1 Homework 1

Problem 1 (2.7.6) *Convert* $1/12$ *to binary.*

Solution. The doubling approach results in

$$
\begin{array}{ccccccccc}
1/12 & 1/6 & 1/3 & 2/3 & 4/3 & 2/3 & 4/3 & 2/3 & \ldots \\
. & 0 & 0 & 0 & 1 & 0 & 1 & 0 & \ldots
\end{array}
$$

from which we may infer that

$$
\frac{1}{12} = 0.0001010\ldots = 0.000\overline{10}.
$$

To verify the above formula note that

$$
0.000\overline{10} = 2^{-3} \cdot 0.\overline{10} = 2^{-3} \sum_{k=1}^{\infty} 2 \cdot 2^{-2k}
$$

$$
= 2^{-2} \sum_{k=1}^{\infty} \frac{1}{4^k} = 2^{-2} \frac{1/4}{1 - 1/4} = 2^{-2} \frac{1}{3} = \frac{1}{12}.
$$

∎

Problem 2 (2.7.7) *Convert* $1/16$ *to binary.*

Solution. Again use the doubling approach to obtain

$$
\begin{array}{cccccccc}
1/16 & 1/8 & 1/4 & 1/2 & 1 & 0 & 0 & \ldots \\
. & 0 & 0 & 0 & 1 & 0 & 0 & \ldots
\end{array}
$$

which implies

$$
\frac{1}{16} = 0.0001.
$$

∎

Problem 3 (2.7.11) *Convert* $0.111\,111$ *to decimal.*

Solution. Converting each group of three digits to base eight first yields

$$
0.111\,111 = \frac{7}{8} + \frac{7}{8^2} = \frac{56 + 7}{64} = 0.984375.
$$

∎

Problem 4 (2.7.12) *Convert* $0.100\,001$ *to decimal.*

Converting each group of three digits to base eight first yields

$$0.100\,001 = \frac{4}{8} + \frac{1}{8^2} = \frac{32 + 1}{64} = 0.515625.$$

Problem 5 (2.6.2) *Examine the propagation of roundoff through division.*

Solution. Suppose computations are performed using floating-point arithmetic accurate to n significant digits. Let $x_1, x_2 \in \mathbf{R}$ with $x_2 \neq 0$ and denote by $\mathbf{x}_1$ and $\mathbf{x}_2$ respectively the nearest floating-point approximations. Since by definition the relative errors in the approximations are bounded by 5×10^{-n}, setting

$$\varepsilon_1 = \frac{\mathbf{x}_1 - x_1}{x_1} \qquad \text{and} \qquad \varepsilon_2 = \frac{\mathbf{x}_2 - x_2}{x_2}$$

yields $\varepsilon_1, \varepsilon_2 \in E$ where $E = [-5 \times 10^{-n}, 5 \times 10^{-n}]$. A simple rearrangement of these expressions yields that

$$\mathbf{x}_1 = (1 + \varepsilon_1)x_1 \qquad \text{and} \qquad \mathbf{x}_2 = (1 + \varepsilon_2)x_2.$$

Since $x_2 \neq 0$ it follows that $\mathbf{x}_2 \neq 0$. Now, let $\mathbf{x}$ be the nearest floating-point approximation to the exact quotient $x = \mathbf{x}_1/\mathbf{x}_2$. Thus,

$$\mathbf{x} = (1 + \varepsilon)x \qquad \text{for some} \qquad \varepsilon \in E.$$

We now combine the above to show how all the rounding errors have propagated into our estimate $\mathbf{x}$ of the exact quotient x_1/x_2. To do this compute the relative error as

$$
\begin{aligned}
\frac{\mathbf{x} - x_1/x_2}{x_1/x_2} &= \frac{(1 + \varepsilon)x - x_1/x_2}{x_1/x_2} = \frac{(1 + \varepsilon)\mathbf{x}_1/\mathbf{x}_2 - x_1/x_2}{x_1/x_2} \\
&= \frac{(1 + \varepsilon)[(1 + \varepsilon_1)x_1]/[(1 + \varepsilon_2)x_2] - x_1/x_2}{x_1/x_2} \\
&= \frac{(1 + \varepsilon)(1 + \varepsilon_1)}{(1 + \varepsilon_2)} - 1.
\end{aligned}
$$

