

## ESO208A

### ASSIGNMENT 1

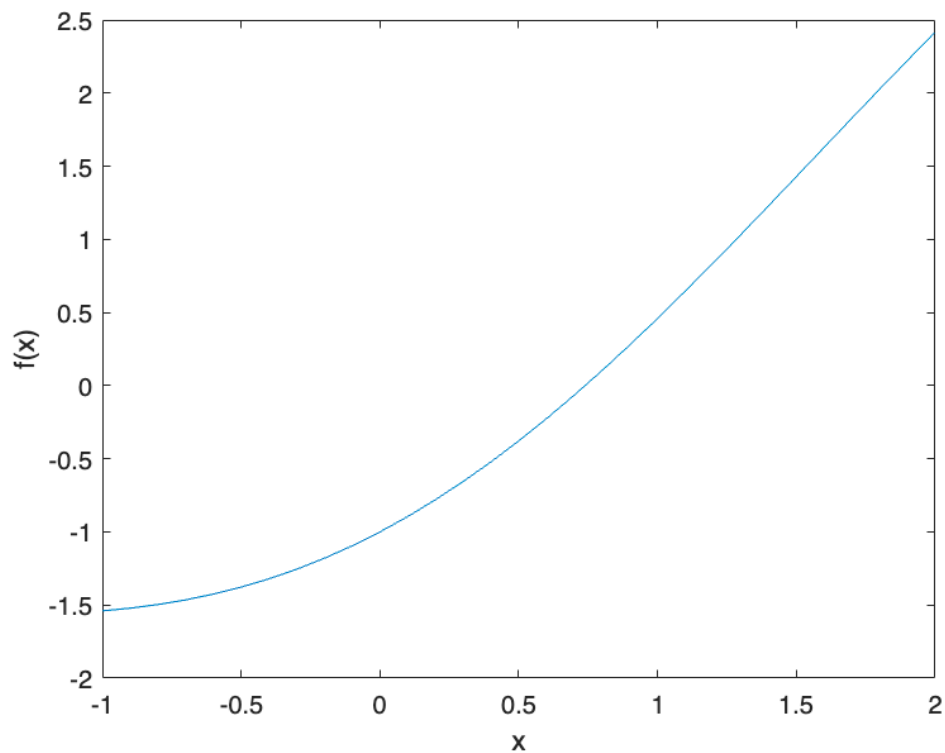
#### QUESTION1

Test functions:

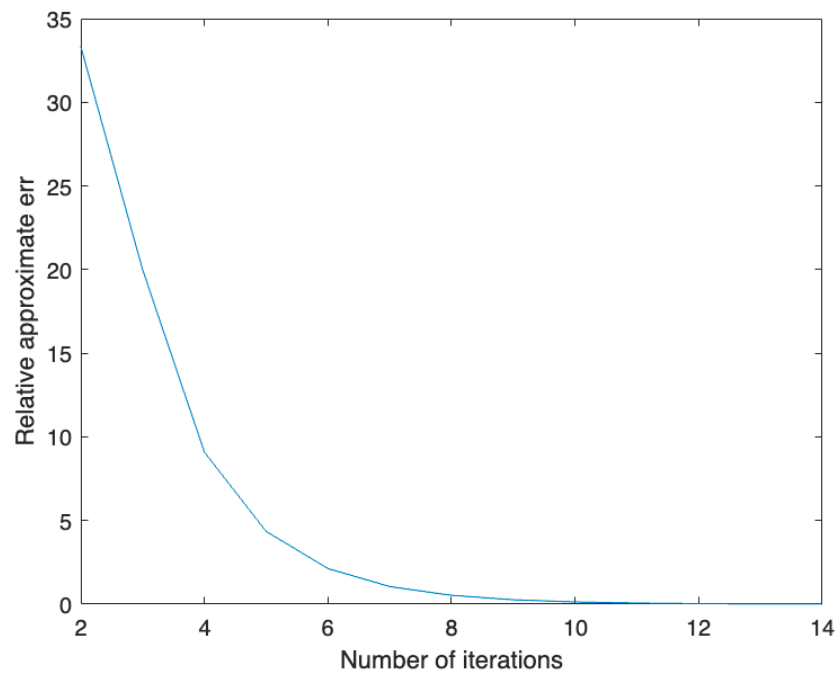
$$f(x) = x - \cos x$$

Use the initial bracket as (0,1) or the initial guess as 0; maximum iterations 50; and maximum  $\varepsilon_r = 0.01\%$ .  
For Fixed-Point method, use  $\phi(x) = \cos x$ .

