2.3 Newton's Method and Its Extensions

Newton's (or the *Newton-Raphson*) **method** is one of the most powerful and well-known numerical methods for solving a root-finding problem. There are many ways of introducing Newton's method.

If we only want an algorithm, we can consider the technique graphically, as is often done in calculus. Another possibility is to derive Newton's method as a technique to obtain faster convergence than offered by other types of functional iteration, as is done in Section 2.4. A third means of introducing Newton's method, which is discussed next, is based on Taylor polynomials. We will see there that this particular derivation produces not only the method, but also a bound for the error of the approximation.

Suppose that $f \in C^2[a, b]$. Let $p_0 \in [a, b]$ be an approximation to p such that $f'(p_0) \neq 0$ and $|p - p_0|$ is "small." Consider the first Taylor polynomial for $f(x)$ expanded about p_0 and evaluated at $x = p$.

$$f(p) = f(p_0) + (p - p_0)f'(p_0) + \frac{(p - p_0)^2}{2}f''(\xi(p)),$$

where $\xi(p)$ lies between p and p_0. Since $f(p) = 0$, this equation gives

$$0 = f(p_0) + (p - p_0)f'(p_0) + \frac{(p - p_0)^2}{2}f''(\xi(p)).$$

Newton's method is derived by assuming that since $|p - p_0|$ is small, the term involving $(p - p_0)^2$ is much smaller, so

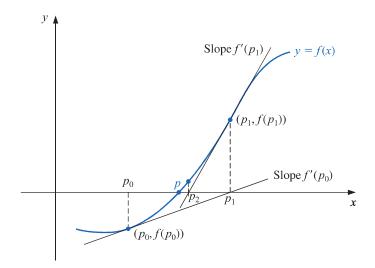$$0 \approx f(p_0) + (p - p_0)f'(p_0).$$

Solving for p gives

$$p \approx p_0 - \frac{f(p_0)}{f'(p_0)} \equiv p_1.$$

This sets the stage for Newton's method, which starts with an initial approximation p_0 and generates the sequence $\{p_n\}_{n=0}^{\infty}$, by

$$p_n = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})}, \quad \text{for } n \geq 1. \tag{2.7}$$

Figure 2.8 on page 68 illustrates how the approximations are obtained using successive tangents. (Also see Exercise 15.) Starting with the initial approximation p_0, the approximation p_1 is the x-intercept of the tangent line to the graph of f at $(p_0, f(p_0))$. The approximation p_2 is the x-intercept of the tangent line to the graph of f at $(p_1, f(p_1))$ and so on. Algorithm 2.3 follows this procedure.

To find a solution to $f(x) = 0$ given an initial approximation p_0:

INPUT   initial approximation p_0; tolerance TOL; maximum number of iterations N_0.

OUTPUT   approximate solution p or message of failure.

Step 1   Set $i = 1$.

Step 2   While $i \leq N_0$ do Steps 3–6.

    Step 3   Set $p = p_0 - f(p_0)/f'(p_0)$.   (*Compute p_i.*)

    Step 4   If $|p - p_0| < TOL$ then
        OUTPUT (p);   (*The procedure was successful.*)
        STOP.

    Step 5   Set $i = i + 1$.

    Step 6   Set $p_0 = p$.   (*Update p_0.*)

Step 7   OUTPUT ('The method failed after N_0 iterations, $N_0 =$', N_0);
    (*The procedure was unsuccessful.*)
    STOP.

    The stopping-technique inequalities given with the Bisection method are applicable to Newton's method. That is, select a tolerance $\varepsilon > 0$, and construct $p_1, \ldots p_N$ until

$$|p_N - p_{N-1}| < \varepsilon, \tag{2.8}$$

$$\frac{|p_N - p_{N-1}|}{|p_N|} < \varepsilon, \quad p_N \neq 0, \tag{2.9}$$

or

$$|f(p_N)| < \varepsilon. \tag{2.10}$$

A form of Inequality (2.8) is used in Step 4 of Algorithm 2.3. Note that none of the inequalities (2.8), (2.9), or (2.10) give precise information about the actual error $|p_N - p|$. (See Exercises 16 and 17 in Section 2.1.)

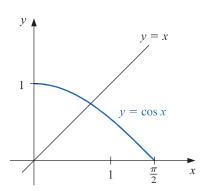Newton's method is a functional iteration technique with $p_n = g(p_{n-1})$, for which

$$g(p_{n-1}) = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})}, \quad \text{for } n \geq 1. \tag{2.11}$$

In fact, this is the functional iteration technique that was used to give the rapid convergence we saw in column (e) of Table 2.2 in Section 2.2.