In other words,

$$\mathbf{x} = \frac{(1 + \varepsilon)(1 + \varepsilon_1)}{(1 + \varepsilon_2)} \cdot \frac{x_1}{x_2}.$$

2

If we assume that all terms involving products of ε, ε_1 and ε_2 can be neglected, the above expression can be simplified as

$$\frac{(1+\varepsilon)(1+\varepsilon_1)}{(1+\varepsilon_2)} = (1+\varepsilon+\varepsilon_1+\varepsilon\varepsilon_1)\sum_{k=0}^{\infty}\binom{-1}{k}(\varepsilon_2)^k$$
$$\approx (1+\varepsilon+\varepsilon_1)(1-\varepsilon_2) \approx (1+\varepsilon+\varepsilon_1-\varepsilon_2).$$

Therefore, the propagated relative error is approximately $\varepsilon + \varepsilon_1 - \varepsilon_2$ and is guaranteed to satisfy the bound

$$|\varepsilon + \varepsilon_1 - \varepsilon_2| \leq 3 \cdot 5 \times 10^{-n} = 1.5 \times 10^{-n+1}.$$

■

Problem 6 (2.8.1) *Derive the corresponding results for division relating to the frequency distribution of the mantissa.*

Solution. We example the way division transforms various distributions. Let x come from the probability density $f(x)$, let y come from $g(y)$ and let the quotient z have the density $h(z)$. Further denote the cumulative distributions by

$$F(x) = \int_{1/b}^{x} f(x)dx, \quad G(y) = \int_{1/b}^{y} g(y)dy \quad \text{and} \quad H(z) = \int_{1/b}^{z} h(z)dz.$$

In order to understand how $H(z)$ depends on the distributions $f(x)$ and $g(y)$ we first note the mantissa of the quotient corresponding to x/y is given by

$$m(x,y) = \begin{cases} x/y & x/y < 1 \\ x/(yb) & x/y \geq 1. \end{cases}$$

where second case occurs when the exponent has to be shifted by one so z is still of the form `0.DDDDDDDDD` where the D's represent the base-b digits the first of which is non-zero. We interpret $H(z) = \mathbf{P}\{m(x,y) \leq z\}$ as the probability measure of the set of points for which the mantissa of the x/y is less than or equal z.

3

Graphically the condition $m(x, y) \le z$ may be depicted by

Values of x and y such that m(x,y) ≤ z for b=10
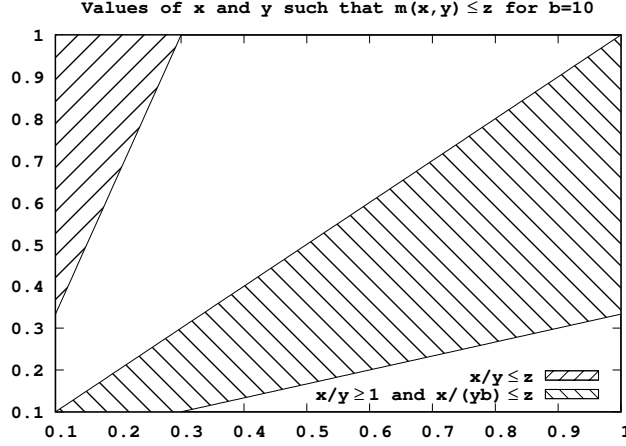
We now integrate over this set to obtain the cumulative distribution function

$$H(z) = \int_{1/b}^{z} \int_{x/z}^{1} f(x)g(y)dy\,dx + \int_{1/b}^{z} \int_{1/b}^{x} f(x)g(y)dy\,dx$$

$$+ \int_{z}^{1} \int_{x/(zb)}^{x} f(x)g(y)dy\,dx$$

$$= \int_{1/b}^{z} f(x)\Big[G(1) - G\Big(\frac{x}{z}\Big)\Big]dx + \int_{1/b}^{z} f(x)\Big[G(x) - G\Big(\frac{1}{b}\Big)\Big]dx$$

$$+ \int_{z}^{1} f(x)\Big[G(x) - G\Big(\frac{x}{zb}\Big)\Big]dx.$$

Then differentiating gives the density

$$h(z) = f(z)\Big[G(1) - G\Big(\frac{z}{z}\Big) + G(z) - G\Big(\frac{1}{b}\Big) - G(z) + G\Big(\frac{z}{zb}\Big)\Big]$$

$$+ \int_{1/b}^{z} f(x)g\Big(\frac{x}{z}\Big)\frac{x}{z^2}dx + \frac{1}{b}\int_{z}^{1} f(x)g\Big(\frac{x}{zb}\Big)\frac{x}{z^2}dx.$$