#### 1. BISECTION

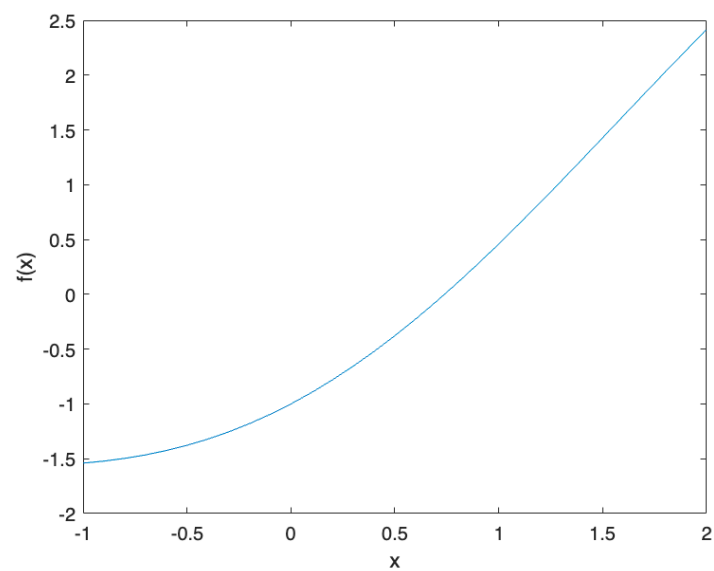


The root of the equation: **0.739075**

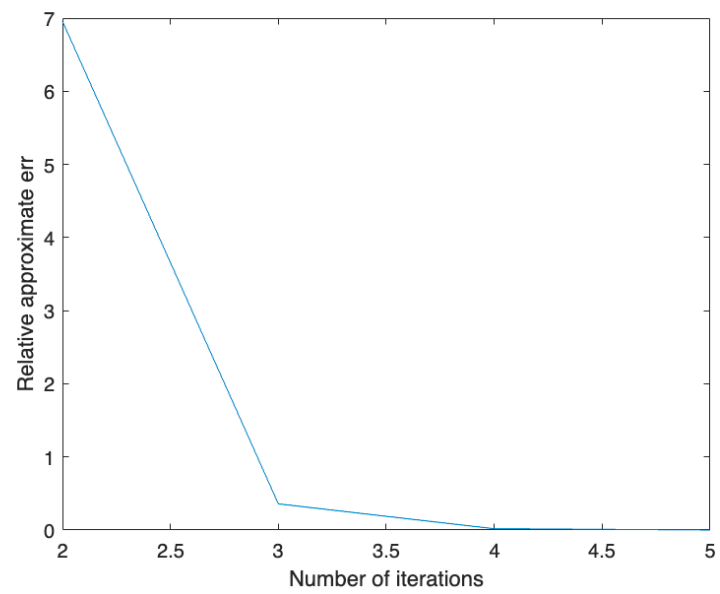


## 2. FALSE-POSITION

MaxErr = **0.0100**

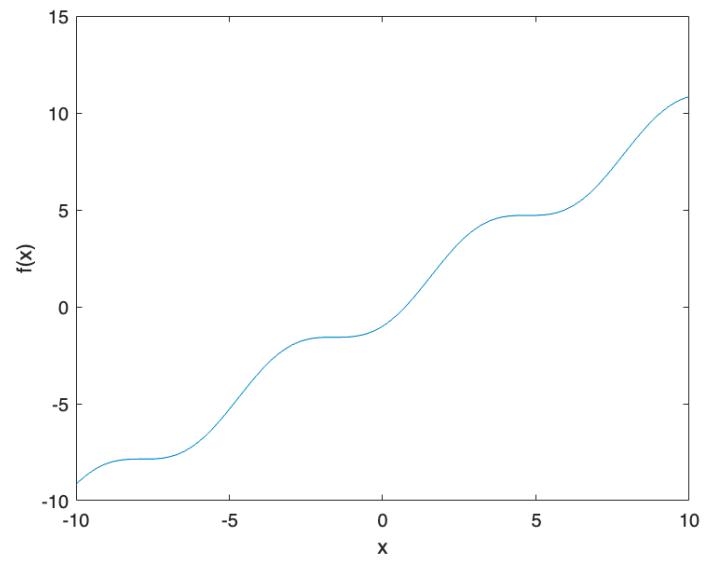


The root of the equation: **0.739085**

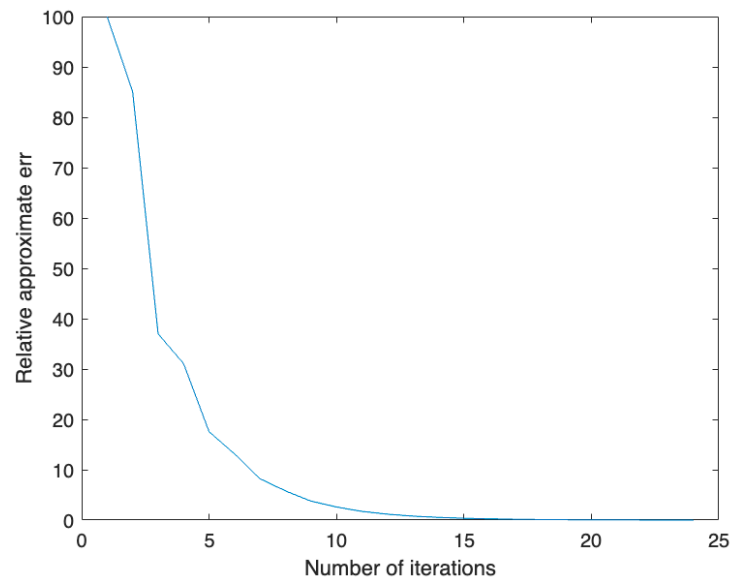


### 3. FIXED POINT

MaxErr = **0.0100**

