1. [Burden and Faires, Section 1.1 Problem 1]. Show that the following equations have at least one solution in the given intervals.

a. $x\cos x - 2x^2 + 3x - 1 = 0$ on $[0.2, 0.3]$ and $[1.2, 1.3]$.

Let $f(x) = x\cos x - 2x^2 + 3x - 1$. As f is continuous, by the intermediate value theorem it is sufficient to show that $f(0.2)$ and $f(0.3)$ have different signs in order to show there is at least one solution in $[0.2, 0.3]$. Since

$$f(0.2) \approx -.2839866844 < 0 \qquad \text{and} \qquad f(0.3) \approx 0.006600947 > 0$$

the result follows that $f(x) = 0$ for some $x \in (0.2, 0.3)$. Similarly,

$$f(1.2) \approx 0.154829305 > 0 \qquad \text{and} \qquad f(1.3) \approx -.132251523 < 0$$

show also that $f(x) = 0$ for some $x \in (1.2, 1.3)$.

b. $(x-2)^2 - \log x = 0$ on $[1, 2]$ and $[e, 4]$.

Let $f(x) = (x-2)^2 - \log x = 0$. Again, as f is continuous, then

$$f(1) = 1 > 0 \qquad \text{and} \qquad f(2) \approx -0.6931471806 < 0$$

show by the intermediate value theorem that $f(x) = 0$ for some $x \in (1, 2)$. Similarly,

$$f(e) \approx -.4840712154 \qquad \text{and} \qquad f(4) \approx 2.613705639$$

show also that $f(x) = 0$ for some $x \in (e, 4)$.

c. $2x\cos(2x) - (x-2)^2 = 0$ on $[2, 3]$ and $[3, 4]$.

Let $f(x) = 2x\cos(2x) - (x-2)^2$. Again, as f is continuous, then

$$f(2) \approx -2.614574484 < 0 \qquad \text{and} \qquad f(3) \approx 4.761021720 > 0$$

show by the intermediate value theorem that $f(x) = 0$ for some $x \in (2, 3)$. Similarly,

$$f(3) \approx 4.761021720 > 0 \qquad \text{and} \qquad f(4) \approx -5.164000270 < 0$$

show also that $f(x) = 0$ for some $x \in (3, 4)$.

d. $x - (\log x)^x = 0$ on $[4, 5]$.

Let $f(x) = x - (\log x)^x$. Again, as f is continuous, then

$$f(4) \approx 0.306638424 > 0 \qquad \text{and} \qquad f(5) \approx -5.79869156 < 0$$

show by the intermediate value theorem that $f(x) = 0$ for some $x \in (4, 5)$.

2. [Burden and Faires, Section 1.1 Problem 6]. Suppose $f \in C[a,b]$ and $f'(x)$ exists on (a,b). Show that if $f'(x) \neq 0$ for all $x \in (a,b)$, then there can exist at most one number $p \in [a,b]$ with $f(p) = 0$.

Suppose, for contradiction, that there were two points p_1 and p_2 in $[a,b]$ such that

$$f(p_1) = 0 \qquad \text{and} \qquad f(p_2) = 0.$$

We may assume $p_1 < p_2$ without loss of generality by relabeling the subscripts if necessary. From Rolle's theorem there would then exist $\xi \in (p_1, p_2)$ such that $f'(\xi) = 0$. Since $(p_1, p_2) \subseteq (a,b)$ then $\xi \in (a,b)$. However, this would then contradict the hypothesis that $f(x) \neq 0$ for all $x \in (a,b)$. Therefore, there is at most one $p \in [a,b]$ with $f(p) = 0$.

3. [Burden and Faires, Section 1.1 Problem 18]. Let $f(x) = (1-x)^{-1}$ and $x_0 = 0$. Find the n-th Taylor polynomial $P_n(x)$ for $f(x)$ about x_0. Find a value of n necessary for $P_n(x)$ to approximate $f(x)$ to within 10^{-6} on $[0, 0.5]$.

Differentiating yields

$$\begin{aligned} f'(x) &= (1-x)^{-2} \\ f''(x) &= 2(1-x)^{-3} \\ f'''(x) &= 3!(1-x)^{-4} \\ &\vdots \\ f^{(n)}(x) &= n!(1-x)^{-n-1}. \end{aligned}$$

Therefore

$$P_n(x) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k = 1 + x + x^2 + \cdots + x^n.$$

Upon summing the geometric it follows that

$$\begin{aligned} R_n(x) = f(x) - P_n(x) &= \frac{1}{1-x} - (1 + x + x^2 + \cdots + x^n) \\ &= \frac{1}{1-x} - \frac{1 - x^{n+1}}{1-x} = \frac{x^{n+1}}{1-x}. \end{aligned}$$

Now

$$R_n(x)' = \frac{x^n}{(1-x)^2}(n + 1 - nx) = 0 \qquad \text{when} \qquad x = \frac{n+1}{n}.$$

Since $(n+1)/n > 1$, this that $R_n(x)$ is increasing on $[0,1]$. Since $R_n(0) = 0$ then

$$\max\left\{\, |R_n(x)| : x \in [0, 0.5] \,\right\} = R_n(0.5) = (0.5)^n \leq 10^{-6}$$

when

$$n \geq \log(10^{-6}) / \log(0.5) \approx 19.93156857.$$

Therefore $n = 20$ is sufficient. Since the formula for remainder used here is exact, then $n = 19$ will not suffice. Therefore $n = 20$ is also necessary.

4. [Burden and Faires, Section 1.1 Problem 19]. Let $f(x) = e^x$ and $x_0 = 0$. Find the n-th Taylor polynomial $P_n(x)$ for $f(x)$ about x_0. Find a value of n necessary for $P_n(x)$ to approximate $f(x)$ to within 10^{-6} on $[0, 0.5]$.