Suppose the divisor has the reciprocal distribution; that is,

$$g(y) = \frac{1}{y \log b}.$$

4

Substituting this into the formula for $h(z)$ yields

$$h(z) = \int_{1/b}^{z} f(x) \left(\frac{z}{x \log b} \right) \frac{x}{z^2} dx + \frac{1}{b} \int_{z}^{1} f(x) \left(\frac{zb}{x \log b} \right) \frac{x}{z^2} dx$$

$$= \int_{1/b}^{z} f(x) \left(\frac{1}{z \log b} \right) dx + \int_{z}^{1} f(x) \left(\frac{1}{z \log b} \right) dx = \frac{1}{z \log b}.$$

Thus, if one of the divisor of a quotient comes from the reciprocal distribution, then regardless of the distribution of the dividend, the quotient has the reciprocal distribution.

Note for products treated in the text there is a symmetry which implies one does not need consider separately the case when $f(x)$ is given by the reciprocal distribution. However, such is not the case with quotients, so we alternatively suppose

$$f(x) = \frac{1}{x \log b}.$$

Now, substituting this into the formula for $h(z)$ yields

$$h(z) = \int_{1/b}^{z} \frac{1}{x \log b} g\left(\frac{x}{z} \right) \frac{x}{z^2} dx + \frac{1}{b} \int_{z}^{1} \frac{1}{x \log b} g\left(\frac{x}{zb} \right) \frac{x}{z^2} dx$$

$$= \frac{1}{z \log b} \left\{ \int_{1/b}^{z} g\left(\frac{x}{z} \right) \frac{dx}{z} + \int_{z}^{1} g\left(\frac{x}{zb} \right) \frac{dx}{zb} \right\}$$

$$= \frac{1}{z \log b} \left\{ \int_{1/(zb)}^{1} g(u) du + \int_{1/b}^{1/(zb)} g(v) dv \right\} = \frac{1}{z \log b}.$$

Here we have made the substitutions $u = x/z$ and $v = x/(zb)$.

Having treated both cases, we see that provided either $f(x)$ or $g(y)$ have the reciprocal distribution then $h(z)$ will also have the reciprocal distribution. Thus, the persistence of the reciprocal distribution is also established in the case of division.

Next we show how the reciprocal distribution can arise. Let $r(z)$ be the reciprocal distribution and define

$$D\{h\} = \max \left\{ \left| \frac{h(z) - r(z)}{r(z)} \right| : \frac{1}{b} \leq z \leq 1 \right\}.$$

Since

$$r(z) = \int_{1/b}^{z} f(x) r\left(\frac{x}{z} \right) \frac{x}{z^2} dx + \frac{1}{b} \int_{z}^{1} f(x) r\left(\frac{x}{zb} \right) \frac{x}{z^2} dx,$$

5

it follows that

$$\frac{h(z) - r(z)}{r(z)} = \int_{1/b}^{z} f(x)\left\{\frac{g(x/z) - r(x/z)}{r(z)}\right\}\frac{x}{z^2}dx$$

$$+ \frac{1}{b}\int_{z}^{1} f(x)\left\{\frac{g[x/(zb)] - r[x/(zb)]}{r(z)}\right\}\frac{x}{z^2}dx.$$