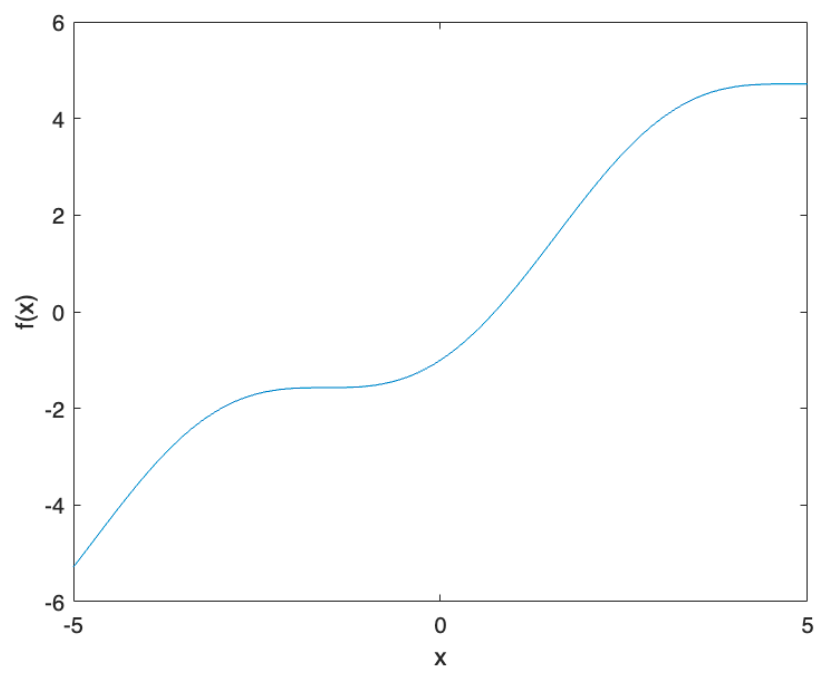


The root of the equation: **0.739106**

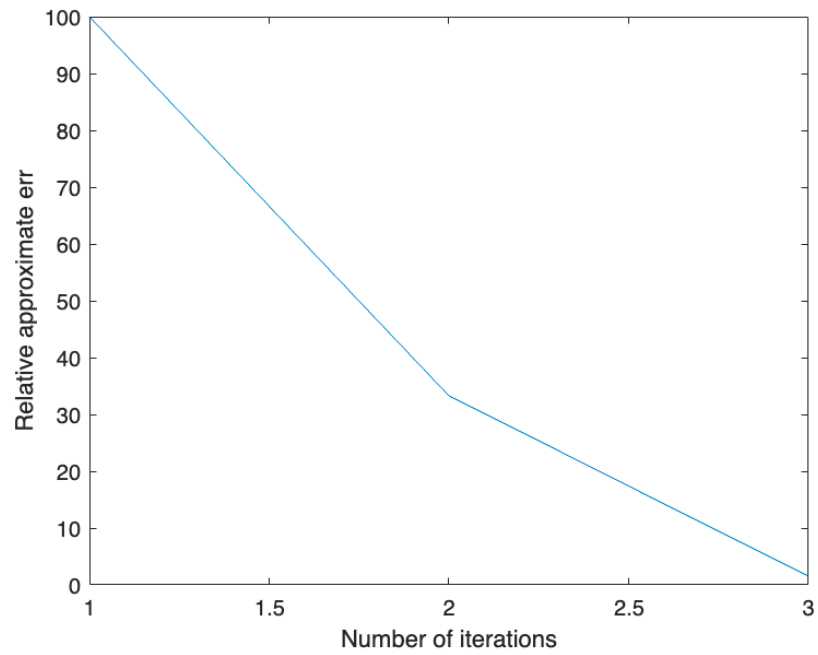


#### 4. NEWTON RAPHSON

MaxErr = **0.0100**

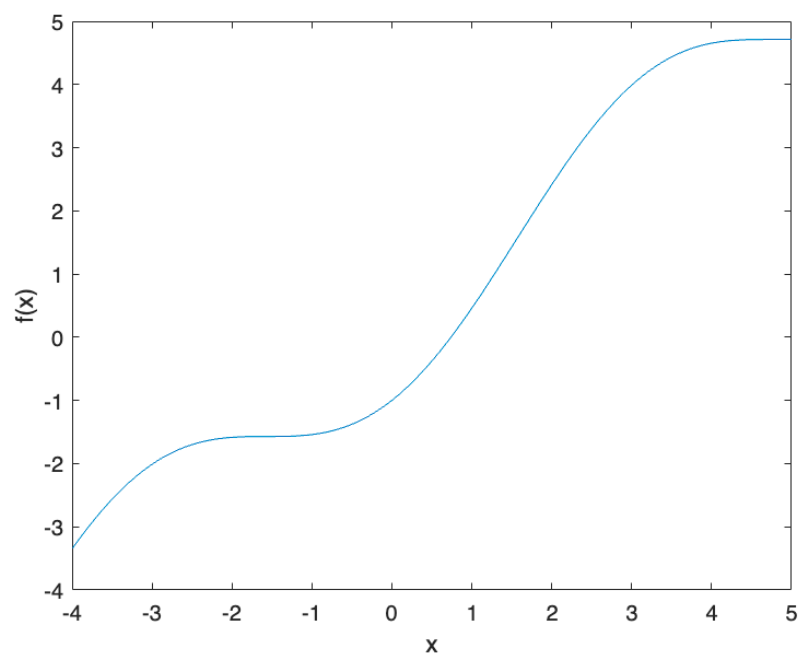


The root of the equation: 0.739085

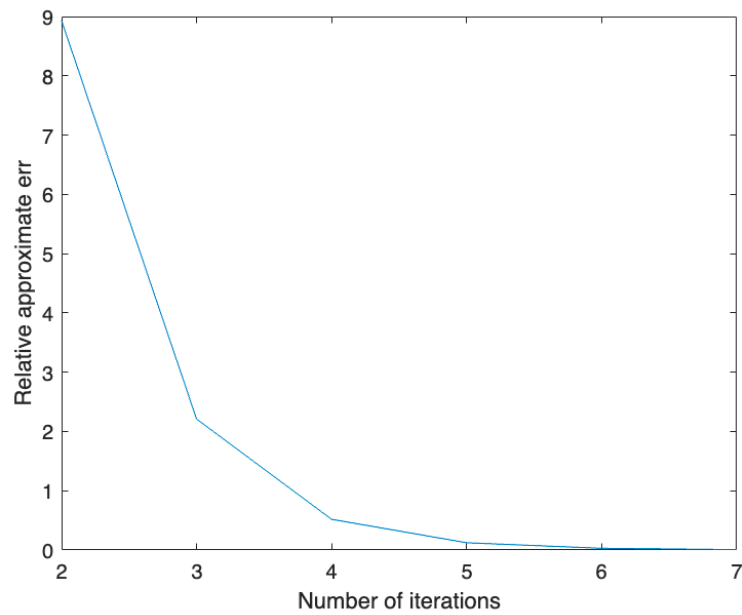


## 5. SECANT

MaxErr = 0.0100



