

# Método da Bissecção

Prof. Americo Cunha

Universidade do Estado do Rio de Janeiro – UERJ

[americo.cunha@uerj.br](mailto:americo.cunha@uerj.br)

[www.americocunha.org](http://www.americocunha.org)



$$f(x) = 0$$

**É possível garantir que essa equação tem solução?**



$$f(x) = 0$$

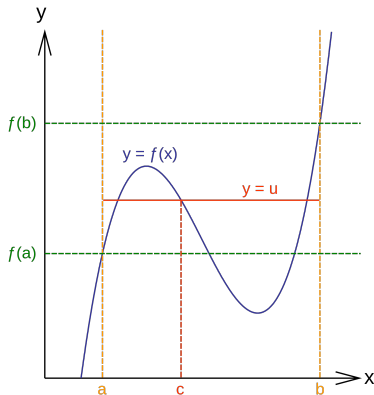
É possível garantir que essa equação tem solução?

Quando  $f$  é contínua, sim!



## Teorema do valor intermediário

Se  $f : [a, b] \subset \mathbb{R} \rightarrow \mathbb{R}$  é uma função contínua, então para qualquer valor  $u$  entre  $f(b)$  e  $f(a)$  existe  $c \in [a, b]$  tal que  $f(c) = u$ .



### Teorema de Bolzano:

Se  $f$  é uma **função contínua** e assume valores com  **sinais contrários**  nos extremos de um intervalo, então  $f$  tem **raiz** no interior desse intervalo.



# Método da bisseção

É um *método numérico* para calcular uma *solução aproximada* para a equação  $f(x) = 0$ , num intervalo  $[a, b]$  onde existe uma solução  $x^*$ .

Hipóteses:

$$f \in C[a, b] \quad \text{e} \quad f(a)f(b) < 0$$

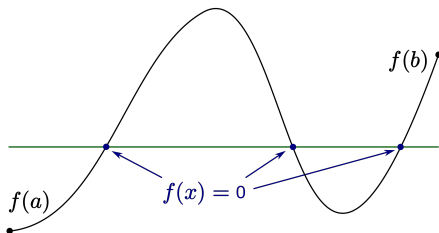



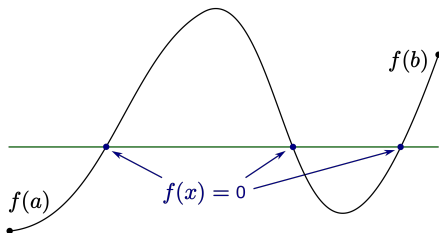
Figura obtida em [https://en.wikipedia.org/wiki/Intermediate\\_value\\_theorem](https://en.wikipedia.org/wiki/Intermediate_value_theorem) 

# Método da bisseção




É um *método numérico* para calcular uma *solução aproximada* para a equação  $f(x) = 0$ , num intervalo  $[a, b]$  onde existe uma solução  $x^*$ .

Hipóteses:

$$f \in C[a, b] \quad \text{e} \quad f(a)f(b) < 0$$



Teorema de Bolzano  $\implies$  existe uma raiz  $x^* \in (a, b)$

Figura obtida em [https://en.wikipedia.org/wiki/Intermediate\\_value\\_theorem](https://en.wikipedia.org/wiki/Intermediate_value_theorem)   

# Método da bisseção

## Ideia do método:

- Calcule o ponto médio de  $[a, b]$ , i.e.,  $x_m = (a + b)/2$
- Com base no sinal de  $f(x_m)$  verifique se

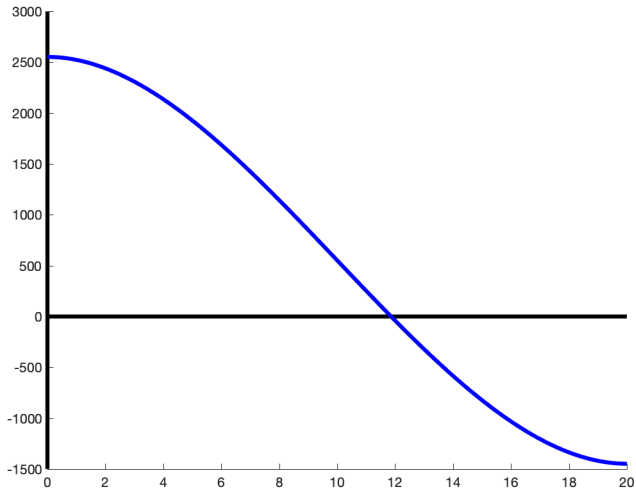
$$x^* \in [a, x_m] \text{ ou } x^* \in [x_m, b]$$

Se  $f(x_m) = 0$  a raiz já foi encontrada!