But

$$\frac{z^2}{x}r(z) = \frac{z}{x\log b} = r\left(\frac{x}{z}\right) \qquad \text{and} \qquad \frac{bz^2}{x}r(z) = \frac{bz}{x\log b} = r\left(\frac{x}{bz}\right)$$

imply

$$\frac{h(z) - r(z)}{r(z)} = \int_{1/b}^{z} f(x)\left\{\frac{g(x/z) - r(x/z)}{r(x/z)}\right\}dx$$

$$+ \int_{z}^{1} f(x)\left\{\frac{g[x/(zb)] - r[x/(zb)]}{r[x/(zb)]}\right\}dx.$$

Since $f(x) \geq 0$ in the two intervals

$$\left|\frac{h(z) - r(z)}{r(z)}\right| \leq \int_{1/b}^{z} f(x)D\{g\}dx + \int_{z}^{1} f(x)D\{g\}dx \leq D\{g\}.$$

for all z. Consequently $D\{h\} \leq D\{g\}$. By a similar argument $D\{h\} \leq D\{f\}$. In particular, it is impossible for the distribution of mantissas of the quotient to get farther from the reciprocal distribution than either the distribution of the dividend or the divisor. ∎

Problem 7 (3.3.1) *For large x rearrange*

$$\frac{1}{1+x} - \frac{1}{x}$$

for evaluation with respect to x.

Solution. Simply by finding a common denominator as

$$\frac{1}{1+x} - \frac{1}{x} = \frac{x - (1+x)}{x(1+x)} = \frac{-1}{x(1+x)}.$$

Note after simplifying that the difference of two nearly equal numbers has been eliminated from the expression so there is no longer any loss of precision due to cancellation. ∎

6

Problem 8 (3.3.4) *For large x rearrange*

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

for evaluation with respect to x.

Solution. Since $a^3 - b^3 = (a - b)(a^2 + ab + b^2)$ then simplify by completing the difference of cubes as

$$\sqrt[3]{x+1} - \sqrt[3]{x} = \left((x+1)^{1/3} - x^{1/3}\right) \cdot \frac{(x+1)^{2/3} + (x+1)^{1/3}x^{1/3} + x^{2/3}}{(x+1)^{2/3} + (x+1)^{1/3}x^{1/3} + x^{2/3}}$$

$$= \frac{1}{(x+1)^{2/3} + (x+1)^{1/3}x^{1/3} + x^{2/3}}.$$

After simplifying the difference of two nearly equal numbers has been eliminated so there is no longer any loss of precision due to cancellation. ∎

Problem 9 (3.3.5) *For small ε rearrange*

$$\cos(x + \varepsilon) - \cos x$$

for evaluation with respect to x.

Solution. Using the angle addition and subtraction formulas

$$\cos(a + b) = \cos a \cos b - \sin a \sin b$$
$$\cos(a - b) = \cos a \cos b + \sin a \sin b$$

write

$$\cos(x + \varepsilon) = \cos\left((x + \varepsilon/2) + \varepsilon/2\right)$$
$$= \cos(x + \varepsilon/2)\cos(\varepsilon/2) - \sin(x + \varepsilon/2)\sin(\varepsilon/2)$$
$$\cos(x) = \cos\left((x + \varepsilon/2) - \varepsilon/2\right)$$
$$= \cos(x + \varepsilon/2)\cos(\varepsilon/2) + \sin(x + \varepsilon/2)\sin(\varepsilon/2).$$

Consequently

$$\cos(x + \varepsilon) - \cos x = -2\sin(x + \varepsilon/2)\sin(\varepsilon/2).$$

Note the subtraction of two nearly equal numbers when ε is small has been eliminated; however, if $\sin x \approx 0$ then loss of precision may still occur in the evaluation of $\sin(x + \varepsilon/2)$. In this case $x \approx k\pi$ for some $k \in \mathbf{Z}$ and there is no way that I know to avoid this other way of losing precision. ∎

Problem 10 (3.3.6) *For large N rearrange*

$$\int_N^{N+1} \frac{dx}{x} = \ln(N+1) - \ln N$$

for evaluation with respect to N.