The root of the equation: **0.739075**

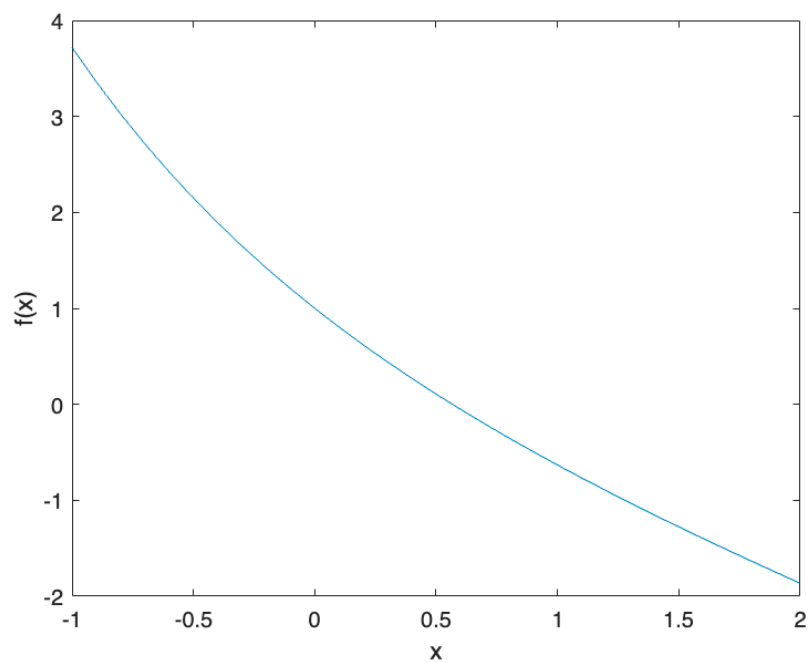


$$f(x) = \exp(-x) - x = 0$$

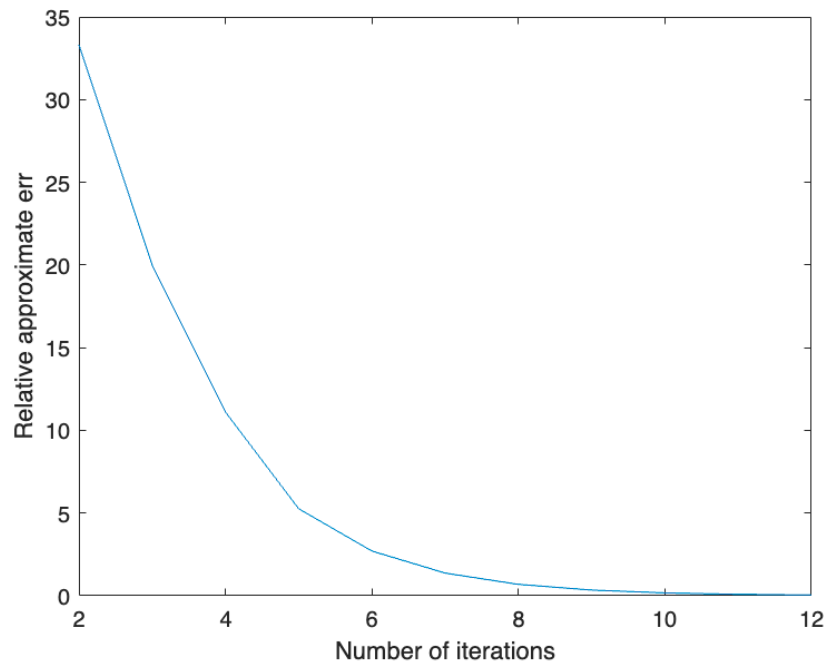
Use the initial bracket as (0,1) or the initial guess as 0; maximum iterations 50; and maximum  $\varepsilon_r = 0.05\%$ . For Fixed-Point method, use  $\phi(x) = \exp(-x)$ .

