

1.

- (a) Implement the bisection method in MATLAB to find the smallest positive root of

$$e^{-x} = \tan x \quad (1)$$

- (b) Solve (1) using the secant method. (Use either a calculator or MATLAB.)

2. Write a MATLAB function `Newton(f, df, x, tol)` to implement Newton's method. You need to supply functions $f(x)$ and $df(x)(f'(x))$. The input x is the initial guess and tol is the desired accuracy which should be attained when $|x_{i+1} - x_i| < tol$. You should limit the number of iterations and report a failure to converge. Use the **error** function.

- (a) Try your function to solve equation (1). Print out the iterates and the function values.
 (b) Use your function to find the first ten positive solutions of

$$x = \cot x.$$

Note: The careful selection of x is critical.

- (c) Try the function on the double root $x = 2$ of

$$x^3 - x^2 - 8x + 12 = 0.$$

Use $x = 3$ and $tol = 10^{-6}$. What is the rate of convergence ?

- (d) Newton can be used to find complex roots also. By starting with a non-real initial guess, find the complex roots of

$$x^3 + 2x - 5 = 0.$$

3. Write down two fixed point procedures for finding a zero of the function $f(x) = 2x^2 + 6e^{-x} - 4$.

4. Which of the following iterations will converge to the indicated fixed point α (provided x_0 is sufficiently close to α)? If it does converge, give the order of convergence; for linear convergence, give the rate of linear convergence.

(a)
$$x_{n+1} = -16 + 6x_n + \frac{12}{x_n} \quad \alpha = 2$$

(b)
$$x_{n+1} = \frac{2}{3}x_n + \frac{1}{x_n^2} \quad \alpha = 3^{1/3}$$

(c)
$$x_{n+1} = \frac{12}{1 + x_n} \quad \alpha = 3$$

5. Ex. 4.13. p.156, *Shampine, Allen & Pruess*. Use the MATLAB code FZERO.
6. Ex. 4.16. p.156, *Shampine, Allen & Pruess*. Use the MATLAB code FZERO.
7. Solve the system

$$x^2 + xy^3 = 9 \quad 3x^2y - y^3 = 4$$

using Newton's method for nonlinear systems. Use each of the initial guesses $(x_0, y_0) = (1.2, 2.5), (-2, 2.5), (-1.2, -2.5), (2, -2.5)$. Observe which root to which the method converges, the number of iterates required, and the speed of convergence. Write a MATLAB function with the initial guess as input. Be sure to take advantage of the fact that MATLAB works with vectors.

8. Ex. 5.12. p.191, *Shampine, Allen & Pruess*. Use QUADL. You will need the MATLAB command GLOBAL to pass the value c into the integrand.
9. Consider the system $A\mathbf{x} = \mathbf{b}$ where

$$A = \begin{pmatrix} 4 & -1 & 0 & -1 & 0 \\ -1 & 4 & -1 & 0 & -1 \\ 0 & -1 & 4 & -1 & 0 \\ -1 & 0 & -1 & 4 & -1 \\ 0 & -1 & 0 & -1 & 4 \end{pmatrix}$$

and $\mathbf{b} = (-2, -1, 6, 7, 14)'$. Solve the system using

- (a) The Cholesky factorization of A (MATLAB: CHOL)
- (b) Jacobi iteration.
- (c) Gauss-Seidel iteration. (The MATLAB command TRIL might be useful.)