



UNIVERSIDADE DE BRASÍLIA

RELATÓRIO DE ATIVIDADE DO MÓDULO 1  
MÉTODOS NUMÉRICOS PARA ENGENHARIA

---

## Zeros de Funções

---

*Aluno:*  
Wilton Rodrigues

*Matrícula:*  
13/0049212

31 de agosto de 2016

## 1 Introdução

O objetivo deste relatório é exercitar os conceitos aprendidos em aula, com relação ao tópico: Zeros de funções. Que tem como objetivo prover métodos matemáticos capazes de determinar o ponto, ou pontos, nos quais a equação cruza ou toca o eixo X, ou seja, um valor numérico que satisfaça à equação. Tarefa que pode dispendir um enorme esforço, ou em alguns casos é até mesmo impossível, em equações que não possuem solução analítica. Como é o caso da Equação de Kepler que é dada por:

$$M = x - E \sin(x) \quad (1)$$

Dado que  $E = 0.2$  e  $M = 0.5$ , o objetivo deste trabalho é obter a raiz da equação (1) com precisão de 10 casas decimais.

Fazendo as devidas substituições e manipulações matemáticas, obtemos a seguinte equação:

$$\begin{aligned} -E \sin(x) + x - M &= 0 \\ -0.2 \sin(x) + x - 0.5 &= 0 \end{aligned} \quad (2)$$

A partir da equação (2), utilizaremos dois passos para encontrar a raiz: O passo 1 tem como objetivo obter um intervalo  $[a, b]$  aproximado no qual  $x \in [a, b]$ . O passo 2 é onde aplicaremos o método numérico da Bissecção para refinar a solução.

## 2 Metodologia

Neste primeiro passo, será feita uma análise da equação, baseando-se no gráfico da mesma e nos conceitos teóricos e teoremas do método escolhido. O sucesso ou falha do próximo passo está diretamente ligado aos resultados obtidos nesta fase de análise. Baseando-se no teorema:

**Teorema 1 (Função Contínua)** *Seja  $f(x)$  uma função contínua num intervalo  $[a, b]$ . Se  $f(a)f(b) < 0$ , então existe pelo menos um ponto  $x = \alpha$  entre  $a$  e  $b$  que é zero de  $f(x)$ .*

E analisando o gráfico da figura 1 é possível perceber que a equação (2) intercepta o eixo X no intervalo  $[0, 1]$

Ao substituírmos estes pontos na equação (2), obtemos os seguintes resultados.

$$\begin{aligned}
 a &= 0, b = 1 \\
 f(a) &= -0.2\sin(0) + 0 - 0.5 \\
 f(a) &= -0.5 \\
 f(b) &= -0.2\sin(1) + 1 - 0.5 \\
 f(b) &= 0.3317058030384207 \\
 f(a) * f(b) &= -0.16585290151921034
 \end{aligned} \tag{3}$$

Então, de acordo com o teorema 1, como o resultado de  $f(a) * f(b) < 0$  de fato há uma raiz entre o intervalo  $[0, 1]$ .