Differentiating yields $f^{(n)}(x) = e^x$ for all n. Therefore

$$P_n(x) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k = \sum_{k=0}^{n} \frac{x^k}{k!}.$$

and

$$R_n(x) = \int_{x_0}^{x} \frac{(x-t)^n}{n!} f^{n+1}(t)\, dt = \int_{0}^{x} \frac{(x-t)^n}{n!} e^t\, dt.$$

To find a sufficient choice for n estimate the remainder as

$$\begin{aligned} |R_n(x)| &= \int_0^x \frac{(x-t)^n}{n!} e^t\, dt \le \max\left\{e^t : t \in [0, 0.5]\right\}\Big(\int_0^x \frac{(x-t)^n}{n!}\, dt\Big) \\ &= e^{0.5} \int_0^x \frac{(x-t)^n}{n!}\, dt = e^{0.5} \frac{x^{n+1}}{(n+1)!} \le e^{0.5} \frac{(0.5)^{n+1}}{(n+1)!}. \end{aligned}$$

Since

$$|R_7(x)| \le e^{0.5} \frac{(0.5)^{n+1}}{(n+1)!}\bigg|_{n=7} \approx 1.597300958 \times 10^{-7} < 10^{-6},$$

then $n = 7$ is sufficient. To show $n = 7$ is necessary note that

$$|R_n(0.5)| = |\exp(0.5) - P_6(0.5)| \approx 1.653000 \times 10^{-6} \ge 10^{-6}.$$

5. [Burden and Faires, Section 1.1 Problem 24]. Verify $|\sin x| \leq |x|$ using the following two steps:

a. Show that for all $x \geq 0$ we have $f(x) = x - \sin x$ is non-decreasing, which implies that $\sin x \leq x$ with equality only when $x = 0$.

Differentiating and using the fact that $|\cos x| \leq 1$ yields

$$f'(x) = 1 - \cos x \geq 1 - |\cos x| \geq 0.$$

Therefore $f(x)$ is non-decreasing. It follows that

$$f(x) \geq f(0) = 0 - \sin 0 = 0 \qquad \text{for} \qquad x \geq 0.$$

Consequently

$$\sin x \leq x \qquad \text{for} \qquad x \geq 0. \tag{5.1}$$

b. Use the fact that the sine function is odd to reach the conclusion.

Consider the function $g(x) = x + \sin(x)$. Then

$$g'(x) = 1 + \cos(x) \geq 1 - |\cos(x)| \geq 0.$$

Therefore $g(x)$ is non-decreasing. It follows that

$$g(x) \geq g(0) = 0 + \sin 0 = 0 \qquad \text{for} \qquad x \geq 0.$$

Consequently

$$-x \leq \sin x \qquad \text{for} \qquad x \geq 0. \tag{5.2}$$

Combining (5.1) with (5.2) implies that

$$|\sin x| \leq x \qquad \text{for} \qquad x \geq 0. \tag{5.3}$$

Now let $y = -x$. Then $x \geq 0$ when $y \leq 0$ and (5.3) implies

$$|\sin y| = |\sin(-x)| = |-\sin x| = |\sin x| \leq x = -y = |y|.$$

In other words

$$|\sin x| \leq |x| \qquad \text{for} \qquad x \in \mathbf{R}.$$

6. [Burden and Faires, Section 1.1 Problem 26]. The *error function* defined by

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt$$

gives the probability that any one of a series of trials will lie within x units of the mean, assuming that the trials have a normal distribution with mean 0 and standard deviation $\sqrt{2}/2$. This integral cannot be evaluated in terms of elementary functions, so an approximating technique must be used.

a. Integrate the Maclaurin series for e^{-x^2} to show that

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \sum_{k=0}^{\infty} \frac{(-1)^k x^{2k+1}}{(2k+1)k!}.$$

Since

$$e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!}$$

then

$$e^{-t^2} = \sum_{k=0}^{\infty} \frac{(-t^2)^k}{k!} = \sum_{k=0}^{\infty} \frac{(-1)^k t^{2k}}{k!}.$$

Consequently

$$\begin{aligned}
\operatorname{erf}(x) &= \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt = \frac{2}{\sqrt{\pi}} \int_0^x \sum_{k=0}^{\infty} \frac{(-1)^k t^{2k}}{k!}\, dt \\
&= \frac{2}{\sqrt{\pi}} \sum_{k=0}^{\infty} \int_0^x \frac{(-1)^k t^{2k}}{k!}\, dt = \frac{2}{\sqrt{\pi}} \sum_{k=0}^{\infty} \frac{(-1)^k x^{2k+1}}{(2k+1)k!} \\
&= \frac{2}{\sqrt{\pi}} \Big(x - \frac{1}{3}\, x^3 + \frac{1}{10}\, x^5 - \frac{1}{42}\, x^7 + \frac{x^9}{216} + \cdots \Big).
\end{aligned}$$

b. The error function can also be expressed in the form

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} e^{-x^2} \sum_{k=0}^{\infty} \frac{2^k x^{2k+1}}{1 \cdot 3 \cdot 5 \cdots (2k+1)}.$$

Verify that the two series agree for $k = 1$, 2, 3 and 4.

Let

$$S_1 = e^{-x^2} \qquad \text{and} \qquad S_2 = \sum_{k=0}^{\infty} \frac{2^k x^{2k+1}}{1 \cdot 3 \cdot 5 \cdots (2k+1)}.$$

Multiplying the series for S_1 by S_2 yields

$$\begin{array}{rrcccccccc}
 & S_1 & = 1 - & x^2 & + \frac{1}{2}x^4 & - \frac{1}{6}x^6 & + \frac{1}{24}x^8 & + \cdots \\
\times & S_2 & = x + \frac{2}{3}x^3 + & \frac{4\,x^5}{15} & + \frac{8\,x^7}{105} & + \frac{16\,x^9}{945} & + \cdots & \\
\hline
 & & x - & x^3 & + \frac{1}{2}x^5 & - \frac{1}{6}x^7 & + \frac{1}{24}x^9 & + \cdots \\
 & & & \frac{2}{3}x^3 - & \frac{2}{3}x^5 & + \frac{1}{3}x^7 & - \frac{1}{9}x^9 & + \cdots \\
 & & & & \frac{4\,x^5}{15} & - \frac{4\,x^7}{15} & + \frac{2}{15}x^9 & + \cdots \\
 & & & & & \frac{8\,x^7}{105} & - \frac{8\,x^9}{105} & + \cdots \\
+ & & & & & & \frac{16\,x^9}{945} & + \cdots \\
\hline
 & S_1S_2 & = x - \frac{1}{3}x^3 + & \frac{1}{10}x^5 & - \frac{1}{42}x^7 & + \frac{x^9}{216} & + \cdots &
\end{array}$$

Since the first terms of the series in part (a) are

$$\sum_{k=0}^{\infty} \frac{(-1)^k x^{2k+1}}{(2k+1)k!} = x - \frac{1}{3}x^3 + \frac{1}{10}x^5 - \frac{1}{42}x^7 + \frac{x^9}{216} + \cdots$$

we have verified that the series agree for $k = 0$, 1, 2, 3 and 4.

c. Use the series in part (a) to approximate $\operatorname{erf}(1)$ to within 10^{-7}.