#### 1. BISECTION

MaxErr = **0.0500**

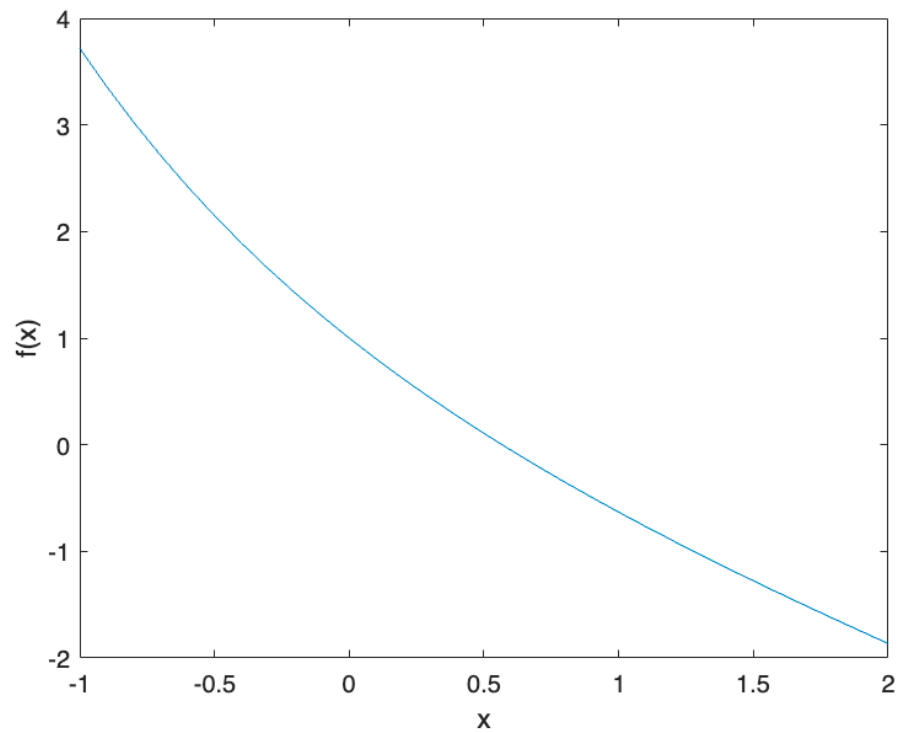


The root of the equation: 0.567139

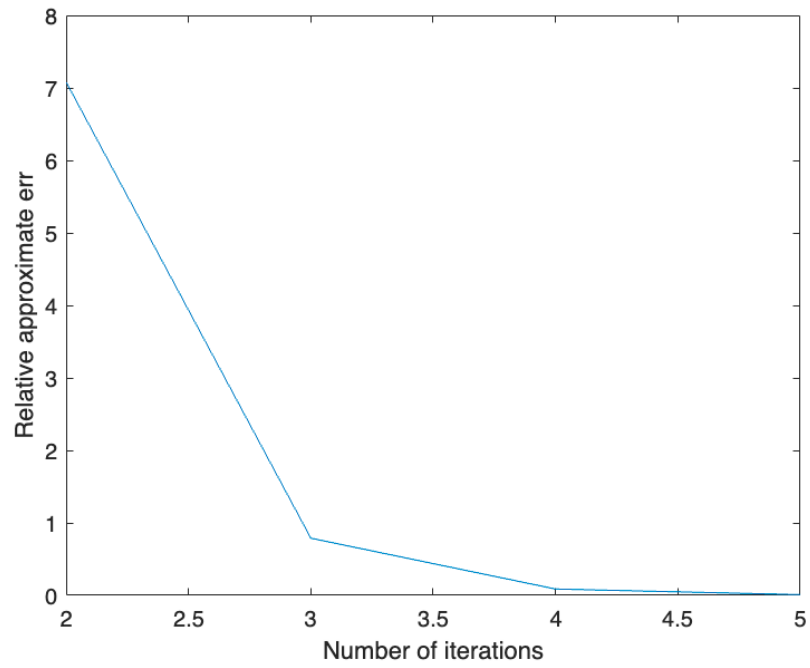


## 2. FALSE-POSITION

MaxErr = 0.0500

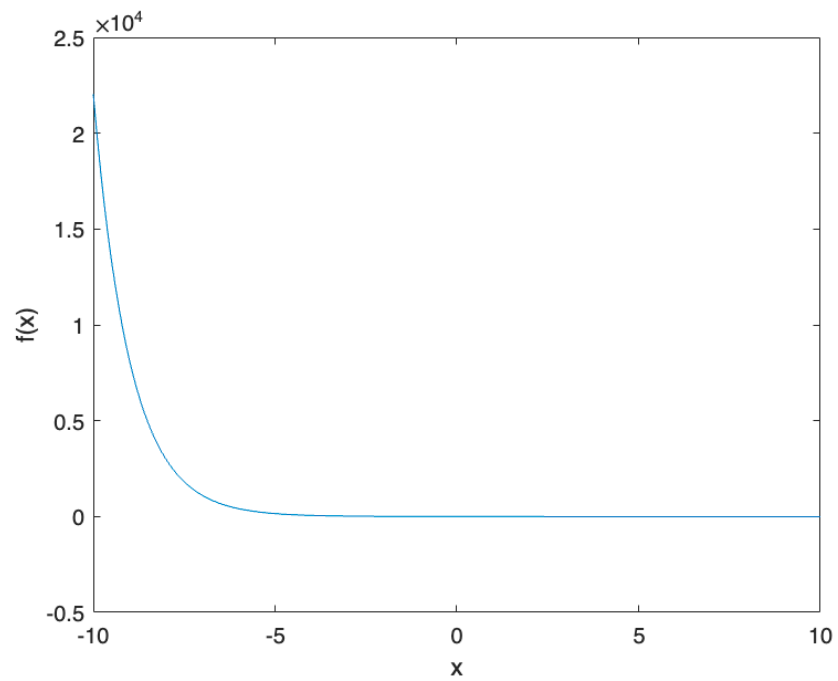


The root of the equation: **0.567150**



