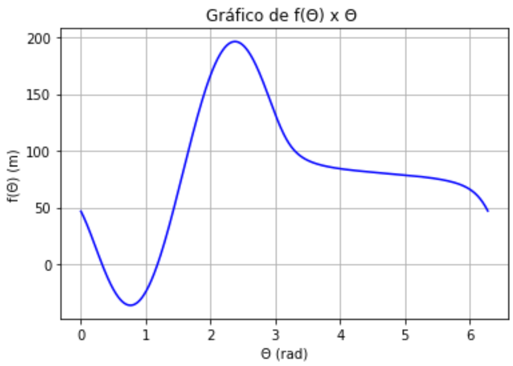
```
import math as m
import matplotlib.pyplot as plt
import numpy as np

def f(x):
    # return x**2 - 4
    return m.sin(x)

def f(x):
    g = 9.98   #[m/s**2]        #convertendo todos para metros.
    y0 = 5     #[m]
    v0 = 120.0 / 3.6 #[m/s]
    x0 = 80     #[m]
    radicando = v0*v0 * m.sin(x)*m.sin(x) + 2*g*y0
    termogrande = v0*m.sin(x) + m.sqrt(radicando)
    return x0 - (v0*m.cos(x)/g) * termogrande


def acha_raiz(f, xinicial, prec):
    passo = 0.5
    x = xinicial
    fold = f(x)

    while abs(passo) > prec :
        x = x + passo
        fnew = f(x)
        if fnew * fold < 0 :
            passo = -passo/2
        fold = fnew
    return x

def yt(t, theta):
    g = 9.98   #[m/s**2]
    y0 = 5    #[m]
    v0 = 120.0 / 3.6 #[m/s]

    return y0 + v0*m.sin(theta)*t - 0.5*g*t*t

def xt(t, theta):
    v0 = 120.0 / 3.6 #[m/s]
    return v0*m.cos(theta)*t

def tempo_queda(theta):
    g = 9.98   #[m/s**2]
    y0 = 5    #[m]
    v0 = 120.0 / 3.6 #[m/s]

    return -1/g * (-v0*m.sin(theta) + m.sqrt(v0*v0*m.sin(theta)**2 + 2*g*y0))

theta_graf = np.arange(0, 2*m.pi, 0.01)

plt.figure()
plt.grid()
func = [f(i) for i in theta_graf]
plt.plot(theta_graf, func, 'b')
plt.xlabel('θ (rad)')
plt.ylabel('f(θ) (m)')
plt.title('Gráfico de f(θ) x θ ')
plt.show()

precisao = 10**(-14)
primeira_raiz = acha_raiz(f, 1, precisao)
segunda_raiz = acha_raiz(f, 0, precisao)
print(primeira_raiz)
print(segunda_raiz)

# Primeira raiz
y = 1; y1 = []; x1 = []

theta = primeira_raiz
tempo1 = tempo_queda(theta)
t = 0
dt = 0.001
while y >= 0:
    y = yt(t, theta)
    x = xt(t, theta)
    y1.append(y)
    x1.append(x)
    t += dt

# Segunda raiz
```

```
y = 1; y2 = []; x2 = []

theta = segunda_raiz
tempo2 = tempo_queda(theta)
t = 0
while y >= 0:
    y = yt(t, theta)
    x = xt(t, theta)
    y2.append(y)
    x2.append(x)
    t += dt


plt.figure()
plt.grid()
plt.plot(x1, y1, 'r',label="θ(RAD)=1.2222198974749716")
plt.plot(x2, y2, 'b', label="θ(RAD)=0.328579095346754")
plt.xlabel('Deslocamento horizontal(m)')
plt.ylabel('Deslocamento vertical(m)')
plt.title('Trajetória do projétil')
plt.legend(loc='center')
plt.show()
```




```
1.1826395963981753
0.3257379204007549
```