Solution. Recall the definition of the inverse hyperbolic tangent

$$\text{artanh}\, x = \frac{1}{2}\ln\left(\frac{1+x}{1-x}\right)$$

which is available as the standard subroutine `atanh` for evaluation on the computer. Setting $x = 1/(2N+1)$ yields

$$\frac{1+x}{1-x} = \frac{1+1/(2N+1)}{1-1/(2N+1)} = \frac{2N+2}{2N} = \frac{N+1}{N}.$$

Therefore

$$\int_N^{N+1} \frac{dx}{x} = \ln(N+1) - \ln N = \ln\left(\frac{N+1}{N}\right) = 2\,\text{artanh}\left(\frac{1}{2N+1}\right).$$

Note that a direct evaluation of the expression

$$\ln\left(1+\frac{1}{N}\right)$$

on the computer using the built-in `log` function for N large also leads to loss of precision. This is why we used the inverse hyperbolic tangent. The computer may also have a function called `log1p` defined as $\texttt{log1p}(x) = \log(1+x)$ which could be used instead. In this case one could simply evaluate

$$\int_N^{N+1} \frac{dx}{x} = \ln(N+1) - \ln N \approx \texttt{log1p}\left(\frac{1}{N}\right)$$

without any loss of precision. ∎

Problem 11 (3.4.1) *For ε small compute*

$$\frac{1-\cos\varepsilon}{\varepsilon^2}.$$

Solution. It's tempting to write

$$\frac{1 - \cos \varepsilon}{\varepsilon^2} = \frac{1 - \cos \varepsilon}{\varepsilon^2} \cdot \frac{1 + \cos \varepsilon}{1 + \cos \varepsilon} = \left(\frac{\sin \varepsilon}{\varepsilon}\right)^2 \left(\frac{1}{1 + \cos \varepsilon}\right) = \frac{\mathrm{sinc}^2 \varepsilon}{1 + \cos \varepsilon}.$$

This removes the loss of precision, but unfortunately `sinc` is not part of the standard mathematical library available for evaluation on the computer. One could use an approximation such as

$$\mathrm{sinc}\, \varepsilon = \begin{cases} (\sin \varepsilon)/\varepsilon & \text{for } |\varepsilon| > 10^{-15} \\ 0 & \text{otherwise,} \end{cases}$$

but since this section is about using series we choose a different approach. Since

$$\cos \varepsilon = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k)!} \varepsilon^{2k}$$

then

$$1 - \cos \varepsilon = -\sum_{k=1}^{\infty} \frac{(-1)^k}{(2k)!} \varepsilon^{2k} = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k + 2)!} \varepsilon^{2k+2}$$

and consequently

$$\frac{1 - \cos \varepsilon}{\varepsilon^2} = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k + 2)!} \varepsilon^{2k} \approx \frac{1}{2} - \frac{1}{24} x^2 + \frac{1}{720} x^4 - \frac{1}{40320} x^6 + \cdots.$$

Supposing ε is small one may now truncate at only a few terms N such that

$$\frac{2}{(2N + 2)!} \varepsilon^{2N} < 10^{-15}$$

to obtain an approximation good to within the accuracy of double-precision floating-point arithmetic. ∎

Problem 12 (3.4.4) *For ε small compute*

$$\sqrt{\frac{e^{2\varepsilon} - 1}{e^{\varepsilon} - 1}}.$$

Solution. Again it is tempting to make the exact algebraic simplifications

$$\frac{e^{2\varepsilon} - 1}{e^{\varepsilon} - 1} = \frac{(e^{\varepsilon} - 1)(e^{\varepsilon} + 1)}{e^{\varepsilon} - 1} = e^{\varepsilon} + 1$$

and the compute

$$\sqrt{\frac{e^{2\varepsilon} - 1}{e^{\varepsilon} - 1}} = \sqrt{e^{\varepsilon} + 1}.$$

This results in a exact expression which avoids any loss of precision due to subtraction of nearly equal numbers. However, this expression still involves the numerical computation of `exp` and `sqrt` which are unnecessarily time consuming when ε is known to be small. Rather than expanding the original expression in terms of series we proceed as follows.

$$e^{\varepsilon} + 1 = \left(\sum_{k=0}^{\infty} \frac{1}{k!} \varepsilon^k \right) + 1 = 2 + \sum_{k=1}^{\infty} \frac{1}{k!} \varepsilon^k = 2(1 + \delta)$$

where

$$\delta = \frac{1}{2} \sum_{k=1}^{\infty} \frac{1}{k!} \varepsilon^k.$$

By the binomial theorem

$$\sqrt{e^{\varepsilon} + 1} = \sqrt{2} \cdot \sqrt{1 + \delta} = \sqrt{2} \sum_{l=0}^{\infty} \binom{1/2}{l} \delta^l$$

$$= \sqrt{2} \sum_{l=0}^{\infty} \binom{1/2}{l} \left(\frac{1}{2} \sum_{k=1}^{\infty} \frac{1}{k!} \varepsilon^k \right)^l$$

$$= \sqrt{2} \left(1 + \frac{1}{4} x + \frac{3}{32} x^2 + \frac{7}{384} x^3 + \frac{3}{2048} x^4 + \cdots \right).$$