It is clear from Equation (2.7) that Newton's method cannot be continued if $f'(p_{n-1}) = 0$ for some n. In fact, we will see that the method is most effective when f' is bounded away from zero near p.

Consider the function $f(x) = \cos x - x = 0$. Approximate a root of f using **(a)** a fixed-point method, and **(b)** Newton's Method

(a) A solution to this root-finding problem is also a solution to the fixed-point problem $x = \cos x$, and the graph in Figure 2.9 implies that a single fixed-point p lies in $[0, \pi/2]$.

n	p_n
0	0.7853981635
1	0.7071067810
2	0.7602445972
3	0.7246674808
4	0.7487198858
5	0.7325608446
6	0.7434642113
7	0.7361282565

Newton's Method

n	p_n
0	0.7853981635
1	0.7395361337
2	0.7390851781
3	0.7390851332
4	0.7390851332

Table 2.3 shows the results of fixed-point iteration with $p_0 = \pi/4$. The best we could conclude from these results is that $p \approx 0.74$.

(b) To apply Newton's method to this problem we need $f'(x) = -\sin x - 1$. Starting again with $p_0 = \pi/4$, we generate the sequence defined, for $n \geq 1$, by

$$p_n = p_{n-1} - \frac{f(p_{n-1})}{f(p'_{n-1})} = p_{n-1} - \frac{\cos p_{n-1} - p_{n-1}}{-\sin p_{n-1} - 1}.$$

This gives the approximations in Table 2.4. An excellent approximation is obtained with $n = 3$. Because of the agreement of p_3 and p_4 we could reasonably expect this result to be accurate to the places listed.

Example 1 shows that Newton's method can provide extremely accurate approximations with very few iterations. For that example, only one iteration of Newton's method was needed to give better accuracy than 7 iterations of the fixed-point method. It is now time to examine Newton's method more carefully to discover why it is so effective.

The Taylor series derivation of Newton's method at the beginning of the section points out the importance of an accurate initial approximation. The crucial assumption is that the term involving $(p - p_0)^2$ is, by comparison with $|p - p_0|$, so small that it can be deleted. This will clearly be false unless p_0 is a good approximation to p. If p_0 is not sufficiently close to the actual root, there is little reason to suspect that Newton's method will converge to the root. However, in some instances, even poor initial approximations will produce convergence. (Exercises 20 and 21 illustrate some of these possibilities.)

The following convergence theorem for Newton's method illustrates the theoretical importance of the choice of p_0.

Let $f \in C^2[a, b]$. If $p \in (a, b)$ is such that $f(p) = 0$ and $f'(p) \neq 0$, then there exists a $\delta > 0$ such that Newton's method generates a sequence $\{p_n\}_{n=1}^{\infty}$ converging to p for any initial approximation $p_0 \in [p - \delta, p + \delta]$.

The proof is based on analyzing Newton's method as the functional iteration scheme $p_n = g(p_{n-1})$, for $n \geq 1$, with

$$g(x) = x - \frac{f(x)}{f'(x)}.$$

Let k be in $(0, 1)$. We first find an interval $[p - \delta, p + \delta]$ that g maps into itself and for which $|g'(x)| \leq k$, for all $x \in (p - \delta, p + \delta)$.

Since f' is continuous and $f'(p) \neq 0$, part (a) of Exercise 29 in Section 1.1 implies that there exists a $\delta_1 > 0$, such that $f'(x) \neq 0$ for $x \in [p - \delta_1, p + \delta_1] \subseteq [a, b]$. Thus g is defined and continuous on $[p - \delta_1, p + \delta_1]$. Also

$$g'(x) = 1 - \frac{f'(x)f'(x) - f(x)f''(x)}{[f'(x)]^2} = \frac{f(x)f''(x)}{[f'(x)]^2},$$

for $x \in [p - \delta_1, p + \delta_1]$, and, since $f \in C^2[a, b]$, we have $g \in C^1[p - \delta_1, p + \delta_1]$.