- Repita o procedimento no subintervalo que contém a raiz, até obter uma aproximação com a tolerância desejada

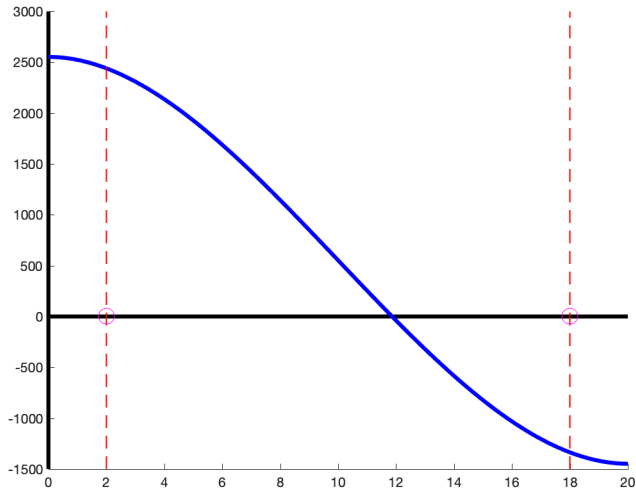


# Método da bisseção

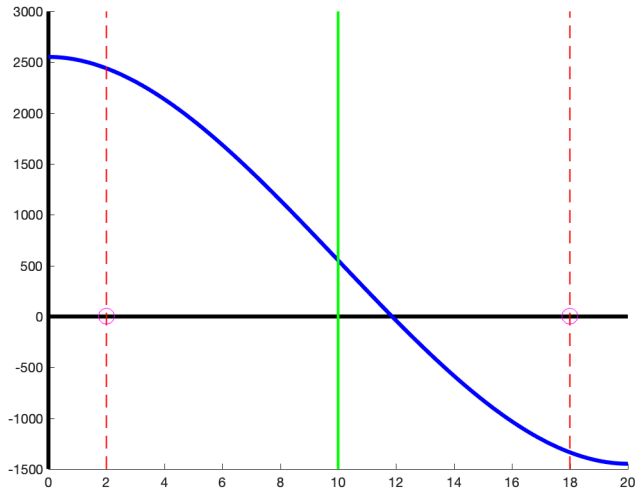




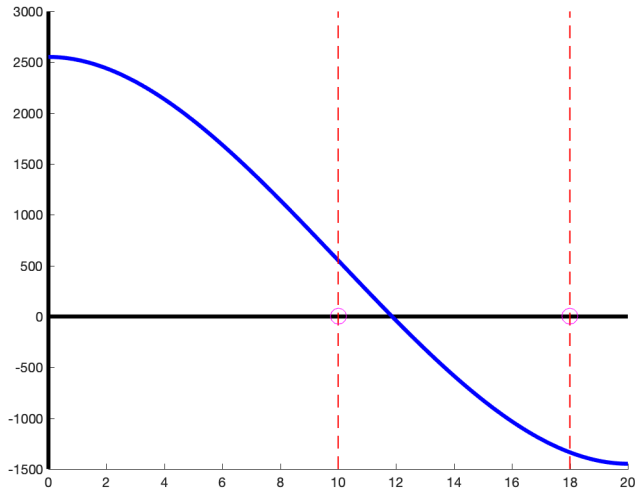
# Método da bisseção



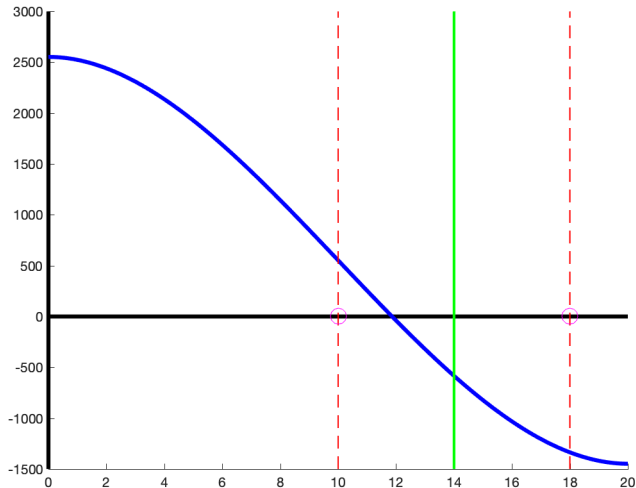
# Método da bisseção



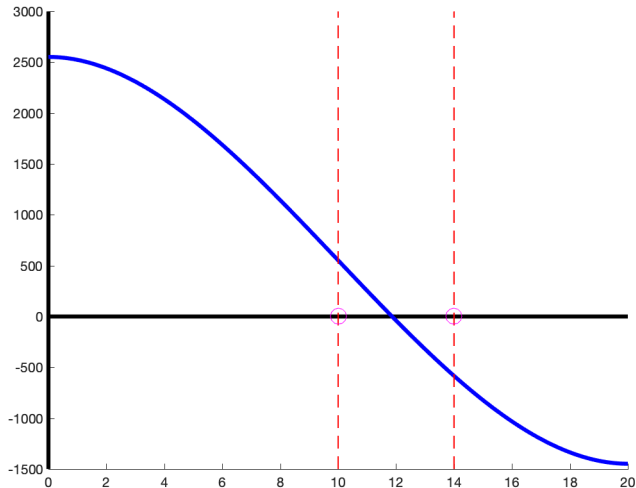
# Método da bisseção



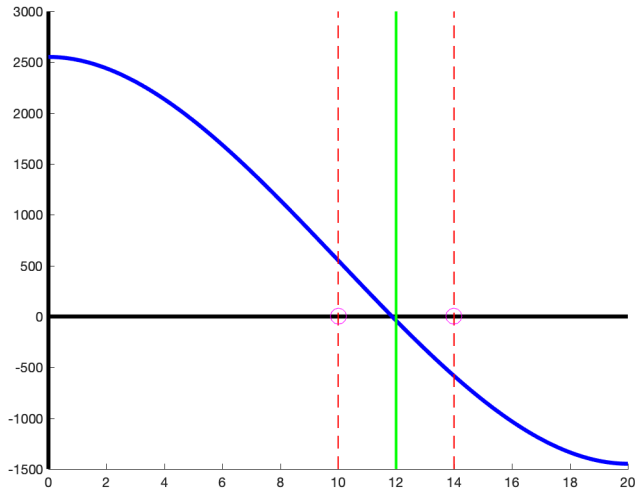
# Método da bisseção



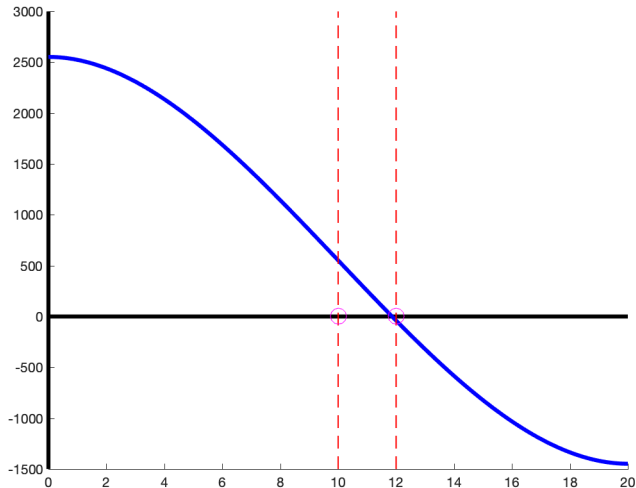