∎

Problem 13 (3.4.5) *For ε small find an expression to compute*

$$\frac{\varepsilon - \sin \varepsilon}{\varepsilon - \tan \varepsilon}.$$

Solution. First note that

$$\frac{\varepsilon - \sin \varepsilon}{\varepsilon - \tan \varepsilon} = \frac{\varepsilon \cos \varepsilon - \sin \varepsilon \cos \varepsilon}{\varepsilon \cos \varepsilon - \sin \varepsilon} = \frac{\varepsilon \cos \varepsilon - 2^{-1} \sin(2\varepsilon)}{\varepsilon \cos \varepsilon - \sin \varepsilon}.$$

10

Now

$$\varepsilon \cos \varepsilon - \sin \varepsilon = \varepsilon \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k)!} \varepsilon^{2k} - \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!} \varepsilon^{2k+1}$$

$$= \varepsilon \sum_{k=0}^{\infty} (-1)^k \left(\frac{1}{(2k)!} - \frac{1}{(2k+1)!} \right) \varepsilon^{2k}$$

$$= \varepsilon \sum_{k=1}^{\infty} \frac{(-1)^k 2k}{(2k+1)!} \varepsilon^{2k}$$

and similarly

$$\varepsilon \cos \varepsilon - 2^{-1} \sin(2\varepsilon) = \varepsilon \sum_{j=0}^{\infty} \frac{(-1)^j}{(2j)!} \varepsilon^{2j} - \frac{1}{2} \sum_{j=0}^{\infty} \frac{(-1)^j}{(2j+1)!} (2\varepsilon)^{2j+1}$$

$$= \varepsilon \sum_{j=0}^{\infty} (-1)^j \left(\frac{1}{(2j)!} - \frac{2^{2j}}{(2j+1)!} \right) \varepsilon^{2j}$$

$$= \varepsilon \sum_{j=1}^{\infty} \frac{(-1)^j (2j+1-2^{2j})}{(2j+1)!} \varepsilon^{2j}.$$

By the binomial theorem

$$\frac{\varepsilon^3}{\varepsilon \cos \varepsilon - \sin \varepsilon} = \frac{1}{\sum_{k=1}^{\infty} \frac{(-1)^k 2k}{(2k+1)!} \varepsilon^{2k-2}}$$

$$= \frac{1}{-\frac{1}{3} + \sum_{k=2}^{\infty} \frac{(-1)^k 2k}{(2k+1)!} \varepsilon^{2k-2}} = \frac{-3}{1+\delta} = -3 \sum_{l=0}^{\infty} \binom{-1}{l} \delta^l$$

where

$$\delta = -3 \sum_{k=2}^{\infty} \frac{(-1)^k 2k}{(2k+1)!} \varepsilon^{2k-2}.$$

It follows that

$$\frac{\varepsilon \cos \varepsilon - 2^{-1} \sin(2\varepsilon)}{\varepsilon \cos \varepsilon - \sin \varepsilon} = -3 \sum_{j=1}^{\infty} \frac{(-1)^j (2j+1-2^{2j})}{(2j+1)!} \varepsilon^{2j-2} \sum_{l=0}^{\infty} \binom{-1}{l} \delta^l$$

$$= -3 \sum_{j=1}^{\infty} \frac{(-1)^j (2j+1-2^{2j})}{(2j+1)!} \varepsilon^{2j-2} \sum_{l=0}^{\infty} \binom{-1}{l} \left(-3 \sum_{k=2}^{\infty} \frac{(-1)^k 2k}{(2k+1)!} \varepsilon^{2k-2} \right)^l$$

$$= -\frac{1}{2} + \frac{9}{40} \varepsilon^2 - \frac{27}{2800} \varepsilon^4 + \frac{27}{112000} \varepsilon^6 - \cdots .$$