Since

$$\operatorname{erf}(1) - \frac{2}{\sqrt{\pi}} \sum_{k=0}^{8} \frac{(-1)^k}{(2k+1)k!} \approx -1.499924 \times 10^{-0}$$

and

$$\operatorname{erf}(1) - \frac{2}{\sqrt{\pi}} \sum_{k=0}^{9} \frac{(-1)^k}{(2k+1)k!} \approx 1.366608 \times 10^{-8}$$

we conclude that $n = 9$ is necessary to approximate $\operatorname{erf}(1)$ to within 10^{-7}.

d. Use the same number of terms as in part (c) to approximate $\operatorname{erf}(1)$ with the series in part (b).

Computing yields

$$\operatorname{erf}(1) - \frac{2}{\sqrt{\pi}} e^{-1} \sum_{k=0}^{9} \frac{2^k}{1 \cdot 3 \cdot 5 \cdots (2k+1)} \approx 3.383623 \times 10^{-8}.$$

e. Explain why difficulties occur using the series in part (b) to approximate $\operatorname{erf}(x)$.

In reality there were not many difficulties with either series. Both have an exponential term in the denominator and converge quickly; however, the second series also has a factor 2^k in the numerator which slows convergence somewhat and leads to slightly larger errors bounds. It should be pointed out that series (a) is an alternating series such that every other term is either positive or negative. Such series converge in such a way that each partial sum is either too big or too small depending on whether the last term in the sum was positive or negative. This allows for simple error estimates of the form

$$\frac{2}{\sqrt{\pi}} \sum_{k=0}^{2n+1} \frac{(-1)^k x^{2k+1}}{(2k+1)k!} \leq \operatorname{erf}(x) \leq \frac{2}{\sqrt{\pi}} \sum_{k=0}^{2n} \frac{(-1)^k x^{2k+1}}{(2k+1)k!}$$

for every n. Equivalently the remainder is bounded by the next term in the series as

$$\left| \operatorname{erf}(x) - \frac{2}{\sqrt{\pi}} \sum_{k=0}^{n} \frac{(-1)^k x^{2k+1}}{(2k+1)k!} \right| \leq \frac{|x|^{2n+3}}{(2n+3)(n+1)!}$$

On the other hand, assuming $x > 0$, the second series consists of all positive terms. Therefore, the second series is always less than the actual value of $\operatorname{erf}(x)$. This can make bounds similar to those given above for the first series more difficult to obtain.

To illustrate the difference between the two series numerically a C program was written to compute a table of approximations for different values of n. Note that in all cases, the series from part (b) performs slightly worse in terms of error then the series from part (a).

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

/*  Solution to Section 1.1 Problem 26cd in Burden and Faires.
    Compute error function using two different Maclaurin series.
    Written Sept 20, 2016 by Eric Olson for Math/CS 466/666. */

double Pn_a(double x, int n){

    /* Pn_a(x,n) = 2/sqrt(pi) * sum_{k=0}^{n} (-1)^k x^{2k+1} / ((2k+1)k!)  */

    int k;
    double x2=x*x, ak=x, S=x;
    for(k=1;k<=n;k++){
        ak*=-x2/k;
        S+=ak/(2*k+1);
    }
    return 2*S/sqrt(M_PI);
}

double Pn_b(double x, int n){

    /* Pn_b(x,n) = 2/sqrt(pi) e^{-x^2} * sum_{k=0}^{n} 2^k x^{2k+1} / (1*3*5*...*(2k+1))  */

    int k;
    double x2=x*x, ak=x, S=x;
    for(k=1;k<=n;k++){
        ak*=2*x2/(2*k+1);
        S+=ak;
    }
    return 2*S*exp(-x2)/sqrt(M_PI);
}

int main(){
    int n;
    double erf1=erf(1);
    printf("%3s %17s %17s %17s %17s\n",
        "n","Pn_a","Rn_a","Pn_b","Rn_b");
    for(n=1;n<10;n++){
        double pna=Pn_a(1,n),pnb=Pn_b(1,n);
        printf("%3d %17.10e %17.10e %17.10e %17.10e\n",
            n,pna,erf1-pna,pnb,erf1-pnb);
```

```
47      }
48      return 0;
49  }
```

The output from running the program was