By assumption, $f(p) = 0$, so

$$g'(p) = \frac{f(p)f''(p)}{[f'(p)]^2} = 0.$$

Since g' is continuous and $0 < k < 1$, part (b) of Exercise 29 in Section 1.1 implies that there exists a δ, with $0 < \delta < \delta_1$, and

$$|g'(x)| \leq k, \quad \text{for all } x \in [p - \delta, p + \delta].$$

It remains to show that g maps $[p - \delta, p + \delta]$ into $[p - \delta, p + \delta]$. If $x \in [p - \delta, p + \delta]$, the Mean Value Theorem implies that for some number ξ between x and p, $|g(x) - g(p)| = |g'(\xi)||x - p|$. So

$$|g(x) - p| = |g(x) - g(p)| = |g'(\xi)||x - p| \leq k|x - p| < |x - p|.$$

Since $x \in [p - \delta, p + \delta]$, it follows that $|x - p| < \delta$ and that $|g(x) - p| < \delta$. Hence, g maps $[p - \delta, p + \delta]$ into $[p - \delta, p + \delta]$.

All the hypotheses of the Fixed-Point Theorem 2.4 are now satisfied, so the sequence $\{p_n\}_{n=1}^{\infty}$, defined by

$$p_n = g(p_{n-1}) = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})}, \quad \text{for } n \geq 1,$$

converges to p for any $p_0 \in [p - \delta, p + \delta]$.

Theorem 2.6 states that, under reasonable assumptions, Newton's method converges provided a sufficiently accurate initial approximation is chosen. It also implies that the constant k that bounds the derivative of g, and, consequently, indicates the speed of convergence of the method, decreases to 0 as the procedure continues. This result is important for the theory of Newton's method, but it is seldom applied in practice because it does not tell us how to determine δ.

In a practical application, an initial approximation is selected and successive approximations are generated by Newton's method. These will generally either converge quickly to the root, or it will be clear that convergence is unlikely.

Newton's method is an extremely powerful technique, but it has a major weakness: the need to know the value of the derivative of f at each approximation. Frequently, $f'(x)$ is far more difficult and needs more arithmetic operations to calculate than $f(x)$.

To circumvent the problem of the derivative evaluation in Newton's method, we introduce a slight variation. By definition,

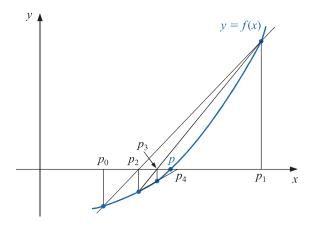$$f'(p_{n-1}) = \lim_{x \to p_{n-1}} \frac{f(x) - f(p_{n-1})}{x - p_{n-1}}.$$

If p_{n-2} is close to p_{n-1}, then

$$f'(p_{n-1}) \approx \frac{f(p_{n-2}) - f(p_{n-1})}{p_{n-2} - p_{n-1}} = \frac{f(p_{n-1}) - f(p_{n-2})}{p_{n-1} - p_{n-2}}.$$

Using this approximation for $f'(p_{n-1})$ in Newton's formula gives

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}. \tag{2.12}$$

This technique is called the **Secant method** and is presented in Algorithm 2.4. (See Figure 2.10.) Starting with the two initial approximations p_0 and p_1, the approximation p_2 is the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$. The approximation p_3 is the x-intercept of the line joining $(p_1, f(p_1))$ and $(p_2, f(p_2))$, and so on. Note that only one function evaluation is needed per step for the Secant method after p_2 has been determined. In contrast, each step of Newton's method requires an evaluation of both the function and its derivative.

ALGORITHM

2.4

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1:

INPUT   initial approximations p_0, p_1; tolerance *TOL*; maximum number of iterations N_0.

OUTPUT   approximate solution p or message of failure.

Step 1   Set $i = 2$;
    $q_0 = f(p_0)$;
    $q_1 = f(p_1)$.

Step 2   While $i \le N_0$ do Steps 3–6.

    Step 3   Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$.   (*Compute* p_i.)

    Step 4   If $|p - p_1| < TOL$ then
            OUTPUT (p);   (*The procedure was successful.*)
            STOP.

    Step 5   Set $i = i + 1$.

    Step 6   Set $p_0 = p_1$;   (*Update* p_0, q_0, p_1, q_1.)
        $q_0 = q_1$;
        $p_1 = p$;
        $q_1 = f(p)$.

Step 7   OUTPUT ('The method failed after N_0 iterations, $N_0 =$', N_0);
        (*The procedure was unsuccessful.*)
        STOP.

The next example involves a problem considered in Example 1, where we used Newton's method with $p_0 = \pi/4$.

Use the Secant method to find a solution to $x = \cos x$, and compare the approximations with those given in Example 1 which applied Newton's method.