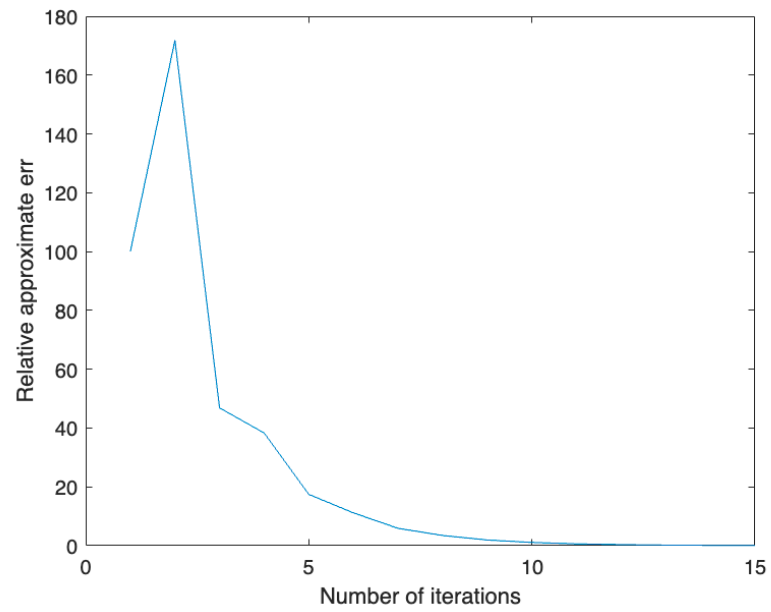
### 3. FIXED-POINT

MaxErr = **0.0500**



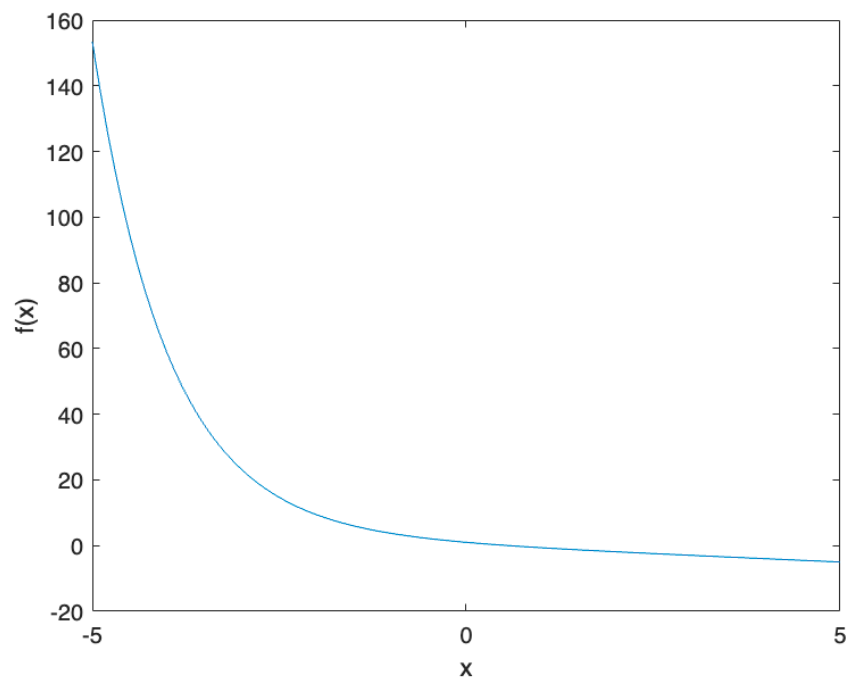


The root of the equation: 0.567068

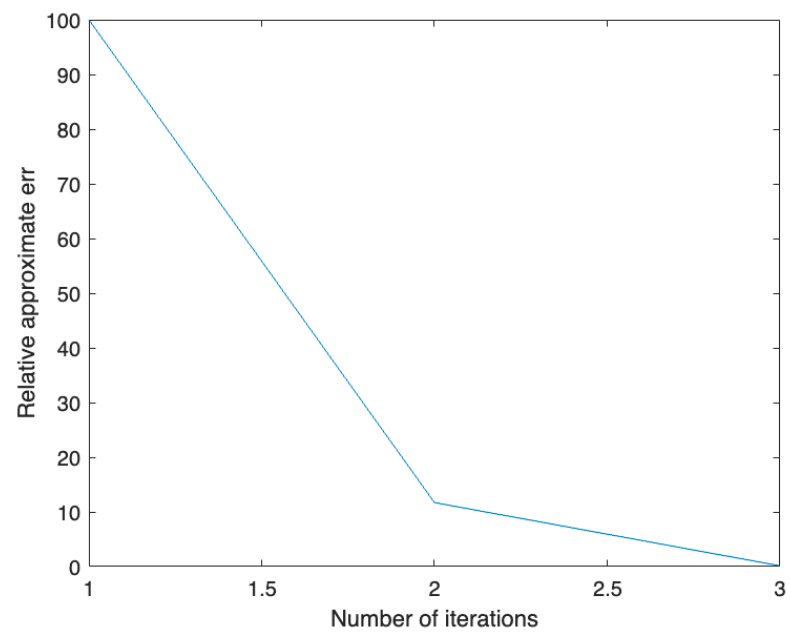


#### 4. NEWTON-RAPHSON

MaxErr = 0.0500

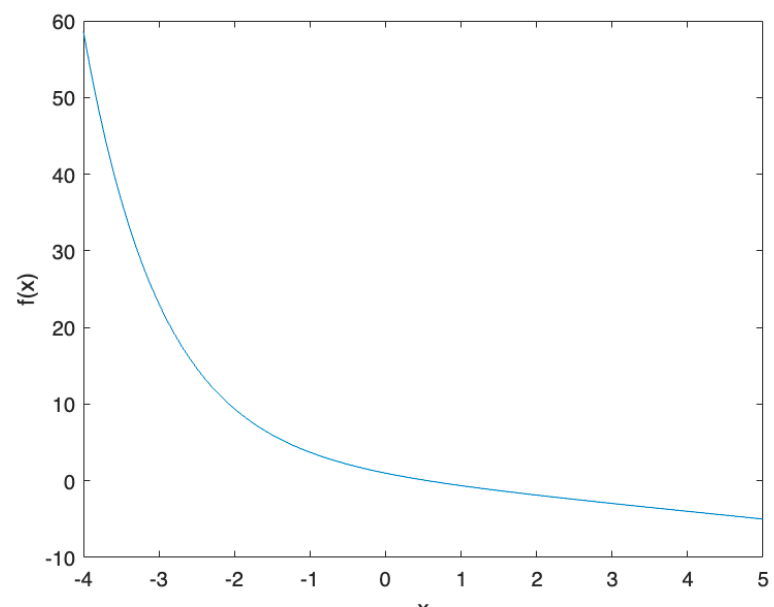