```
n              Pn_a              Rn_a              Pn_b              Rn_b
1  7.5225277806e-01  9.0448014886e-02  6.9184582903e-01  1.5085496392e-01
2  8.6509069477e-01 -2.2389901824e-02  8.0254116168e-01  4.0159631270e-02
3  8.3822452413e-01  4.4762688216e-03  8.3416839958e-01  8.5323933712e-03
4  8.4344850175e-01 -7.4770880382e-04  8.4119667467e-01  1.5041182826e-03
5  8.4259366905e-01  1.0712389852e-04  8.4247454287e-01  2.2625008466e-04
6  8.4271422238e-01 -1.3429431295e-05  8.4267113797e-01  2.9654977282e-05
7  8.4269929673e-01  1.4962190631e-06  8.4269735065e-01  3.4422962992e-06
8  8.4270094294e-01 -1.4999237341e-07  8.4270043450e-01  3.5845147772e-07
9  8.4270077928e-01  1.3666073384e-08  8.4270075911e-01  3.3836233371e-08
```

A comparison of remainders `Rn_a` for series (a) with the remainders `Rn_b` for series (b) illustrates that series (a) is more accurate for each value of n tested.

7. [Burden and Faires, Section 1.2 Problem 4]. Perform the following computations (i) exactly, (ii) using three-digit chopping arithmetic, and (iii) using three-digit rounding arithmetic. (iv) Compute the relative effors in part (ii) and (iii).

a. $\frac{4}{5} + \frac{1}{3}$.

(i) Exact computation

$$\frac{4}{5} + \frac{1}{3} = \frac{17}{15}.$$

(ii) Using three-digit chopping arithmetic

$$(4 \div 5) \oplus (1 \div 3) = 0.800 \oplus 0.333 = 1.13.$$

(iii) Using three-digit rounding arithmetic

$$(4 \div 5) \oplus (1 \div 3) = 0.800 \oplus 0.333 = 1.13.$$

(iv) The relative errors in part (ii) and (iii)

$$E_{\text{chop}} = E_{\text{round}} = \frac{|17/15 - 1.13|}{|17/15|} \approx 0.002941176176.$$

b. $\frac{4}{5} \cdot \frac{1}{3}$.

(i) Exact computation

$$\frac{4}{5} \cdot \frac{1}{3} = \frac{4}{15}.$$

(ii) Using three-digit chopping arithmetic

$$(4 \div 5) \otimes (1 \div 3) = 0.800 \otimes 0.333 = 2.66.$$

(iii) Using three-digit rounding arithmetic

$$(4 \div 5) \otimes (1 \div 3) = 0.800 \otimes 0.333 = 0.266.$$

(iv) The relative errors in part (ii) and (iii)

$$E_{\text{chop}} = E_{\text{round}} = \frac{|4/15 - 0.266|}{|4/15|} \approx 0.002500000125.$$

c. $\left(\frac{1}{3} - \frac{3}{11}\right) + \frac{3}{20}$.

(i) Exact computation

$$\left(\frac{1}{3} - \frac{3}{11}\right) + \frac{3}{20} = \frac{139}{660}.$$

(ii) Using three-digit chopping arithmetic

$$\begin{aligned}\big((1 \div 3) \ominus (3 \div 11)\big) \oplus (3 \div 20) &= (0.333 \ominus 0.272) \oplus 0.150 \\ &= 0.061 \oplus 0.150 = 0.211.\end{aligned}$$

(iii) Using three-digit rounding arithmetic

$$\begin{aligned}\big((1 \div 3) \ominus (3 \div 11)\big) \oplus (3 \div 20) &= (0.333 \ominus 0.273) \oplus 0.150 \\ &= 0.060 \oplus 0.150 = 0.210.\end{aligned}$$

(iv) The relative errors in part (ii) and (iii)

$$E_{\text{chop}} = \frac{|139/660 - 0.211|}{|139/660|} \approx 0.001870503626.$$

and

$$E_{\text{round}} = \frac{|139/660 - 0.210|}{|139/660|} \approx 0.002877697813.$$

d. $\left(\frac{1}{3} + \frac{3}{11}\right) + \frac{3}{20}$.

(i) Exact computation

$$\left(\frac{1}{3} + \frac{3}{11}\right) + \frac{3}{20} = \frac{301}{660}.$$

(ii) Using three-digit chopping arithmetic

$$\begin{aligned}\big((1 \div 3) \oplus (3 \div 11)\big) \ominus (3 \div 20) &= (0.333 \oplus 0.272) \ominus 0.150 \\ &= 0.605 \ominus 0.150 = 0.455.\end{aligned}$$

(iii) Using three-digit rounding arithmetic

$$\begin{aligned}\big((1 \div 3) \oplus (3 \div 11)\big) \ominus (3 \div 20) &= (0.333 \oplus 0.273) \ominus 0.150 \\ &= 0.606 \ominus 0.150 = 0.456.\end{aligned}$$

(iv) The relative errors in part (ii) and (iii)

$$E_{\text{chop}} = \frac{|301/660 - 0.455|}{|301/660|} \approx 0.002325581482.$$

and

$$E_{\text{round}} = \frac{|301/660 - 0.456|}{|301/660|} \approx 0.0001328904518.$$

8. [Burden and Faires, Section 1.2 Problem 10]. The number e can be defined by $e = \sum_{n=0}^{\infty}(1/n!)$ where $n! = n(n-1)\cdots 2\cdot 1$ for $n \neq 0$ and $0! = 1$. Compute the absolute error and the relative error in the following approximations of e:

a. $\displaystyle\sum_{n=0}^{5} \frac{1}{n!}$.

Computing

$$\sum_{n=0}^{5} \frac{1}{n!} = \frac{163}{60}.$$

Therefore

$$E_{\text{abs}} = |e - 163/60| \approx 0.001615161.$$

and

$$E_{\text{rel}} = \frac{E_{\text{abs}}}{e} \approx 0.0005941845262.$$

b. $\displaystyle\sum_{n=0}^{10} \frac{1}{n!}$.

Computing

$$\sum_{n=0}^{10} \frac{1}{n!} = \frac{9864101}{3628800}.$$

Therefore

$$E_{\text{abs}} = |e - 9864101/3628800| \approx 2.73126607556 \times 10^{-8}.$$

and

$$E_{\text{rel}} = \frac{E_{\text{abs}}}{e} \approx 1.004776637569 \times 10^{-8}.$$

9. [Burden and Faires, Section 1.2 Problem 15]. Use the 64-bit long real format to find the decimal equivalent of the following floating-point machine numbers.

a. `0 10000001010 1001001100000000000000000000000000000000000000000000`

Since the first bit is `0` the number is positive. The next 11 bits `10000001010` represent the exponent in base two as

$$\text{exponent} = b - 1023 = 2^1 + 2^3 + 2^10 - 1023 = 11.$$

The final 52 bits represent the mantisa with an implied first digit of 1 as

$$\text{mantissa} = 1 + 2^{-1} + 2^{-4} + 2^{-7} + 2^{-8} = 1.57421875.$$

Therefore

$$x = 1.57421875 \times 2^{11} = 3224.$$

b. `1 10000001010 1001001100000000000000000000000000000000000000000000`

Since the first bit is `1` the number is negative. The next 11 bits `10000001010` represent the exponent in base two as

$$\text{exponent} = b - 1023 = 2^1 + 2^3 + 2^10 - 1023 = 11.$$

The final 52 bits represent the mantisa with an implied first digit of 1 as

$$\text{mantissa} = 1 + 2^{-1} + 2^{-4} + 2^{-7} + 2^{-8} = 1.57421875.$$

Therefore

$$x = -1.57421875 \times 2^{11} = -3224.$$

c. `0 01111111111 0101001100000000000000000000000000000000000000000000`

Since the first bit is `0` the number is positive. The next 11 bits `10000001010` represent the exponent in base two as

$$\begin{aligned}\text{exponent} = b - 1023 &= 1 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7 + 2^8 + 2^9 \\ &= 1023 - 1023 = 0.\end{aligned}$$

The final 52 bits represent the mantisa with an implied first digit of 1 as

$$\text{mantissa} = 1 + 2^{-2} + 2^{-4} + 2^{-7} + 2^{-8} = 1.32421875.$$

Therefore

$$x = 1.32421875 \times 2^0 = 1.32421875.$$

d. `0 01111111111 0101001100000000000000000000000000000000000000000000000001`

Since the first bit is `0` the number is positive. The next 11 bits `10000001010` represent the exponent in base two as

$$\begin{aligned}\text{exponent} = b - 1023 &= 1 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7 + 2^8 + 2^9 \\ &= 1023 - 1023 = 0.\end{aligned}$$

The final 52 bits represent the mantisa with an implied first digit of 1 as

$$\begin{aligned}\text{mantissa} &= 1 + 2^{-2} + 2^{-4} + 2^{-7} + 2^{-8} + 2^{-52} \\ &= 1.32421875000000022204460492503130808472633361816406 25.\end{aligned}$$

Therefore

$$\begin{aligned}x &= 1.32421875000000022204460492503130808472633361816406 25 \times 2^0 \\ &= 1.32421875000000022204460492503130808472633361816406 25.\end{aligned}$$

10. Find the next largest and smallest machine numbers in decimal form for the numbers in the previous exercise.

In each cases, since the mantissa contains both 1's and 0's there is no carry that could propagate in a way to change the exponent. Therefore, the next largest and smallest numbers can be obtained by adding and subtracting $2^{-52} \times 2^{\text{exponent}}$. Denoting by x_a, x_b, x_c and x_d the numbers from the four parts of the previous question and ϵ the amount added in each case to obtain the next largest and smallest machine numbers, we obtain

$$\begin{aligned}x_a + \epsilon &= 3224 + 2^{-52} \times 2^{11} \\ &= 3224.00000000000045474735088646411895751953125\end{aligned}$$

$$\begin{aligned}x_a - \epsilon &= 3224 - 2^{-52} \times 2^{11} \\ &= 3223.99999999999954525264911353588104248046875\end{aligned}$$

$$\begin{aligned}x_b + \epsilon &= -3224 + 2^{-52} \times 2^{11} \\ &= -3223.99999999999954525264911353588104248046875\end{aligned}$$

$$\begin{aligned}x_b - \epsilon &= -3224 + 2^{-52} \times 2^{11} \\ &= -3224.00000000000045474735088646411895751953125\end{aligned}$$

$$\begin{aligned}x_c + \epsilon &= 1.32421875 + 2^{-52} \\ &= 1.32421875000000022204460492503130808472633361816406 25\end{aligned}$$

$$\begin{aligned}x_c - \epsilon &= 1.32421875 - 2^{-52} \\ &= 1.32421874999999977795539507496869191527366638183593 75\end{aligned}$$

$$\begin{aligned}x_d + \epsilon &= 1.32421875000000022204460492503130808472633361816406 25 + 2^{-52} \\ &= 1.32421875000000044408920985006261616945266723632812 5\end{aligned}$$

$$\begin{aligned}x_d - \epsilon &= 1.32421875000000022204460492503130808472633361816406 25 - 2^{-52} \\ &= 1.32421875\end{aligned}$$

The following C program

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

typedef union {
    double x;
    unsigned long long b;
} rawbits;

char *ftob(double x){

    /*  Return a string with the bits corresponding to x */

    static char buf[sizeof(rawbits)*8+1];
    rawbits r;
    r.x=x;
    unsigned long long m;
    int j=sizeof(rawbits)*8;
    buf[j]=0;
    for(m=1;m;m<<=1) buf[--j]=(m&r.b)?'1':'0';
    return buf;
}

double btof(char buf[sizeof(rawbits)*8+1]){

    /*  Interpret the 0's and 1's in the string buf and set the
        corresponding bits in r.b.  Then return the double.  */

    rawbits r;
    r.b=0;
    unsigned long long m;
    int j=sizeof(rawbits)*8;
    for(m=1;m;m<<=1) if(buf[--j]=='1') r.b|=m;
    return r.x;
}

double epsplus(double x){

    /*  Perform a binary search to find the smallest number b
        that can be added to x and change its value.  */

    double a=0;
    double b=fabs(x);
```

```
    double c=0;
    for(;;){
        double cnew=(a+b)/2;
        if(c==cnew) return x+b;
        else c=cnew;
        double y=x+c;
        if(y==x) a=c;
        else b=c;
    }
}

double epsminus(double x){

    /*  Perform a binary search to find the smallest number b
        that can be subtracted from x and change its value.  */

    double a=0;
    double b=fabs(x);
    double c=0;
    for(;;){
        double cnew=(a+b)/2;
        if(c==cnew) return x-b;
        else c=cnew;
        double y=x-c;
        if(y==x) a=c;
        else b=c;
    }
}

char *pretty(char *p){

    /*  Add spaces to separate sign, exponenet and mantissa */

    static char buf[sizeof(rawbits)*8+3];
    int i;
    char *q=buf;
    *q=*p;
    for(i=1;i<sizeof(rawbits)*8+1;i++){
        switch(i){
case 1:
case 12:
            *++q=' ';
default:
            *++q=*++p;
```

```
88          }
89          if(!*p) break;
90      }
91      return buf;
92  }
93
94  int main(){
95      if(sizeof(double)!=sizeof(unsigned long long)){
96          fprintf(stderr,"Error %d != %d please fix!\n",
97              (int)sizeof(double),
98              (int)sizeof(unsigned long long));
99          exit(1);
100     }
101     char *input[] = {
102         "0100000010101001001100000000000000000000000000000000000000000000000000000000",
103         "1100000010101001001100000000000000000000000000000000000000000000000000000000",
104         "0011111111110101001100000000000000000000000000000000000000000000000000000000",
105         "0011111111110101001100000000000000000000000000000000000000000000000000000001",
106         0
107     };
108     char **p;
109     for(p=input;*p;){
110         printf("input=%s\n",pretty(*p));
111         double x=btof(*p);
112         printf("     =%+-.59e\n",x);
113         double xplus=epsplus(x);
114         double xminus=epsminus(x);
115         printf("x+eps=%s\n",pretty(ftob(xplus)));
116         printf("     =%+-.59e\n",xplus);
117         printf("x-eps=%s\n",pretty(ftob(xminus)));
118         printf("     =%+-.59e\n",xminus);
119         if(*++p) putchar('\n');
120     }
121     return 0;
122 }
```