In Example 1 we compared fixed-point iteration and Newton's method starting with the initial approximation $p_0 = \pi/4$. For the Secant method we need two initial approximations. Suppose we use $p_0 = 0.5$ and $p_1 = \pi/4$. Succeeding approximations are generated by the formula

$$p_n = p_{n-1} - \frac{(p_{n-1} - p_{n-2})(\cos p_{n-1} - p_{n-1})}{(\cos p_{n-1} - p_{n-1}) - (\cos p_{n-2} - p_{n-2})}, \quad \text{for } n \ge 2.$$

These give the results in Table 2.5.

Secant	
n	p_n
0	0.5
1	0.7853981635
2	0.7363841388
3	0.7390581392
4	0.7390851493
5	0.7390851332

Newton	
n	p_n
0	0.7853981635
1	0.7395361337
2	0.7390851781
3	0.7390851332
4	0.7390851332

Comparing the results in Table 2.5 from the Secant method and Newton's method, we see that the Secant method approximation p_5 is accurate to the tenth decimal place, whereas Newton's method obtained this accuracy by p_3. For this example, the convergence of the Secant method is much faster than functional iteration but slightly slower than Newton's method. This is generally the case. (See Exercise 14 of Section 2.4.)

Newton's method or the Secant method is often used to refine an answer obtained by another technique, such as the Bisection method, since these methods require good first approximations but generally give rapid convergence.

Each successive pair of approximations in the Bisection method brackets a root p of the equation; that is, for each positive integer n, a root lies between a_n and b_n. This implies that, for each n, the Bisection method iterations satisfy

$$|p_n - p| < \frac{1}{2}|a_n - b_n|,$$

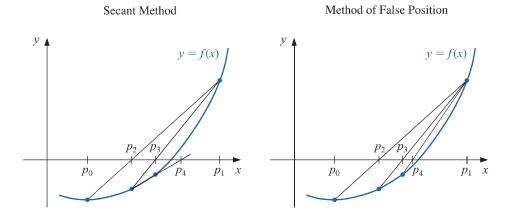which provides an easily calculated error bound for the approximations.

Root bracketing is not guaranteed for either Newton's method or the Secant method. In Example 1, Newton's method was applied to $f(x) = \cos x - x$, and an approximate root was found to be 0.7390851332. Table 2.5 shows that this root is not bracketed by either p_0 and p_1 or p_1 and p_2. The Secant method approximations for this problem are also given in Table 2.5. In this case the initial approximations p_0 and p_1 bracket the root, but the pair of approximations p_3 and p_4 fail to do so.

The **method of False Position** (also called *Regula Falsi*) generates approximations in the same manner as the Secant method, but it includes a test to ensure that the root is always bracketed between successive iterations. Although it is not a method we generally recommend, it illustrates how bracketing can be incorporated.

First choose initial approximations p_0 and p_1 with $f(p_0) \cdot f(p_1) < 0$. The approximation p_2 is chosen in the same manner as in the Secant method, as the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$. To decide which secant line to use to compute p_3, consider $f(p_2) \cdot f(p_1)$, or more correctly sgn $f(p_2) \cdot$ sgn $f(p_1)$.

- If sgn $f(p_2) \cdot$ sgn $f(p_1) < 0$, then p_1 and p_2 bracket a root. Choose p_3 as the x-intercept of the line joining $(p_1, f(p_1))$ and $(p_2, f(p_2))$.

- If not, choose p_3 as the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_2, f(p_2))$, and then interchange the indices on p_0 and p_1.

In a similar manner, once p_3 is found, the sign of $f(p_3) \cdot f(p_2)$ determines whether we use p_2 and p_3 or p_3 and p_1 to compute p_4. In the latter case a relabeling of p_2 and p_1 is performed. The relabeling ensures that the root is bracketed between successive iterations. The process is described in Algorithm 2.5, and Figure 2.11 shows how the iterations can differ from those of the Secant method. In this illustration, the first three approximations are the same, but the fourth approximations differ.

Secant Method

Method of False Position

**ALGORITHM
2.5**

To find a solution to $f(x) = 0$ given the continuous function f on the interval $[p_0, p_1]$ where $f(p_0)$ and $f(p_1)$ have opposite signs:

INPUT initial approximations p_0, p_1; tolerance TOL; maximum number of iterations N_0.

OUTPUT approximate solution p or message of failure.

Step 1 Set $i = 2$;
$\quad\quad q_0 = f(p_0)$;
$\quad\quad q_1 = f(p_1)$.

