

# Introdução à Otimização

Guilherme Silva

Janeiro de 2026

## 1 Slide 1

### 1.1 Exercício 05

**Definição das variáveis:**

- $x$ : hectares de feijão
- $y$ : hectares de arroz

**Função objetivo:**

$$\max f(x, y) = 450x + 1200y \quad (1)$$

**Restrições:**

$$30x + 120y \leq 2400 \quad (2)$$

$$24x + 60y \leq 1380 \quad (3)$$

$$x + y \leq 40 \quad (4)$$

$$y \leq 20 \quad (5)$$

$$x, y \geq 0 \quad (6)$$

## 2 Slide 4

### 2.1 Exercício 03

Código em SageMath utilizando o Método da Secante para encontrar uma raiz:

```
1 # Function
2 var('x')
3 f(x) = x^2+6*x+8
4 f1(x) = x^3-15*x+12
5 f2(x) = x^4-3*x^3+5*x-10
6 f3(x) = exp(2*x)-3-4*sin(x)
7 f4(x) = 3*x^2-2*sin(x)+x-exp(2*x)
8
9 # Get values
10 a = RR(input("Enter lower limit: "))
11 b = RR(input("Enter upper limit: "))
12 tol = RR(input("Enter tolerance: "))
13
14 # Pseudo random points
15 def random_point(a,b):
16     while 1:
17         x = RR.random_element(a, b)
18         y = RR.random_element(a, b)
19         if x!=y:
20             return vector([x,y])
21
22 # Secant Method
23 def secant_method(f,a,b,tol):
24     i = 1000 # Max iterations
25     c = 0
26     j = 0
27     while j<i or abs(b - a)<=tol:
28         if f(x=b)-f(x=a) == 0:
29             res = 0
30             break
31         if f(x=a) == 0:
32             res = f(x=a)
33             break
34         if f(x=b) == 0:
35             res = f(x=b)
36             break
37         c = b - f(x=(b))*(b-a)/(f(x=b)-f(x=a))
38         a = b
39         b = c
40         j+=1
41     return vector([c, j])
42
```

```

43 # Results
44 v = vector(secant_method(f,a,b,tol))
45 print("Function: ", f, "\nApproximate root: ", v[0], "\nIterations: "
      , v[1], "\n")
46
47 v1 = vector(secant_method(f1,a,b,tol))
48 print("Function: ", f1, "\nApproximate root: ", v1[0], "\nIterations: "
      , v1[1], "\n")
49
50 v2 = vector(secant_method(f2,a,b,tol))
51 print("Function: ", f2, "\nApproximate root: ", v2[0], "\nIterations: "
      , v2[1], "\n")
52
53 v3 = vector(secant_method(f3,a,b,tol))
54 print("Function: ", f3, "\nApproximate root: ", v3[0], "\nIterations: "
      , v3[1], "\n")
55
56 v4 = vector(secant_method(f4,a,b,tol))
57 print("Function: ", f4, "\nApproximate root: ", v4[0], "\nIterations: "
      , v4[1], "\n")

```