The root of the equation: 0.567143

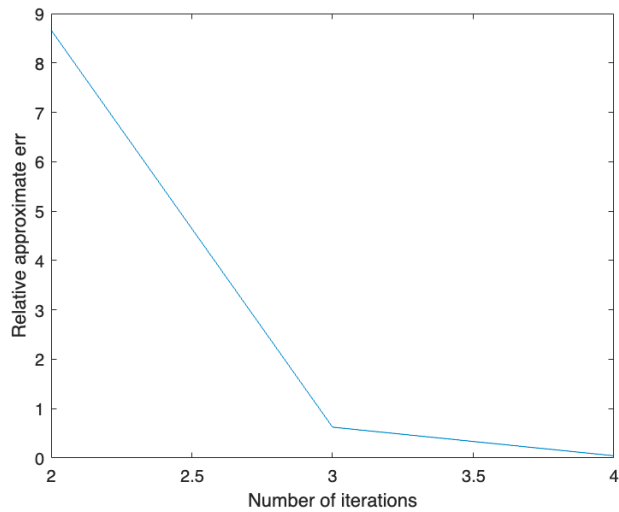


## 5. SECANT

MaxErr = 0.0500



The root of the equation: 0.567126



## QUESTION2

**Test polynomial:**

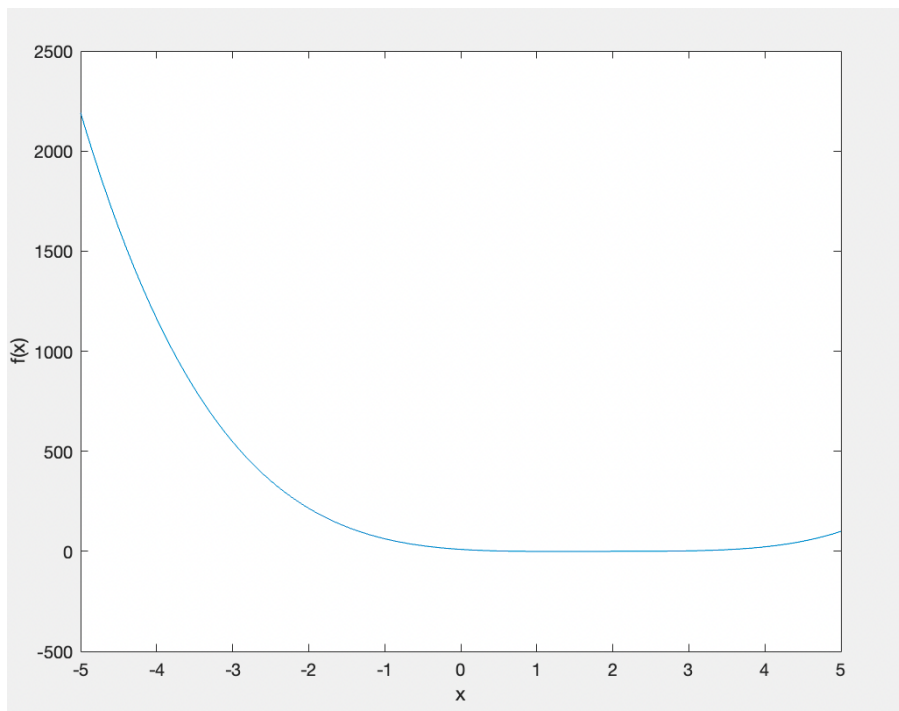
$$f(x) = x^4 - 7.4x^3 + 20.44x^2 - 24.184x + 9.6448 = 0$$