# Método da bisseção



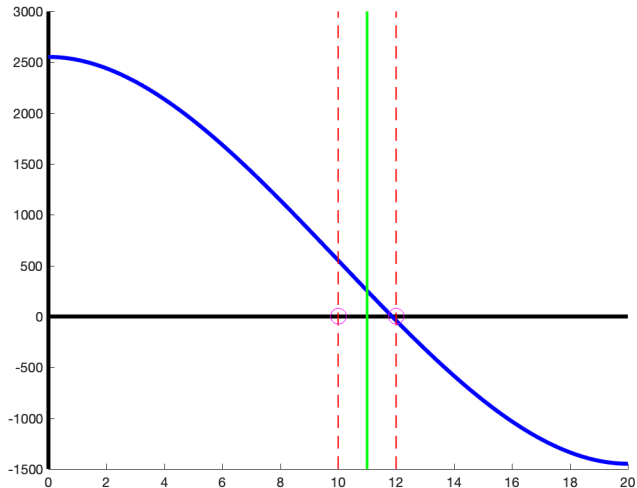
# Método da bisseção



# Método da bisseção

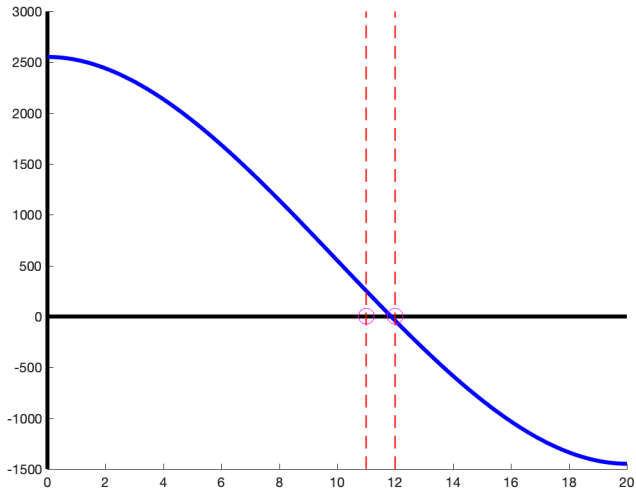


# Método da bisseção

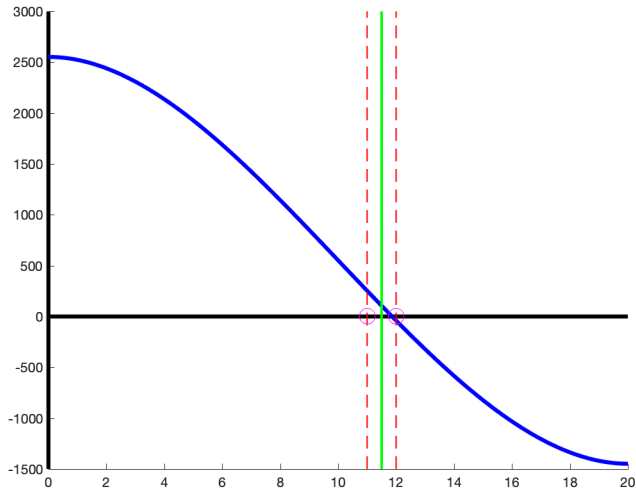




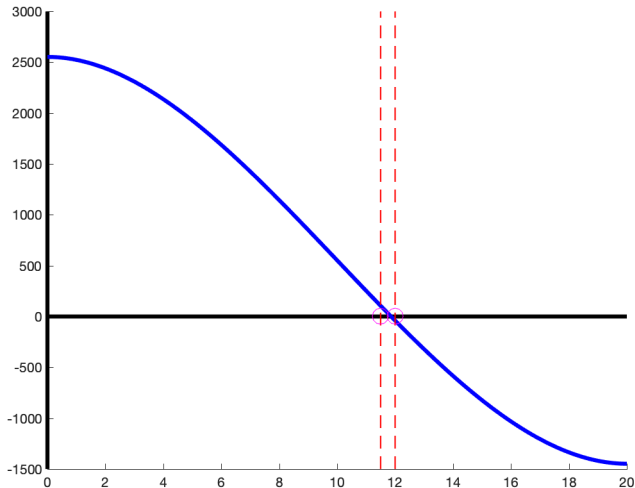
# Método da bisseção



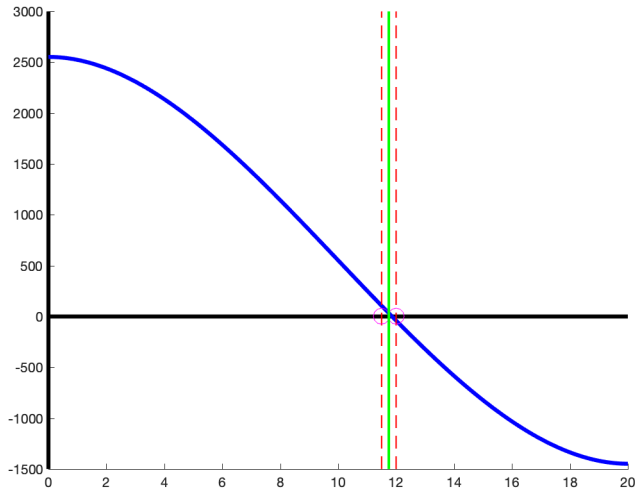
# Método da bisseção



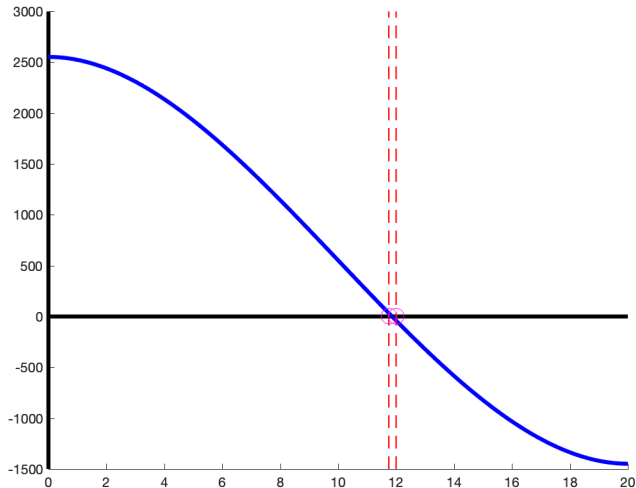
# Método da bisseção



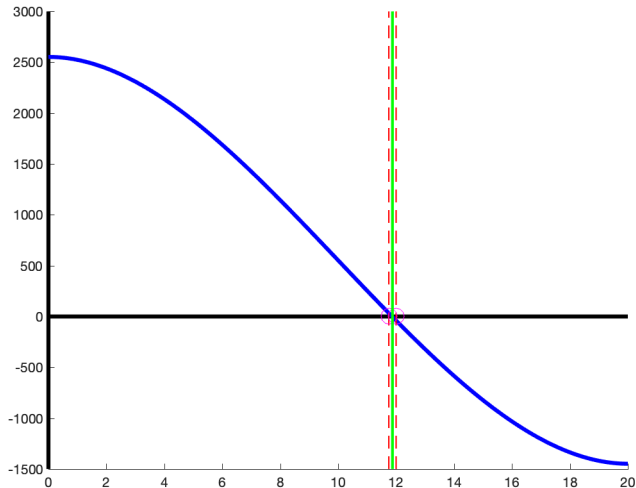
# Método da bisseção



# Método da bisseção



# Método da bisseção



# Algoritmo da bisseção

Input:  $f$ ,  $[a, b]$ , and  $\text{tol}$