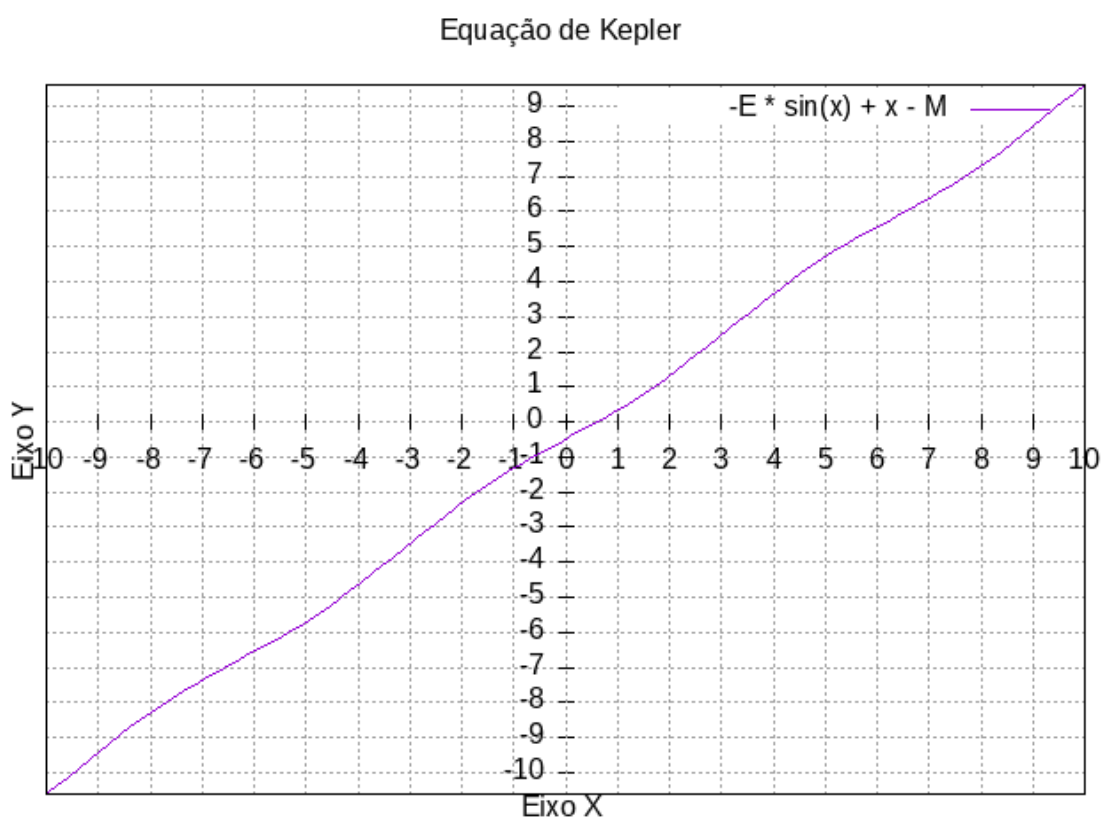


Figura 1: Plotagem da função no intervalo  $[-10, 10]$

### 3 Diagrama esquemático de execução

Nesta seção, encontra-se o fluxo de execução da solução da equação (1) utilizando a linguagem C. Que é apresentada na próxima sessão.