Step 2 While $i \leq N_0$ do Steps 3–7.

$\quad$ *Step 3* Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i.)

$\quad$ *Step 4* If $|p - p_1| < TOL$ then
$\quad\quad\quad$ OUTPUT (p); (*The procedure was successful.*)
$\quad\quad\quad$ STOP.

$\quad$ *Step 5* Set $i = i + 1$;
$\quad\quad\quad q = f(p)$.

$\quad$ *Step 6* If $q \cdot q_1 < 0$ then set $p_0 = p_1$;
$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad q_0 = q_1$.

$\quad$ *Step 7* Set $p_1 = p$;
$\quad\quad\quad q_1 = q$.

Step 8 OUTPUT ('Method failed after N_0 iterations, $N_0 =$', N_0);
$\quad\quad$ (*The procedure unsuccessful.*)
$\quad\quad$ STOP.

Use the method of False Position to find a solution to $x = \cos x$, and compare the approximations with those given in Example 1 which applied fixed-point iteration and Newton's method, and to those found in Example 2 which applied the Secant method.

To make a reasonable comparison we will use the same initial approximations as in the Secant method, that is, $p_0 = 0.5$ and $p_1 = \pi/4$. Table 2.6 shows the results of the method of False Position applied to $f(x) = \cos x - x$ together with those we obtained using the Secant and Newton's methods. Notice that the False Position and Secant approximations agree through p_3 and that the method of False Position requires an additional iteration to obtain the same accuracy as the Secant method.

	False Position	Secant	Newton
n	p_n	p_n	p_n
0	0.5	0.5	0.7853981635
1	0.7853981635	0.7853981635	0.7395361337
2	0.7363841388	0.7363841388	0.7390851781
3	0.7390581392	0.7390581392	0.7390851332
4	0.7390848638	0.7390851493	0.7390851332
5	0.7390851305	0.7390851332	
6	0.7390851332		

The added insurance of the method of False Position commonly requires more calculation than the Secant method, just as the simplification that the Secant method provides over Newton's method usually comes at the expense of additional iterations. Further examples of the positive and negative features of these methods can be seen by working Exercises 17 and 18.

Maple has Newton's method, the Secant method, and the method of False Position implemented in its *NumericalAnalysis* package. The options that were available for the Bisection method are also available for these techniques. For example, to generate the results in Tables 2.4, 2.5, and 2.6 we could use the commands

$with(Student[NumericalAnalysis])$

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

$Newton\left(f, x = \dfrac{\pi}{4.0}, tolerance = 10^{-8}, output = sequence, maxiterations = 20\right)$

$Secant\left(f, x = \left[0.5, \dfrac{\pi}{4.0}\right], tolerance = 10^{-8}, output = sequence, maxiterations = 20\right)$

and

$FalsePosition\left(f, x = \left[0.5, \dfrac{\pi}{4.0}\right], tolerance = 10^{-8}, output = sequence, maxiterations = 20\right)$

1. Let $f(x) = x^2 - 6$ and $p_0 = 1$. Use Newton's method to find p_2.

2. Let $f(x) = -x^3 - \cos x$ and $p_0 = -1$. Use Newton's method to find p_2. Could $p_0 = 0$ be used?

3. Let $f(x) = x^2 - 6$. With $p_0 = 3$ and $p_1 = 2$, find p_3.
   a. Use the Secant method.
   b. Use the method of False Position.
   c. Which of **a.** or **b.** is closer to $\sqrt{6}$?

4. Let $f(x) = -x^3 - \cos x$. With $p_0 = -1$ and $p_1 = 0$, find p_3.
   a. Use the Secant method.   b. Use the method of False Position.

5. Use Newton's method to find solutions accurate to within 10^{-4} for the following problems.
   a. $x^3 - 2x^2 - 5 = 0$, $[1, 4]$ b. $x^3 + 3x^2 - 1 = 0$, $[-3, -2]$
   c. $x - \cos x = 0$, $[0, \pi/2]$ d. $x - 0.8 - 0.2 \sin x = 0$, $[0, \pi/2]$

