on the properties of the fluid, the applied temperature gradient, and the geometry of the problem. Typical values Lorenz used are $\sigma = 10$, $r = 28$, and $b = 8/3$. Write a program using Newton's method to solve this system of equations. You should find three different solutions.

5.22. (*a*) Solve the following nonlinear system by the obvious fixed-point iteration scheme.

$$\begin{aligned} x_1 &= -\frac{\cos(x_1)}{81} + \frac{x_2^2}{9} + \frac{\sin(x_3)}{3}, \\ x_2 &= \frac{\sin(x_1)}{3} + \frac{\cos(x_3)}{3}, \\ x_3 &= -\frac{\cos(x_1)}{9} + \frac{x_2}{3} + \frac{\sin(x_3)}{6}. \end{aligned}$$

(*b*) At the fixed point, what is the value of the constant C in the linear convergence rate? How does this compare with your observation of the actual convergence behavior?

(*c*) Solve the same system using Newton's method and compare the convergence behavior with that for fixed-point iteration.

5.23. Write a program to solve the system of nonlinear equations

$$\begin{aligned} 16x^4 + 16y^4 + z^4 &= 16, \\ x^2 + y^2 + z^2 &= 3, \\ x^3 - y &= 0 \end{aligned}$$

using Newton's method. You may solve the resulting linear system at each iteration either by a library routine or by a linear system solver of your own design. As starting guess, you may take each where σ (the Prandtl number), r (the Rayleigh number), and b are positive constants that depend on the properties of the fluid, the applied temperature gradient, and the geometry of the problem. Typical values Lorenz used are $\sigma = 10$, $r = 28$, and $b = 8/3$. Write a program using Newton's method to solve this system of equations. You should find three different solutions.

5.22. (*a*) Solve the following nonlinear system by the obvious fixed-point iteration scheme.

$$\begin{aligned} x_1 &= -\frac{\cos(x_1)}{81} + \frac{x_2^2}{9} + \frac{\sin(x_3)}{3}, \\ x_2 &= \frac{\sin(x_1)}{3} + \frac{\cos(x_3)}{3}, \\ x_3 &= -\frac{\cos(x_1)}{9} + \frac{x_2}{3} + \frac{\sin(x_3)}{6}. \end{aligned}$$

(*b*) At the fixed point, what is the value of the constant C in the linear convergence rate? How does this compare with your observation of the actual convergence behavior?

(*c*) Solve the same system using Newton's method and compare the convergence behavior with that for fixed-point iteration.

5.23. Write a program to solve the system of nonlinear equations

$$\begin{aligned} 16x^4 + 16y^4 + z^4 &= 16, \\ x^2 + y^2 + z^2 &= 3, \\ x^3 - y &= 0 \end{aligned}$$

using Newton's method. You may solve the resulting linear system at each iteration either by a library routine or by a linear system solver of your own design. As starting guess, you may take each variable to be 1. In addition, try nonlinear solvers from a subroutine library, based on both Newton and secant updating methods, and compare the solutions obtained and the convergence rates with those for your program.

5.24. The derivation of a two-point Gaussian quadrature rule (which we will consider in Section 8.3.3) on the interval $[-1, 1]$ using the method of undetermined coefficients leads to the following system of nonlinear equations for the nodes x_1, x_2 and weights w_1, w_2:

$$\begin{aligned} w_1 + w_2 &= 2, \\ w_1 x_1 + w_2 x_2 &= 0, \\ w_1 x_1^2 + w_2 x_2^2 &= 2/3, \\ w_1 x_1^3 + w_2 x_2^3 &= 0. \end{aligned}$$

Solve this system for x_1, x_2, w_1, and w_2 using a library routine or one of your own design. How many different solutions can you find?

5.25. Use a library routine, or one of your own design, to solve the following system of nonlinear equations:

$$\begin{aligned} \sin(x) + y^2 + \log(z) &= 3, \\ 3x + 2^y - z^3 &= 0, \\ x^2 + y^2 + z^3 &= 6. \end{aligned}$$

Try to find as many different solutions as you can. You should find at least four real solutions (there are also complex solutions).