with output

```
input=0 10000001010 1001001100000000000000000000000000000000000000000000000000
     =+3.22400000000000000000000000000000000000000000000000000000000000000000e+03
x+eps=0 10000001010 1001001100000000000000000000000000000000000000000000000001
     =+3.22400000000000045474735088646411895751953125000000000000000000e+03
x-eps=0 10000001010 1001001011111111111111111111111111111111111111111111111111
     =+3.22399999999999954525264911353588104248046875000000000000000000e+03

input=1 10000001010 1001001100000000000000000000000000000000000000000000000000
     =-3.22400000000000000000000000000000000000000000000000000000000000000000e+03
```

```
x+eps=1 10000001010 10010010111111111111111111111111111111111111111111111111
     =-3.22399999999999545252649113535881042480468750000000000000000e+03
x-eps=1 10000001010 10010011000000000000000000000000000000000000000000000001
     =-3.22400000000000045474735088646411895751953125000000000000000e+03

input=0 01111111111 01010011000000000000000000000000000000000000000000000000
     =+1.32421875000000000000000000000000000000000000000000000000000e+00
x+eps=0 01111111111 01010011000000000000000000000000000000000000000000000001
     =+1.32421875000000022204460492503130808472633361816406250000000e+00
x-eps=0 01111111111 01010010111111111111111111111111111111111111111111111111
     =+1.32421874999999977795539507496869191527366638183593750000000e+00

input=0 01111111111 01010011000000000000000000000000000000000000000000000001
     =+1.32421875000000022204460492503130808472633361816406250000000e+00
x+eps=0 01111111111 01010011000000000000000000000000000000000000000000000010
     =+1.32421875000000044408920985006261616945266723632812500000000e+00
x-eps=0 01111111111 01010011000000000000000000000000000000000000000000000000
     =+1.32421875000000000000000000000000000000000000000000000000000e+00
```

