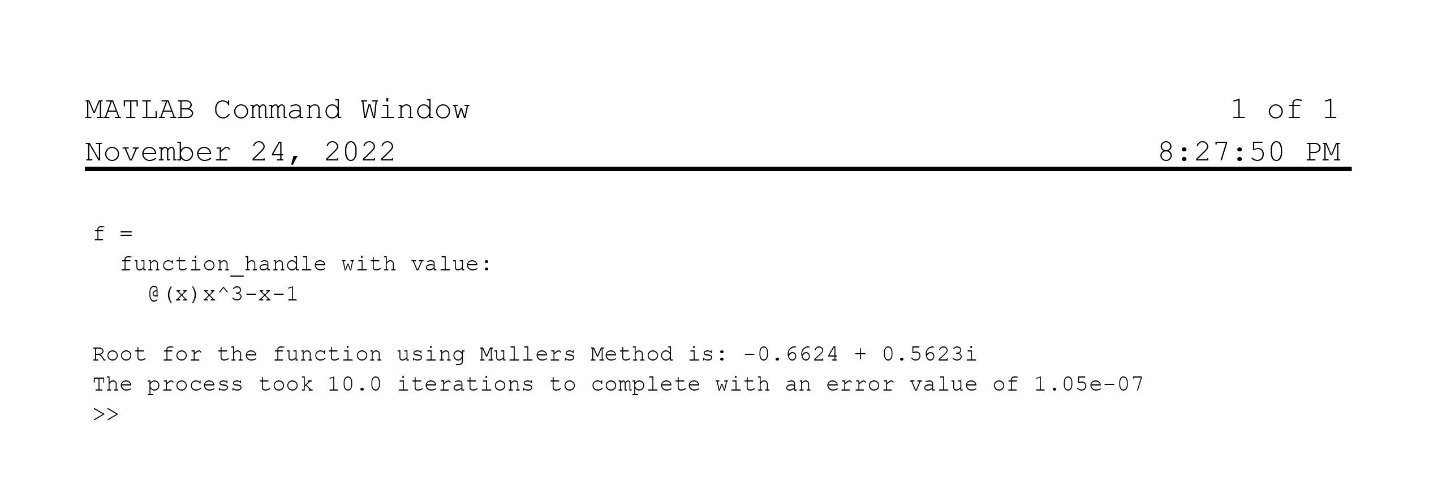
|  |  |
| --- | --- |
| LAB # 3 | |
| **Implementation of Mullers Method to locate the Root of the Given Equation** | |
| Student Name: | Kamil Rasheed Siddiqui |
| Registration Number: | ME201024 |
| Semester: | 5 |
| Group: | A |
| Date of Submission: | November 25, 2022 |
|  |  |

**Problem Statement:**

**Determine the real root of f (x) = x^3 – x – 1 using mullers method. Use x0 = 3, x1 = 2.5, x2 = 4.5 and iterate until the estimated error falls below a level of error = 0.0001**

**Program Code:**

|  |
| --- |
| clear, clc  %% Defining Function and Variables  f = @(x) x^3 - 2\*x - 1    x0 = 3;  x1 = 2.5;  x2 = 4.5;  e = .00001;  eCalc = e\*2;  n = 0;    %% Program  while(eCalc > e)  fx0 = f(x0);  fx1 = f(x1);  fx2 = f(x2);    h0 = x1 - x0;  h1 = x2 - x1;    delta0 = (fx1 - fx0) / (x1 - x0);  delta1 = (fx2 - fx1) / (x2 - x1);    a = (delta1 - delta0) / (h1 + h0);  b = (a \* h1) + delta1;  c = fx2;    denoRoot = (b^2 - 4\*a\*c)^(1/2);  denoPos = b + denoRoot;  denoNeg = b - denoRoot;    if (abs(denoPos) > abs(denoNeg))  deno = denoPos;  else if (abs(denoNeg) > abs(denoPos))  deno = denoNeg;  else  deno = denoPos;  end  end    x3 = x2 - ((2\*c) / (deno));  eCalc = abs((x3 - x2) / (x3));    if (eCalc > e)  p = x0; q = x1; r = x2;  x0 = x1; x1 = x2; x2 = x3;  end    n = n + 1;  end    %% Print Statements    if imag(x3) == 0  fprintf("\nRoot for the function using Mullers Method is: %.4f\n", x3)  else if imag(x3) ~= 0  fprintf("\nRoot for the function using Mullers Method is: %.4f + %.4fi\n", real(x3), imag(x3))  end  end  fprintf("The process took %.1f iterations to complete with an error value of %.2s \n", n, eCalc) |

**Results:**