5.26. A model for combustion of propane in air yields the system of nonlinear equations

$$\begin{aligned} x_1 + x_4 &= 3, \\ 2x_1 + x_2 + x_4 + x_7 + x_8 + x_9 + 2x_{10} &= R + 10, \\ 2x_2 + 2x_5 + x_6 + x_7 &= 8, \\ 2x_3 + x_5 &= 4R, \\ x_1 x_5 - 0.193\, x_2 x_4 &= 0, \\ x_6\sqrt{x_2} - 0.002597\sqrt{x_2 x_4 S} &= 0, \\ x_7\sqrt{x_4} - 0.003448\sqrt{x_1 x_4 S} &= 0, \\ x_4 x_8 - 0.00001799\, x_2 S &= 0, \\ x_4 x_9 - 0.0002155\, x_1\sqrt{x_3 S} &= 0, \\ x_4^2(x_{10} - 0.00003846\, S) &= 0, \end{aligned}$$

where $R = 4.056734$ and $S = \sum_{i=1}^{10} x_i$. Use a library routine to solve this nonlinear system. (*Hint*: If any square root should fail because a variable becomes negative, replace that variable by its absolute value.)

5.27. Each of the following systems of nonlinear equations may present some difficulty in computing a solution. Use a library routine, or one of your own design, to solve each of the systems from the given starting point. In some cases, the nonlinear solver may fail to converge or may converge to a point other than a solution. When this happens, try to explain the reason for the observed behavior. Also note the convergence rate attained, and if it is slower than expected, try to explain why.

Experiment with both loose and strict error tolerances. Make a plot analogous to Fig. 8.4 to show graphically where the integrand is sampled by the adaptive routine.

(*a*)

$$f(x) = \begin{cases} 0 & 0 \le x < 0.3 \\ 1 & 0.3 \le x \le 1 \end{cases}$$

(*b*)

$$f(x) = \begin{cases} 1/(x+2) & 0 \le x < e-2 \\ 0 & e-2 \le x \le 1 \end{cases}$$

(*c*)

$$f(x) = \begin{cases} e^x & -1 \le x < 0 \\ e^{1-x} & 0 \le x \le 2 \end{cases}$$

(*d*)

$$f(x) = \begin{cases} e^{10x} & -1 \le x < 0.5 \\ e^{10(1-x)} & 0.5 \le x \le 1.5 \end{cases}$$

(*e*)

$$f(x) = \begin{cases} \sin(\pi x) & 0 \le x < 0.5 \\ \sin^2(\pi x) & 0.5 \le x \le 1.0 \end{cases}$$

8.6. Evaluate the following quantities using each of the given methods:

(*a*) Use an adaptive quadrature routine to evaluate each of the integrals

$$I_k = e^{-1} \int_0^1 x^k e^x \, dx$$

for $k = 0, 1, \ldots, 20$.

(*b*) Verify that the integrals just defined satisfy the recurrence

$$I_k = 1 - kI_{k-1},$$

and use it to generate the same quantities, starting with $I_0 = 1 - e^{-1}$.

(*c*) Generate the same quantities using the backward recurrence

$$I_{k-1} = (1 - I_k)/k,$$

beginning with $I_n = 0$ for some chosen value $n > 20$. Experiment with different values of n to see the effect on the accuracy of the values generated.

(*d*) Compare the three methods with respect to accuracy, stability, and execution time. Can you explain these results?

8.7. The surface area of an ellipsoid obtained by rotating an ellipse about its major axis is given by the integral

$$I(f) = 4\pi\sqrt{\alpha} \int_0^{1/\sqrt{\beta}} \sqrt{1 - Kx^2} \, dx,$$

where $\beta = 100$, $\alpha = (3 - 2\sqrt{2})/\beta$, and $K = \beta\sqrt{1 - \alpha\beta}$. Use an adaptive quadrature routine to compute this integral. Make a plot analogous to Fig. 8.4 to show graphically where the integrand is sampled by the adaptive routine. Compare your results with the exact integral, which is given by

$$\pi\sqrt{\alpha/K}\,(\pi + \sin(2\theta) - 2\theta),$$

where $\theta = \arccos(\sqrt{K/\beta})$.

