Study Problems

1. If $\tilde{x}$ and $\tilde{y}$ are approximations of x and y with relative errors of at most 0.001 and 0.003, respectively, what can you say about the relative error of $\tilde{x}\tilde{y}$ as an approximation of xy?

2. Give an example where $\tilde{x}$ and $\tilde{y}$ are approximations of numbers x and y each with relative error of less than $1/10\%$, but for which $\tilde{x}+\tilde{y}$ has a relative error of more than 50% as an approximation of $x+y$.

3. Write an algorithm for which the input consists of the n-vector a_i and the scalar x and the output is

$$y = a_1 x^{n-1} + a_2 x^{n-2} + \cdots + a_{n-1}x + a_n.$$

Use Horner's method.

4. Find the degree 2 Taylor polynomial for $f(x) = e^x \sin x$ about the point $x = 0$. Bound the error in this approximation for x in the interval $[-\pi/4, \pi/4]$.

5. Find the LU decomposition of

$$\begin{pmatrix} 2 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{pmatrix}.$$

6. Find the roots of the quadratic equation $x^2 - 40x + 1 = 0$ to five significant digits using $\sqrt{399} \approx 19.975$ (which is correct to five significant digits).

7. Find a positive number δ such that $\sin x$ agrees with x to within a *relative* error of at most 1% for all $|x| \leq \delta$.

8. Use Newton's method to derive an iteration which converges to $\sqrt[3]{2}$. Start with $x_0 = 1$ and compute two iterations.

9. Suppose that x^* is a fixed point of $g(x)$, that $g'(x^*) = 0$, and that $|g''(x)|$ is continuous and bounded. Starting from some x_0, define a sequence x_n by $x_{n+1} = g(x_n)$. Prove that $|e_{n+1}| \leq M|e_n|^2$, where $e_n = x_n - x^*$. Give an expression for M.

10. Write down the formulas for Newton's method and for the Secant method applied to the equation $x^3 - \cos x = 0$.

11. State precisely, with all necessary hypotheses, a theorem asserting the local quadratic convergence of Newton's method.

12. Which of the following sequences appears to be converging to 1 quadratically?

a) 0.8027893, 0.9810498, 0.9980254, 0.9998018

b) 0.8027893, 0.9601194, 0.9984095, 0.9999975

c) 0.8027893, 0.9013942, 0.9500143, 0.9749988

d) 0.8027893, 0.9993521, 0.9992412, 0.9999999

13. Recall that IEEE single precision uses a 24 bit mantissa and an exponent running from -126 to 127. What is the smallest floating point number in this system which is greater than 10?

14. Find equivalent formulas for the following expressions which are less prone to round-off error for small x.

a) $\dfrac{1-\cos x}{x^2}$

b) $\log(x+1) - \log x$

c) $\sqrt[3]{1+x} - 1$

15. Use Taylor polynomials to find approximations for the following expressions which are less prone to round-off error for small x.

a) $\dfrac{e^x - 1}{x}$

b) $\dfrac{e^x - e^{-x}}{2x}$

c) $\dfrac{\log(1-x) + xe^{x/2}}{x^3}$

16. For each of the following statements, fill in the blank with one of the following: "the bisection method", "Newton's method", or "the secant method".

a) _____ converges quickly if you start close to a root of the function, but it requires that you be able to compute the derivative of the function.

b) _____ requires that you know a bracketing interval.

c) _____ has an order of convergence of about 1.6.

d) The errors in _____ obey $|e_{n+1}| \approx M|e_n||e_{n-1}|$.

17. A tridiagonal matrix is a matrix in which a_{ij} is zero unless $i = j-1$, j, or $j+1$. An $n \times n$ tridiagonal matrix can be expressed in terms of an n-vector d giving its diagonal elements, an $(n-1)$-vector l giving the elements just below the diagonal, and an $(n-1)$-vector u giving the elements just above the diagonal, thus:

$$A = \begin{pmatrix} d_1 & u_1 & 0 & 0 & \dots & 0 \\ l_1 & d_2 & u_2 & 0 & \dots & 0 \\ 0 & l_2 & d_3 & u_3 & \dots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ & & & l_{n-2} & d_{n-1} & u_{n-1} \\ 0 & & \dots & & l_{n-1} & d_n \end{pmatrix}$$

a) Write an algorithm which, given the vectors d, l, and u and an n-vector b, returns the vector x solving $Ax = b$.

b) Implement your algorithm in Matlab and make sure it works.

c) What is the operation count for your algorithm? Count multiplications and divisions as one operation each and ignore additions and subtractions. Use `flops` to test your answer to part b.