6. Use Newton's method to find solutions accurate to within 10^{-5} for the following problems.
   a. $e^x + 2^{-x} + 2\cos x - 6 = 0$ for $1 \le x \le 2$
   b. $\ln(x - 1) + \cos(x - 1) = 0$ for $1.3 \le x \le 2$
   c. $2x \cos 2x - (x - 2)^2 = 0$ for $2 \le x \le 3$ and $3 \le x \le 4$
   d. $(x - 2)^2 - \ln x = 0$ for $1 \le x \le 2$ and $e \le x \le 4$
   e. $e^x - 3x^2 = 0$ for $0 \le x \le 1$ and $3 \le x \le 5$
   f. $\sin x - e^{-x} = 0$ for $0 \le x \le 1$ $3 \le x \le 4$ and $6 \le x \le 7$

7. Repeat Exercise 5 using the Secant method.

8. Repeat Exercise 6 using the Secant method.

9. Repeat Exercise 5 using the method of False Position.

10. Repeat Exercise 6 using the method of False Position.

11. Use all three methods in this Section to find solutions to within 10^{-5} for the following problems.
    a. $3xe^x = 0$ for $1 \le x \le 2$
    b. $2x + 3\cos x - e^x = 0$ for $0 \le x \le 1$

12. Use all three methods in this Section to find solutions to within 10^{-7} for the following problems.

   a. $x^2 - 4x + 4 - \ln x = 0$ for $1 \leq x \leq 2$ and for $2 \leq x \leq 4$

   b. $x + 1 - 2 \sin \pi x = 0$ for $0 \leq x \leq 1/2$ and for $1/2 \leq x \leq 1$

13. Use Newton's method to approximate, to within 10^{-4}, the value of x that produces the point on the graph of $y = x^2$ that is closest to $(1, 0)$. [*Hint:* Minimize $[d(x)]^2$, where $d(x)$ represents the distance from (x, x^2) to $(1, 0)$.]

14. Use Newton's method to approximate, to within 10^{-4}, the value of x that produces the point on the graph of $y = 1/x$ that is closest to $(2, 1)$.

15. The following describes Newton's method graphically: Suppose that $f'(x)$ exists on $[a, b]$ and that $f'(x) \neq 0$ on $[a, b]$. Further, suppose there exists one $p \in [a, b]$ such that $f(p) = 0$, and let $p_0 \in [a, b]$ be arbitrary. Let p_1 be the point at which the tangent line to f at $(p_0, f(p_0))$ crosses the x-axis. For each $n \geq 1$, let p_n be the x-intercept of the line tangent to f at $(p_{n-1}, f(p_{n-1}))$. Derive the formula describing this method.

16. Use Newton's method to solve the equation

$$0 = \frac{1}{2} + \frac{1}{4}x^2 - x \sin x - \frac{1}{2} \cos 2x, \quad \text{with } p_0 = \frac{\pi}{2}.$$

Iterate using Newton's method until an accuracy of 10^{-5} is obtained. Explain why the result seems unusual for Newton's method. Also, solve the equation with $p_0 = 5\pi$ and $p_0 = 10\pi$.

17. The fourth-degree polynomial

$$f(x) = 230x^4 + 18x^3 + 9x^2 - 221x - 9$$

has two real zeros, one in $[-1, 0]$ and the other in $[0, 1]$. Attempt to approximate these zeros to within 10^{-6} using the

   a. Method of False Position

   b. Secant method

   c. Newton's method

Use the endpoints of each interval as the initial approximations in (a) and (b) and the midpoints as the initial approximation in (c).

18. The function $f(x) = \tan \pi x - 6$ has a zero at $(1/\pi) \arctan 6 \approx 0.447431543$. Let $p_0 = 0$ and $p_1 = 0.48$, and use ten iterations of each of the following methods to approximate this root. Which method is most successful and why?

   a. Bisection method

   b. Method of False Position

   c. Secant method

19. The iteration equation for the Secant method can be written in the simpler form

$$p_n = \frac{f(p_{n-1})p_{n-2} - f(p_{n-2})p_{n-1}}{f(p_{n-1}) - f(p_{n-2})}.$$

Explain why, in general, this iteration equation is likely to be less accurate than the one given in Algorithm 2.4.

20. The equation $x^2 - 10 \cos x = 0$ has two solutions, ± 1.3793646. Use Newton's method to approximate the solutions to within 10^{-5} with the following values of p_0.

   a. $p_0 = -100$ **b.** $p_0 = -50$ **c.** $p_0 = -25$

   d. $p_0 = 25$ **e.** $p_0 = 50$ **f.** $p_0 = 100$

21. The equation $4x^2 - e^x - e^{-x} = 0$ has two positive solutions x_1 and x_2. Use Newton's method to approximate the solution to within 10^{-5} with the following values of p_0.