8.8. The intensity of diffracted light near a straight edge is determined by the values of the *Fresnel integrals*

$$C(x) = \int_0^x \cos\left(\frac{\pi t^2}{2}\right) dt$$

and

$$S(x) = \int_0^x \sin\left(\frac{\pi t^2}{2}\right) dt.$$

Use an adaptive quadrature routine to evaluate these integrals for enough values of x to draw a smooth plot of $C(x)$ and $S(x)$ over the range $0 \le x \le 5$. You may wish to check your results by obtaining a routine for computing Fresnel integrals from a special function library (see Section 7.5.1).

8.9. The period of a simple pendulum is determined by the *complete elliptic integral* of the first kind

$$K(x) = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1 - x^2 \sin^2\theta}}.$$

Use an adaptive quadrature routine to evaluate this integral for enough values of x to draw a smooth plot of $K(x)$ over the range $0 \le x \le 1$. You may wish to check your results by obtaining a routine for computing elliptic integrals from a special function library (see Section 7.5.1).

$$f(x) = \begin{cases} \sin(\pi x) & 0 \le x < 0.5 \\ \sin^2(\pi x) & 0.5 \le x \le 1.0 \end{cases}$$

8.6. Evaluate the following quantities using each of the given methods:

(*a*) Use an adaptive quadrature routine to evaluate each of the integrals

$$I_k = e^{-1} \int_0^1 x^k e^x \, dx$$

for $k = 0, 1, \ldots, 20$.

(*b*) Verify that the integrals just defined satisfy the recurrence

$$I_k = 1 - kI_{k-1},$$

and use it to generate the same quantities, starting with $I_0 = 1 - e^{-1}$.

(*c*) Generate the same quantities using the backward recurrence

$$I_{k-1} = (1 - I_k)/k,$$

beginning with $I_n = 0$ for some chosen value $n > 20$. Experiment with different values of n to see the effect on the accuracy of the values generated.

(*d*) Compare the three methods with respect to accuracy, stability, and execution time. Can you explain these results?

8.10. The *gamma function* is defined by

$$\Gamma(x) = \int_0^\infty t^{x-1} e^{-t}\, dt, \quad x > 0.$$

Write a program to compute the value of this function from the definition using each of the following approaches:

(*a*) Truncate the infinite interval of integration and use a composite quadrature rule, such as trapezoid or Simpson. You will need to do some experimentation or analysis to determine where to truncate the interval, based on the usual tradeoff between efficiency and accuracy.

(*b*) Truncate the interval and use a standard adaptive quadrature routine. Again, explore the tradeoff between accuracy and efficiency.

(*c*) Gauss-Laguerre quadrature is designed for the interval $[0, \infty]$ and the weight function e^{-t}, so it is ideal for approximating this integral. Look up the nodes and weights for Gauss-Laguerre quadrature rules of various orders (see [2, 385, 435], for example) and compute the resulting estimates for the integral.

(*d*) If available, use an adaptive quadrature routine designed for an unbounded interval of integration.

For each method, compute the approximate value of the integral for several values of x in the range 1 to 10. Compare your results with the values given by the built-in **gamma** function or with the known values for integer arguments,

$$\Gamma(n) = (n-1)!\,.$$

How do the various methods compare in efficiency for a given level of accuracy?

8.11. Planck's theory of blackbody radiation leads to the integral

by truncating
ing a composi
both the limi
the composite
exact value of

(*b*) Repeat pa
quadrature ro

(*c*) Gauss-He
the interval
$\exp(-x^2)$, so
tegral. Look u
Hermite quad
[2, 385, 435],
sulting estima

8.13. In two
uniform charg
$x \le 1,\ -1 \le$
units, the elec
outside the reg

$$\Phi(\hat{x}, \hat{y}) =$$

Evaluate this i
plot the $\Phi(\hat{x}, \hat{y}$
$2 \le \hat{y} \le 10$.