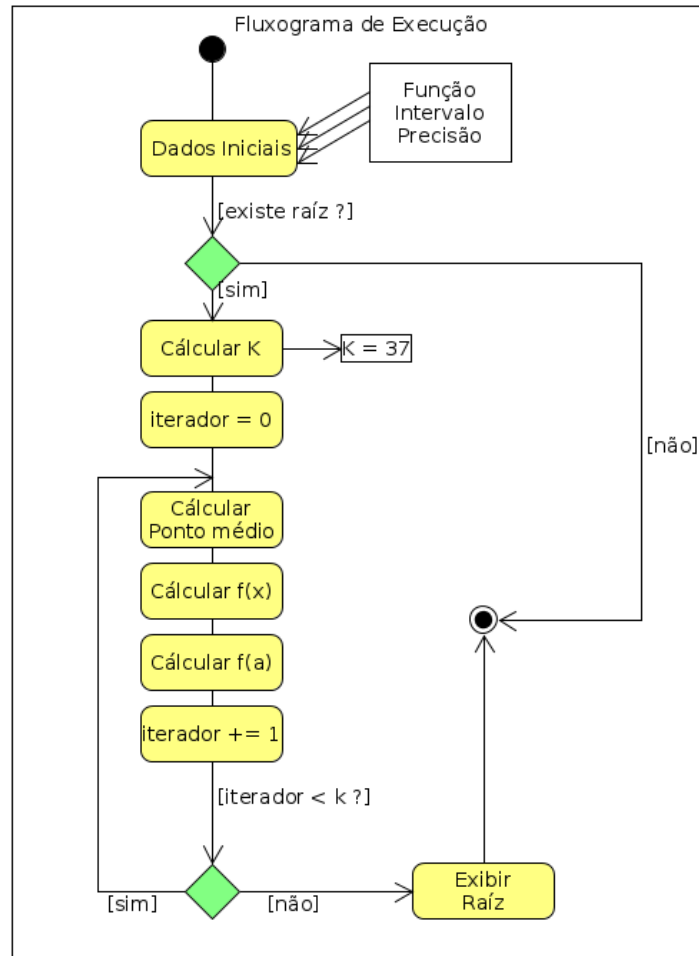


Figura 2: Fluxo de execução da solução

A solução elaborada neste relatório funciona da seguinte maneira. Tanto a função, quanto a precisão desejada são inseridas diretamente no código fonte. Apenas o intervalo no qual se deseja verificar a existencia da raíz é solicitado ao usuário em tempo de execução. Caso o intervalo informado não possua uma raíz, de acordo com teorema 1, uma mensagem é apresentada ao usuário e a execução encerra. Caso o intervalo seja válido a raíz correspondente é apresentada e então o programa se encerra.

## 4 Código Fonte

```
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<math.h>
4 #define precision 0.000000000001
5
6 int calc_k(double a, double b){
7     double k;
8     k = (log(b - a) - log(precision)) / log(2);
9     return (int) ceil(k);
10 }
11
12 double calc_ponto_medio(double a, double b){
13     double x;
14     x = (a + b) / 2;
15     return x;
16 }
17
18 double calc_fx(double x){
19     double fx;
20     fx = -0.2 * sin(x) + x - 0.5;
21     return fx;
22 }
23
24 double calc_fa(double a){
25     double fa;
26     fa = -0.2 * sin(a) + a - 0.5;
27     return fa;
28 }
29
30 int existence(double a, double b){
31     double fa = calc_fx(a);
32     double fb = calc_fx(b);
33     if (fa * fb < 0)
34         return 1;
35     else
36         return 0;
37 }
38
39
40 int main(){
41     double a, b;
42     printf("Digite o primeiro valor do intervalo: \n");
43     scanf("%lf", &a);
44     printf("Digite o segundo valor do intervalo: \n");
45     scanf("%lf", &b);
46 }
```

```
47  if(existence(a, b) == 1){
48      int k = calc_k(a, b);
49      double x = 0.0;
50      double fx = 0.0;
51      double fa = 0.0;
52
53      for(int i = 0; i < k; i++){
54          x = calc_ponto_medio(a, b);
55          fx = calc_fx(x);
56          fa = calc_fa(a);
57          if (fx * fa > 0){
58              a = x;
59          }
60          else{
61              b = x;
62          }
63      }
64
65      printf("%.11lf \n", x);
66  }
67  else{
68      printf("Nao existe raiz neste Ponto\n");
69  }
70  return 0;
71 }
```

## 5 Resultados e discussões

## 6 Ferramentas

Todas as ferramentas utilizadas neste relatório são ferramentas open source (software livre). Permitindo assim que qualquer um possa reproduzir e contestar as afirmações presentes neste documento.

1. Arch Linux (<https://www.archlinux.org>)
  - Sistema operacional utilizado.
2. GCC (<https://gcc.gnu.org>)
  - Compilador de C utilizado para compilar a solução.
3. vim (<http://www.vim.org>)
  - Editor de texto.
4. L<sup>A</sup>T<sub>E</sub>X (<https://www.latex-project.org>)
  - Sistema tipográfico de alta qualidade (utilizado para elaborar o relatório).
5. Gnuplot (<http://www.gnuplot.info>)
  - Utilitário de representação gráfica (utilizado para plotagem do gráfico).
6. UMLet (<http://www.umlet.com>)
  - Ferramenta de UML (utilizado para criar o fluxo de execução).
7. Shutter (<http://shutter-project.org>)
  - Programa de captura de tela (utilizado para capturar os resultados).