imprima “Número negativo”
- se for zero, imprima “Número neutro”

Exercício 12: Crie uma função que receba do usuário um valor e mostre na tela se é par ou impar (use o operador %%).



Exercícios de *if...else*

Exercício 13: Crie uma função que receba uma idade e retorne:

“Criança” se menor que 12

“Adolescente” se entre 12 e 17

“Adulto” se entre 18 e 59

“Idoso” se 60 ou mais



Mais exercícios de *if...else*

Exercício 14: Crie uma função que receba o peso e a altura de uma pessoa e determine a sua situação segundo o IMC (veja exercício 7), conforme a tabela abaixo:

IMC	Situação
Abaixo de 18,5	Abaixo do peso
Acima de 18,5 e menor que 25	Peso normal
A partir de 25 e menor que 30	Sobrepeso
Acima de 30 e menor que 35	Obesidade grau 1
Acima de 35 e menor que 40	Obesidade grau 2
Acima de 40	Obesidade grau 3