can be used to verify the above calculations.

11. [Burden and Faires, Section 1.2 Problem 19]. The two-by-two linear system

$$\begin{cases} ax + by = e \\ cx + dy = f \end{cases}$$

where a, b, c, d, e and f are given, can be solved for x and y as follows:

$$m = c/a, \quad d_1 = d - mb, \quad f_1 = f - me, \quad y = f_1/d_1 \quad \text{and} \quad x = (e - by)/a.$$

Solve the following linear systems using four-digit rounding arithmetic.

a. $\begin{cases} 1.130x - 6.990y = 14.20 \\ 1.013x - 6.099y = 14.22 \end{cases}$

Computing yields

$$\begin{aligned} m &= 1.013 \div 1.130 = 0.8965 \\ d_1 &= -6.099 \ominus (0.8965 \otimes -6.990) = -6.099 \ominus -6.267 = 0.1680 \\ f_1 &= 14.22 \ominus (0.8965 \otimes 14.20) = 14.22 \ominus 12.73 = 1.490 \\ y &= 1.490 \div 0.1680 = 8.869 \\ x &= \big(14.20 \ominus (-6.990 \otimes 8.869)\big) \div 1.130 \\ &= (14.20 \ominus -61.99) \div 1.130 = 76.19 \div 1.130 = 67.42. \end{aligned}$$

b. $\begin{cases} 8.110x + 12.20y = -0.1370 \\ -18.11x + 112.2y = -0.1376 \end{cases}$

Computing yields

$$m = -2.233, \quad d_1 = 139.4, \quad f_1 = -0.4435, \quad y = -0.003181, \quad x = -0.01211.$$

12. [Burden and Faires, Section 1.2 Problem 20]. Repeat the above exercise using four-digit chopping arithmetic.

For part (a) we obtain

$$m = 0.8964, \quad d1 = 0.166, \quad f1 = 1.50, \quad y = 9.036, \quad x = 68.46$$

and for part (b) we obtain

$$m = -2.233, \quad d1 = 139.4, \quad f1 = -0.4435, \quad y = -0.003181, \quad x = -0.01210.$$

As Maple implements the IEEE 854 standard for base-ten arithemetic the above computations can be easily done using Maple. The script

```
1  restart;
2  kernelopts(printbytes=false):
3  Digits:=4;
4  Rounding:=nearest;
5  # Part (a) using 4-digit rounding arithmetic
6  a:=1.130; b:=-6.990; c:=1.013; d:=-6.099; e:=14.20; f:=14.22;
7  m:=c/a; d1:=d-m*b; f1:=f-m*e; y:=f1/d1; x:=(e-b*y)/a;
8  # Part (b) using 4-digit rounding arithmetic
9  a:=8.11; b:=12.2; c:=-18.11; d:=112.2; e:=-0.1370; f:=-0.1376;
10 m:=c/a; d1:=d-m*b; f1:=f-m*e; y:=f1/d1; x:=(e-b*y)/a;
11 Rounding:=0;
12 # Part (a) using 4-digit chopping arithmetic
13 a:=1.130; b:=-6.990; c:=1.013; d:=-6.099; e:=14.20; f:=14.22;
14 m:=c/a; d1:=d-m*b; f1:=f-m*e; y:=f1/d1; x:=(e-b*y)/a;
15 # Part (b) using 4-digit chopping arithmetic
16 a:=8.11; b:=12.2; c:=-18.11; d:=112.2; e:=-0.1370; f:=-0.1376;
17 m:=c/a; d1:=d-m*b; f1:=f-m*e; y:=f1/d1; x:=(e-b*y)/a;
```