```
1: iter = 1
2:  $x_m = (a + b)/2$ 
3: Error =  $|b - a|/2$ 
4: if  $f(x_m) = 0$  or Error < tol then
5:   return
6: end if
7: while Error > tol do
8:   if  $f(a)f(x_m) < 0$  then
9:      $b = x_m$ 
10:  else
11:     $a = x_m$ 
12:  end if
13:   $x_m = (a + b)/2$ 
14:  Error =  $|b - a|/2$ 
15:  iter = iter + 1
16: end while
17: return
```

Output:  $x_m$ , iter



# Implementação em GNU Octave

```
function [xm,iter] = bisection(f,a,b,tol)
    iter = 1;
    xm = 0.5*(a+b);
    Error = 0.5*abs(b-a);
    while Error > tol
        fprintf([' iter = %3d   ',...
                ' root = %.16f ',...
                'Error = %.16f \n'],iter,xm,Error);
        if f(a)*f(xm) < 0
            b = xm;
        else
            a = xm;
        end
        xm = 0.5*(a+b);
        Error = 0.5*abs(b-a);
        iter = iter + 1;
    end
    return
```





# Experimento computacional 1

$$f(x) = x^2 - 2$$

$$[a, b] = [0, 2]$$



```
>> a=0.0; b=2.0; tol=1.0e-9;  
>> f = @(x) x^2-2;  
>> root = bisection(f,a,b,tol);
```