8.14. Using a
double integral

over each of th
(*a*) The unit sq
(*b*) The quarte
quadrant, i.e., a

8.15. (*a*) Writ
using the comp

After debugging, test your routines using some of the integrals in the previous problems and compare the results with those previously obtained. How does the efficiency of your adaptive routine compare with that of your nonadaptive routine?

8.16. Select an adaptive quadrature routine and try to devise an integrand function for which it gives an answer that is completely wrong. (*Hint:* This problem may require at least one round of trial and error.) Can you devise a *smooth* function for which the adaptive routine is seriously in error?

8.17. (*a*) Solve the integral equation

$$\int_0^1 (s^2+t^2)^{1/2} u(t)\, dt = \frac{(s^2+1)^{3/2} - s^3}{3}$$

on the interval $[0,1]$ by discretizing the integral using the composite Simpson quadrature rule with n equally spaced points t_j, and also using the same n points for the s_i. Solve the resulting linear system $\boldsymbol{Ax} = \boldsymbol{y}$ using a library routine for Gaussian elimination with partial pivoting. Experiment with various values for n in the range from 3 to 15, comparing your results with the known unique solution, $u(t) = t$. Which value of n gives the best results? Can you explain why?

(*b*) For each value of n in part *a*, compute the condition number of the matrix $\boldsymbol{A}$. How does it behave as a function of n?

(*c*) Repeat part *a*, this time solving the linear system using the singular value decomposition, but omit any "small" singular values. Try various thresholds for truncating the singular values, and again compare your results with the known true solution.

(*d*) Repeat part *a*, this time using the method of regularization. Experiment with various values for the regularization parameter μ to determine which value yields the best results for a given value of n. For each value of μ, plot a point on a two-dimensional graph whose axes are the norm of the solution and the norm of the residual. What is the shape of the curve traced out as μ varies? Does this shape suggest an optimal value for μ?

(*e*) Repeat part *a*, this time using an optimization routine to minimize $\|\boldsymbol{y} - \boldsymbol{Ax}\|_2^2$ subject to the constraint that the components of the solution must be nonnegative. Again, compare your results with the known true solution.

(*f*) Repeat part *e*, this time imposing the additional constraint that the solution be monotonically increasing, i.e., $x_1 \geq 0$ and $x_i - x_{i-1} \geq 0$, $i = 2, \ldots, n$. How much difference does this make in approximating the true solution?

8.18. In this exercise we will experiment with numerical differentiation using data from Computer Problem 3.1:

t	0.0	1.0	2.0	3.0	4.0	5.0
y	1.0	2.7	5.8	6.6	7.5	9.9

For each of the following methods for estimating the derivative, compute the derivative of the original data and also experiment with randomly perturbing the y values to determine the sensitivity of the resulting derivative estimates. For each method, comment on both the reasonableness of the derivative estimates and their sensitivity to perturbations. Note that the data are monotonically increasing, so one might expect the derivative always to be positive.

(*a*) For $n = 0, 1, \ldots, 5$, fit a polynomial of degree n by least squares to the data, then differentiate the resulting polynomial and evaluate the derivative at each of the given t values.

(*b*) Interpolate the data with a cubic spline, differentiate the resulting piecewise cubic polynomial, and evaluate the derivative at each of the given t values (some spline routines provide the derivative automatically, but it can be done manually if necessary).

(*c*) Repeat part *b*, this time using a smoothing spline routine. Experiment with various levels of smoothing, using whatever mechanism for controlling the degree of smoothing that the routine provides.

(*d*) Interpolate the data with a monotonic Hermite cubic, differentiate the resulting piecewise cubic polynomial, and evaluate the derivative at each of the given t values.