produces the output

```
         |\^/|     Maple 9.5 (IBM INTEL LINUX)
      ._|\|   |/|_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2004
       \  MAPLE  /  All rights reserved. Maple is a trademark of
       <____ ____>  Waterloo Maple Inc.
            |       Type ? for help.
      > restart;
      > kernelopts(printbytes=false):
      > Digits:=4;
                                        Digits := 4

      > Rounding:=nearest;
                                    Rounding := nearest

      # Part (a) using 4-digit rounding arithmetic
      > a:=1.130; b:=-6.990; c:=1.013; d:=-6.099; e:=14.20; f:=14.22;
                                         a := 1.130
```

```
                                  b := -6.990

                                   c := 1.013

                                  d := -6.099

                                   e := 14.20

                                   f := 14.22

> m:=c/a; d1:=d-m*b; f1:=f-m*e; y:=f1/d1; x:=(e-b*y)/a;
                                  m := 0.8965

                                  d1 := 0.168

                                   f1 := 1.49

                                   y := 8.869

                                   x := 67.42

# Part (b) using 4-digit rounding arithmetic
> a:=8.11; b:=12.2; c:=-18.11; d:=112.2; e:=-0.1370; f:=-0.1376;
                                   a := 8.11

                                   b := 12.2

                                  c := -18.11

                                   d := 112.2

                                  e := -0.1370

                                  f := -0.1376

> m:=c/a; d1:=d-m*b; f1:=f-m*e; y:=f1/d1; x:=(e-b*y)/a;
                                  m := -2.233

                                   d1 := 139.4

                                 f1 := -0.4435

                                 y := -0.003181

                                 x := -0.01211

> Rounding:=0;
                                 Rounding := 0

# Part (a) using 4-digit chopping arithmetic
> a:=1.130; b:=-6.990; c:=1.013; d:=-6.099; e:=14.20; f:=14.22;
                                   a := 1.130
```

```
                                   b := -6.990

                                    c := 1.013

                                   d := -6.099

                                    e := 14.20

                                    f := 14.22

> m:=c/a; d1:=d-m*b; f1:=f-m*e; y:=f1/d1; x:=(e-b*y)/a;
                                   m := 0.8964

                                   d1 := 0.166

                                    f1 := 1.50

                                    y := 9.036

                                    x := 68.46

# Part (b) using 4-digit chopping arithmetic
> a:=8.11; b:=12.2; c:=-18.11; d:=112.2; e:=-0.1370; f:=-0.1376;
                                    a := 8.11

                                    b := 12.2

                                   c := -18.11

                                    d := 112.2

                                   e := -0.1370

                                   f := -0.1376

> m:=c/a; d1:=d-m*b; f1:=f-m*e; y:=f1/d1; x:=(e-b*y)/a;
                                   m := -2.233

                                    d1 := 139.4

                                  f1 := -0.4435

                                  y := -0.003181

                                  x := -0.01210

> quit
bytes used=360004, alloc=262096, time=0.02
```

Attached are additional calculations in the form of a Maple worksheet for earlier problems in this assignment.

```
> # HW1 Section 1.1 Problem 1a
> restart;
  f:=x->x*cos(x)-2*x^2+3*x-1;
```

$$f := x \to x\cos(x) - 2x^2 + 3x - 1 \quad (1)$$

```
> f(0.2);
```

$$-0.2839866844 \quad (2)$$

```
> f(0.3);
```

$$0.006600947 \quad (3)$$

```
> f(1.2);
```

$$0.154829305 \quad (4)$$

```
> f(1.3);
```

$$-0.132251523 \quad (5)$$

```
> # HW1 Section 1.1 Problem 1b
> restart;
  f:=x->(x-2)^2-log(x);
```

$$f := x \to (x-2)^2 - \log(x) \quad (6)$$

```
> f(1.0);
```

$$1.00 \quad (7)$$

```
> f(2.0);
```

$$-0.6931471806 \quad (8)$$

```
> f(exp(1.0));
```

$$-0.4840712154 \quad (9)$$

```
> f(4.0);
```

$$2.613705639 \quad (10)$$

```
> # HW1 Section 1.1 Problem 1c
> restart;
  f:=x->2*x*cos(2*x)-(x-2)^2;
```

$$f := x \to 2x\cos(2x) - (x-2)^2 \quad (11)$$

```
> f(2.0);
```

$$-2.614574484 \quad (12)$$

```
> f(3.0);
```

$$4.761021720 \quad (13)$$

```
> f(4.0);
```

$$-5.164000270 \quad (14)$$

```
> # HW1 Section 1.1 Problem 1d
> restart;
  f:=x->x-(log(x))^x;
```

$$f := x \to x - \log(x)^x \quad (15)$$

```
> f(4.0);
```

$$0.306638424 \quad (16)$$

```
> f(5.0);
```

$$-5.79869156 \tag{17}$$

```
> # HW1 Problem 18
> restart;
  Rn:=x^(n+1)/(1-x);
```

$$Rn := \frac{x^{n+1}}{1-x} \tag{18}$$

```
> dRn:=simplify(diff(Rn,x)/x^n);
```

$$dRn := -\frac{n x - n - 1}{(-1+x)^2} \tag{19}$$

```
> solve(dRn=0,x);
```

$$\frac{n+1}{n} \tag{20}$$

```
> log(10.0^(-6))/log(0.5);
```

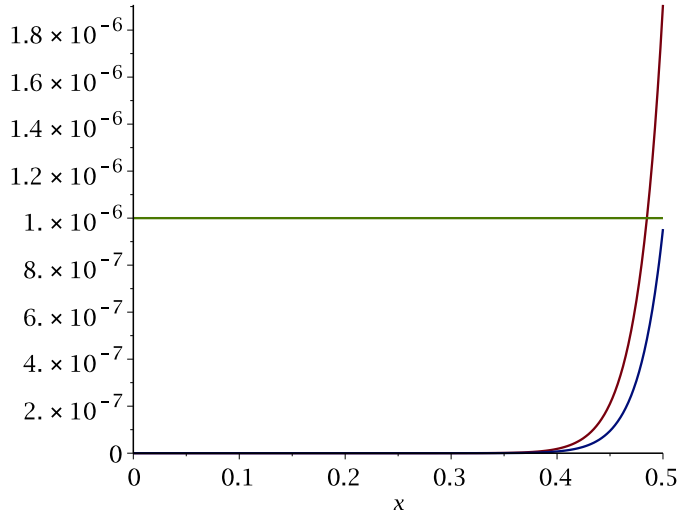
$$19.93156857 \tag{21}$$

```
> R19:=subs(n=19,Rn);
  R20:=subs(n=20,Rn);
```

$$R19 := \frac{x^{20}}{1-x}$$

$$R20 := \frac{x^{21}}{1-x} \tag{22}$$

```
> plot([R19,R20,10^(-6)],x=0..0.5);
```




```
> # HW1 Section 1.1 Problem 19
> restart;
  RnB:=exp(0.5)*(0.5)^(n+1)/(n+1)!;
```

$$RnB := \frac{1.648721271\ 0.5^{n+1}}{(n+1)!} \tag{23}$$

```
> # Note that above is only a bound on the remainder.
> evalf(subs(n=7,RnB));
```

$$1.597300958\ 10^{-7} \tag{24}$$

```
> Pn:=(n,x)->sum(x^k/k!,k=0..n);
```

$$Pn := (n, x) \rightarrow \sum_{k=0}^{n} \frac{x^k}{k!} \tag{25}$$

```
> R6:=exp(x)-Pn(6,x);
  R7:=exp(x)-Pn(7,x);
```