# Experimento computacional 2

$$f(x) = x^3 - 30x^2 + 2552$$

$$[a, b] = [0, 20]$$



```
>> a=0.0; b=20.0; tol=1.0e-9;  
>> f = @(x) x^3 - 30*x^2 + 2552;  
>> root = bisection(f,a,b,tol);
```



# Experimento computacional 3

$$f(x) = \cos(x) \cosh(x) + 1$$

$$[a, b] = [0, 3]$$




```
>> a=0.0; b=3.0; tol=1.0e-9;  
>> f = @(x) cos(x)*cosh(x)+1;  
>> root = bisection(f,a,b,tol);
```

Obs:  $\cosh(x) = (e^x + e^{-x})/2$




# Para pensar em casa ...

## Exercício computacional:

A rotina `bisection.m` possui um *bug* que pode levar à uma falha crítica. Você consegue identificá-lo? 

## Exercício computacional:

Modifique a rotina `bisection.m` para corrigir o *bug* acima e prevenir possíveis erros de entrada do usuário:

- intervalo que não contém raiz;
- tolerância não positiva. 



# Fundamentação teórica

## Teorema (limite para o erro do método da bisseção)

Seja  $f$  uma **função contínua** em  $[a, b]$  que **muda de sinal** no interior desse intervalo, e  $x^*$  uma raiz de  $f$  em  $(a, b)$ .

O **método da bisseção** gera uma **sequência de aproximações**

$$x_0, x_1, \dots, x_n, \dots$$

tal que (para qualquer  $n \in \mathbb{N}$ ):

$$|x^* - x_n| \leq \frac{|b - a|}{2^n}$$

Logo,  $x_n \rightarrow x^*$  quando  $n \rightarrow \infty$ .



# Fundamentação teórica

## Teorema (limite para o erro do método da bisseção)

Seja  $f$  uma **função contínua** em  $[a, b]$  que **muda de sinal** no interior desse intervalo, e  $x^*$  uma raiz de  $f$  em  $(a, b)$ .

O **método da bisseção** gera uma **sequência de aproximações**

$$x_0, x_1, \dots, x_n, \dots$$

tal que (para qualquer  $n \in \mathbb{N}$ ):

$$|x^* - x_n| \leq \frac{|b - a|}{2^n}$$

Logo,  $x_n \rightarrow x^*$  quando  $n \rightarrow \infty$ .

**Pense numa demonstração para esse teorema.**



# Número máximo de iterações até a “convergência”

Para que o método da bisseção produza uma aproximação tal que  $|x^* - x_n| < \text{tol}$  basta escolher  $n$  de forma que

$$\frac{|b - a|}{2^n} < \text{tol}.$$



# Número máximo de iterações até a “convergência”

Para que o método da bisseção produza uma aproximação tal que  $|x^* - x_n| < \text{tol}$  basta escolher  $n$  de forma que

$$\frac{|b - a|}{2^n} < \text{tol}.$$

Assim sendo, o *número máximo de iterações até a “convergência”* é

$$n^* = \text{ceil} \left( \log_2 \left( \frac{|b - a|}{\text{tol}} \right) \right).$$

Obs:  $\text{ceil}(\pi) = 4$ .





# Número máximo de iterações até a “convergência”

Para que o método da bisseção produza uma aproximação tal que  $|x^* - x_n| < \text{tol}$  basta escolher  $n$  de forma que

$$\frac{|b - a|}{2^n} < \text{tol}.$$

Assim sendo, o *número máximo de iterações até a “convergência”* é

$$n^* = \text{ceil} \left( \log_2 \left( \frac{|b - a|}{\text{tol}} \right) \right).$$

Note que esse critério depende apenas de  $[a, b]$  e  $\text{tol}$ , sendo independente do computador utilizado.

Obs:  $\text{ceil}(\pi) = 4$ .



# Características do método da bisseção

- 😊 Simples e fácil de implementar
- 😊 Seguro e robusto (não falha)
- 😊 Convergência garantida (teorema)
- 😊 Requer apenas a continuidade de  $f$
- 😞 Convergência extremamente lenta
- 😞 Difícil de generalizar para várias variáveis



## Como citar esse material?

A. Cunha, *Método da Bisseção*, Universidade do Estado do Rio de Janeiro – UERJ, 2020.

Essas notas de aula podem ser compartilhadas nos termos da licença Creative Commons BY-NC-ND 3.0, com propósitos exclusivamente educacionais.