Maximum iteration: 50

Maximum relative approximate error: 0.01%

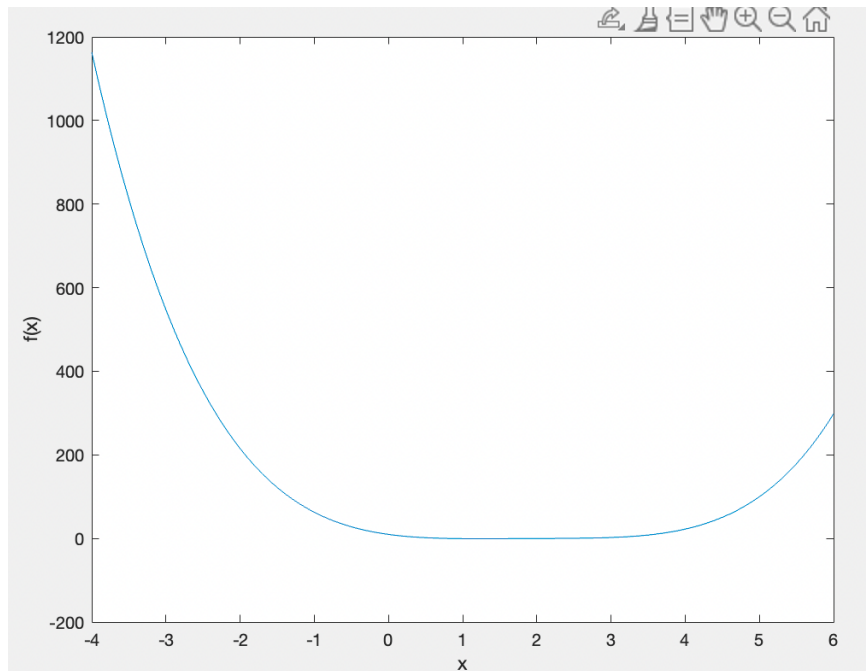
Muller method: (-1,0,1)

**The root is 0.800000**



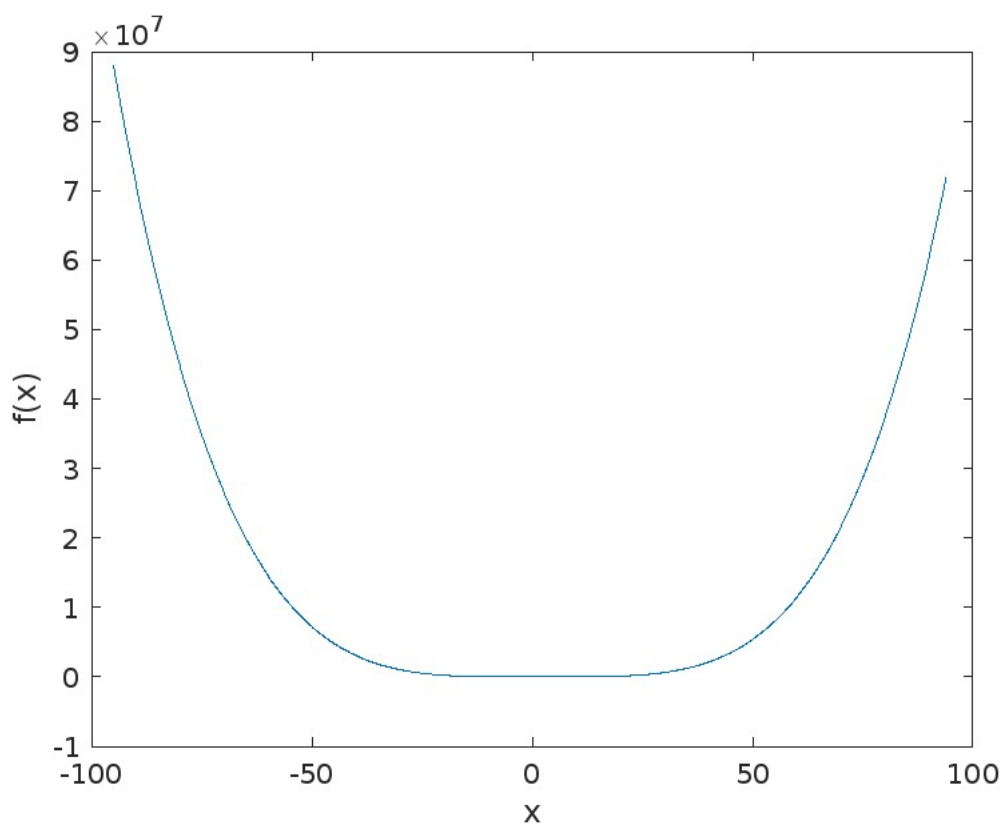
Muller method: (0,1,2)

**The root is 2.200000**



Bairstow method:  $(\alpha_0 = -5, \alpha_1 = 4)$

**The root is 2.200000 and 2.200000**



Bairstow method:  $(\alpha_0 = -2, \alpha_1 = 2)$

**The root is 2.200000 and 0.800000**