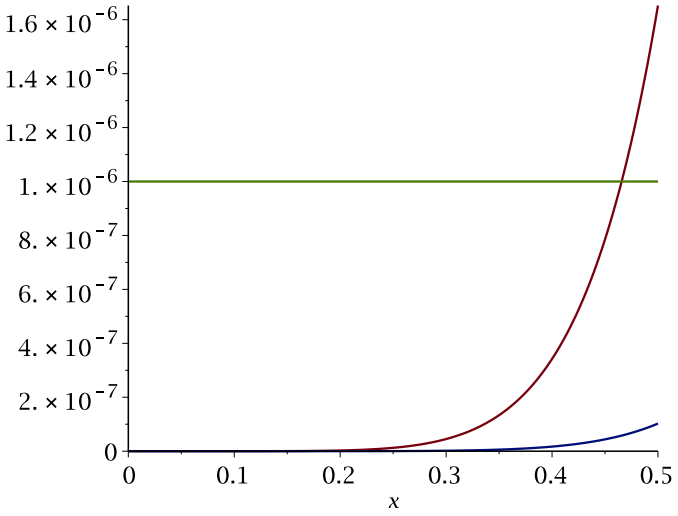
$$R6 := e^x - 1 - x - \frac{1}{2}x^2 - \frac{1}{6}x^3 - \frac{1}{24}x^4 - \frac{1}{120}x^5 - \frac{1}{720}x^6$$

$$R7 := e^x - 1 - x - \frac{1}{2}x^2 - \frac{1}{6}x^3 - \frac{1}{24}x^4 - \frac{1}{120}x^5 - \frac{1}{720}x^6 - \frac{1}{5040}x^7 \tag{26}$$

```
> printf("%e\n",subs(x=0.5,R6));
  printf("%e\n",subs(x=0.5,R7));
-1.653000e-06
```

-1.030000e-07

```
> plot([R6,R7,10^(-6)],x=0..0.5);
```




```
> # HW1 Section 1.1 Problem 26
> restart;
  myprod:=n->product((2*k+1),k=0..n);
```

$$myprod := n \to \prod_{k=0}^{n} (2k+1) \tag{27}$$

```
> S2:=sum(2^k*x^(2*k+1)/myprod(k),k=0..4);
```

$$S2 := x + \frac{2}{3}x^3 + \frac{4}{15}x^5 + \frac{8}{105}x^7 + \frac{16}{945}x^9 \tag{28}$$

```
> S1:=convert(series(exp(-x^2),x,10),polynom);
```

$$S1 := 1 - x^2 + \frac{1}{2}x^4 - \frac{1}{6}x^6 + \frac{1}{24}x^8 \tag{29}$$

```
> S1*S2;
```

$$\left(1 - x^2 + \frac{1}{2}x^4 - \frac{1}{6}x^6 + \frac{1}{24}x^8\right)\left(x + \frac{2}{3}x^3 + \frac{4}{15}x^5 + \frac{8}{105}x^7 + \frac{16}{945}x^9\right) \tag{30}$$

```
> series(S1*S2,x,10);
```

$$x - \frac{1}{3}x^3 + \frac{1}{10}x^5 - \frac{1}{42}x^7 + \frac{1}{216}x^9 + O(x^{11}) \tag{31}$$

```
> S0:=sum((-1)^k*x^(2*k+1)/(2*k+1)/k!,k=0..4);
```

$$S0 := x - \frac{1}{3}x^3 + \frac{1}{10}x^5 - \frac{1}{42}x^7 + \frac{1}{216}x^9 \quad (32)$$

```
> # HW1 Section 1.1 Problem 26c
> Digits:=15;
  # Set significant digits to be comparable with double precision
```

$$Digits := 15 \quad (33)$$

```
> An:=(n,x)->2/sqrt(Pi)*sum((-1)^k*x^(2*k+1)/(2*k+1)/k!,k=0..n);
```

$$An := (n, x) \rightarrow \frac{2\left(\sum_{k=0}^{n} \frac{(-1)^k x^{2k+1}}{(2k+1)\,k!}\right)}{\sqrt{\pi}} \quad (34)$$

```
> Bn:=(n,x)->2/sqrt(Pi)*exp(-x^2)*Sum(2^k*x^(2*k+1)/myprod(k),k=0..
  n);
```

$$Bn := (n, x) \rightarrow \frac{2\,e^{-x^2}\left(\sum_{k=0}^{n} \frac{2^k x^{2k+1}}{myprod(k)}\right)}{\sqrt{\pi}} \quad (35)$$

```
> printf("%e\n",evalf(erf(1)-An(8,1)));
  printf("%e\n",evalf(erf(1)-An(9,1)));
-1.499924e-07
1.366608e-08
> # Therefore 9 terms is enough to evaluate erf(1) to with 10^(-7)
> printf("%e\n",evalf(erf(1)-Bn(9,1)));
3.383623e-08
> # HW1 Section 1.2 Problem 10
> restart;
> p5:=sum(1/n!,n=0..5);
```

$$p5 := \frac{163}{60} \quad (36)$$

```
> Eabs:=evalf(exp(1)-p5);
```

$$Eabs := 0.001615161 \quad (37)$$

```
> Erel:=Eabs/exp(1.0);
```

$$Erel := 0.0005941845262 \quad (38)$$

```
> # parb b
  restart;
  Digits:=30;
  p10:=sum(1/n!,n=0..10);
```

$$Digits := 30$$

$$p10 := \frac{9864101}{3628800} \quad (39)$$

```
> Eabs:=evalf(exp(1)-p10);
```

$$Eabs := 2.73126607556424744202010^{-8} \quad (40)$$

```
> Erel:=Eabs/exp(1.0);
```

$$Erel := 1.0047766375690937054458273475 3\ 10^{-8} \quad (41)$$