11

■

Problem 14 (3.4.6) *For x large find an expression to compute*

$$(x+1)^{1/n} - x^{1/n}.$$

Solution. Factoring and applying the binomial theorem yields

$$\begin{aligned}
(x+1)^{1/n} - x^{1/n} &= x^{1/n}\big((1+1/x)^{1/n} - 1\big)\\
&= x^{1/n}\left\{\left(\sum_{l=0}^{\infty}\binom{1/n}{l}(1/x)^l\right) - 1\right\}\\
&= x^{1/n}\sum_{l=1}^{\infty}\binom{1/n}{l}(1/x)^l\\
&= x^{1/n}\Big(\frac{1}{nx} + \frac{n-1}{2(nx)^2} + \frac{(n-1)(2n-1)}{6(nx)^3}\\
&\qquad + \frac{(n-1)(2n-1)(3n-1)}{24(nx)^4} + \cdots\Big).
\end{aligned}$$

■

Problem 15 (4.7.1) *Apply Newton's method using $x_0 = 1/2$ to solve*

$$xe^x - 1 = 0.$$

Solution. After computing

$$f(x) = xe^x - 1 \qquad \text{and} \qquad f'(x) = (1+x)e^x,$$

I wrote a program in C that looked like

```c
#include <stdio.h>
#include <math.h>

double f(double x){
        return x*exp(x)-1;
}
double df(double x){
        return (x+1)*exp(x);
}
double g(double x){
```

```
11          return x-f(x)/df(x);
12  }
13  int main(){
14          double x=0.5;
15          printf("#%10s %20s\n","n","x_n");
16          for(int n=0;n<7;n++){
17                  printf("%11d %20.14g\n",n,x);
18                  x=g(x);
19          }
20          return 0;
21  }
```

The output was

```
#          n                   x_n
           0                   0.5
           1     0.57102043980842
           2     0.56715556874411
           3     0.56714329053326
           4     0.56714329040978
           5     0.56714329040978
           6     0.56714329040978
```

This shows $x \approx 0.56714329040978$. ■

Problem 16 (4.7.2) *Apply Newton's method using $x_0 = 1$ to solve*

$$\arctan(x) - 1 = 0.$$

Solution. After computing

$$f(x) = \arctan(x) - 1 \qquad \text{and} \qquad f'(x) = \frac{1}{1 + x^2},$$

I wrote a program in C that looked like

```
1   #include <stdio.h>
2   #include <math.h>
3
4   double f(double x){
5           return atan(x)-1;
6   }
7   double df(double x){
8           return 1/(1+x*x);
9   }
10  double g(double x){
```

```
11          return x-f(x)/df(x);
12  }
13  int main(){
14          double x=1.0;
15          printf("#%10s %20s\n","n","x_n");
16          for(int n=0;n<7;n++){
17                  printf("%11d %20.14g\n",n,x);
18                  x=g(x);
19          }
20          return 0;
21  }
```

The output was

```
#           n                   x_n
            0                     1
            1         1.4292036732051
            2         1.5500620800555
            3          1.557383215089
            4         1.5574077243818
            5         1.5574077246549
            6         1.5574077246549
```

This shows $x \approx 1.5574077246549$. ∎

Problem 17 (4.7.3) *Apply Newton's method using $x_0 = 10$ to solve*

$$\ln(x) - 3 = 0.$$

Solution. After computing

$$f(x) = \log(x) - 3 \qquad \text{and} \qquad f'(x) = \frac{1}{x},$$

I wrote a program in C that looked like

```
1   #include <stdio.h>
2   #include <math.h>
3
4   double f(double x){
5           return log(x)-3;
6   }
7   double df(double x){
8           return 1/x;
9   }
10  double g(double x){
11          return x-f(x)/df(x);
```

```
12   }
13   int main(){
14           double x=10.0;
15           printf("#%10s %20s\n","n","x_n");
16           for(int n=0;n<7;n++){
17                   printf("%11d %20.14g\n",n,x);
18                   x=g(x);
19           }
20           return 0;
21   }
```

The output was

```
#          n                   x_n
           0                    10
           1          16.97414907006
           2         19.831041895899
           3          20.08391777272
           4         20.085536857924
           5         20.085536923188
           6         20.085536923188
```

This shows $x \approx 20.085536923188.$ ∎