Math 310 – Numerical Analysis
Supplementary Homework Problems
due Thursday, September 17

Pretend that a computer can only represent the floating point numbers 0, $\pm\infty$, and those which in base 2 have the form

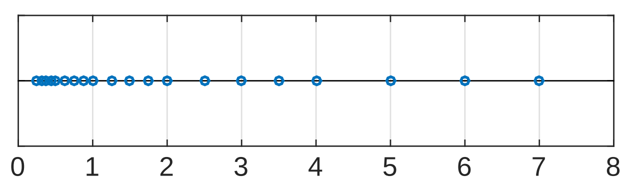
$$\pm 1.a_1 a_2 \times 2^m,$$

where $a_1,\ a_2$ are binary digits (i.e., 0 or 1) and m is an integer with $-2 \leq m \leq 2$. When a real number x is provided as input, it is converted to the machine number $fl(x)$, which is the closest number to x of the above form. When an operation such as addition is performed on inputs x and y, they are first converted to machine numbers, then added exactly, and finally the sum is converted to a machine number, so that the output is $fl(fl(x)+fl(y))$. Assume that this computer uses *rounding* to compute floating point numbers.

1. Give decimal or rational expressions for all 20 of the finite positive machine numbers. Then illustrate them all on a number line.

	2^{-2}	2^{-1}	2^0	2^1	2^2
1.00	$.01_2 = .25$	$.1_2 = .5$	$1_2 = 1$	$10_2 = 2$	$100_2 = 4$
1.01	$.0101_2 = .3125$	$.101_2 = .625$	$1.01_2 = 1.25$	$10.1_2 = 2.5$	$101_2 = 5$
1.10	$.011_2 = .375$	$.11_2 = .75$	$1.1_2 = 1.5$	$11_2 = 3$	$110_2 = 6$
1.11	$.0111_2 = .4375$	$.111_2 = .875$	$1.11_2 = 1.75$	$11.1_2 = 3.5$	$111_2 = 7$

Notice that the floating point numbers are *NOT* equally spaced. There are gaps between them.



2. Express 3.5 and 6.5 in base 2. Is either of these a machine number? Find $fl(3.5)$ and $fl(6.5)$.
$3.5 = 2 + 1 + \frac{1}{2} = 11.1_2$
$6.5 = 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 + 1 \cdot \frac{1}{2} = 110.1_2$
$fl(3.5) = 1.11 \times 2^1 = 11.1_2 = 3.5$
$fl(6.5) = 1.11 \times 2^2 = 111.0_2 = 7$

3. Express $\frac{7}{3}$ in base 2. Find $fl\left(\frac{7}{3}\right)$.
$\frac{7}{3} = 2\frac{1}{3} = 10.01010101\ldots\overline{01}_2$. To see this, use that the geometric series $\displaystyle\sum_{k=1}^{\infty}\left(\frac{1}{4}\right)^k = \frac{1}{3}$.
$fl(\frac{7}{3}) = 1.01 \times 2^1 = 10.1_2 = 2.5$

4. Give examples of machine numbers x and y (other than 0 or $\pm\infty$) such that:
   (a) $fl(x+y) = x$
   Eg. $fl(6 + .25) = 6$ since $110.0_2 + .01_2 = 110.01_2$ and $fl(110.01_2) = 1.10 \times 2^2 = 6$.
   (b) $fl(x \cdot y)$ produces an overflow
   $3 \cdot 3 = 9$ should work, since $9 > 7$. Checking, $11_2 \cdot 11_2 = 1001_2$ and $fl(1001_2) = fl(1010_2)$ with rounding, and this would be represented by 1.01×2^3 — the exponent is too big.

(c) $fl(x+y)$ produces an overflow

For the same reason as above, $1+7=8$ should work. We have $1_2 + 111_2 = 1000_2$ which creates an overflow error. (The power of 2 needed to represent 8 would be 2^3, too big for this machine.)

(d) $fl(x/y)$ produces an underflow

Here you want to divide a smallish number x by a largest number y, so that the computer can not represent the result of the division.

Let's try $x = .25$ and $y = 2$, so that $\frac{x}{y} = .125$ or $\frac{1}{8}$. We have $\frac{1}{8} = .001_2 = 1.00 \times 2^{-3}$ and the exponent is too small.

5. Give examples of real numbers x and y such that:

(a) $fl(x+y) \neq fl(fl(x)+fl(y))$

One way to do this is to use rounding in your favor. That is, choose x and y so that at least one of $fl(x)$ and $fl(y)$ rounds down, but $fl(x+y)$ will round up.

One example is $x = 3.1$ and $y = 3.25$. Check first that

$$fl(3.1) = 3, \; fl(3.25) = 3.5 \quad \text{and} \quad fl(fl(x)+fl(y)) = fl(6.5) = 7.$$

Now check that

$$fl(x+y) = fl(6.35) = 6 \neq 7.$$

(b) $fl(x \cdot y) \neq fl(fl(x) \cdot fl(y))$

For the second example, again you need to use rounding to your advantage. One possibility is $x = 1.375$, $y = 1.375$, and $x \cdot y = (1.375)^2 \approx 1.8906$. The gist of the idea is that the floating point equivalent of 1.375 will be rounded up to 1.5 and the floating point equivalent of $(1.5)^2$ will be greater than the floating point equivalent of 1.8906. The details:

$$\begin{aligned} fl(1.375) = fl(1.5) = 1.5 = 1.10 \times 10^1, \\ fl(x \cdot y) = fl(1.8906) = 2, \\ fl(fl(x) * fl(y)) = fl(1.5^2) = fl(2.25) = 2.25. \end{aligned}$$

For the next problems, suppose that (a different) computer can only represent the floating point numbers 0, $\pm\infty$, and those which in base 2 have the form

$$\pm 1.a_1 a_2 \times 2^m,$$

where a_1, a_2 are binary digits (i.e., 0 or 1) and m is an integer with $-4 \leq m \leq 4$. Moreover, this computer uses *truncation* rather than rounding. To be clear, this computer only differs from the first in that it can store a larger range of exponents and it truncates numbers with more than three binary significant digits.

6. What is machine epsilon ϵ_M for this computer? Recall that ϵ_M it is the largest floating point number such that $fl(1+\epsilon_M) = 1$.

Machine epsilon is $\epsilon_M = 1.11 \times 2^{-3} = \frac{1}{8} + \frac{1}{16} = \frac{1}{32} = \frac{7}{32} = 0.21875$ for this computer. To see this, note that $fl(1+\epsilon_M) = fl(1_2 + .00111_2) = fl(1.00111_2) = fl(1.00_2) = 1$.

Moreover, the next largest floating point number is $1.00 \times 2^{-2} = .25$, and $fl(1+.25) = fl(1_2 + .01_2) = fl(1.01_2) = fl(1.01 \times 2^0) = 1.25$

7. Find $fl(2+\epsilon_M)$ and $fl\left(\frac{1}{2}+\epsilon_M\right)$.

$fl(2+\epsilon_M) = fl(10_2 + .00111_2) = fl(10.00111_2) = fl(10.0_2) = 1.00 \times 2^1 = 2$.

$fl(\frac{1}{2}+\epsilon_M) = fl(.1_2 + .00111_2) = fl(.10111_2) = fl(.101_2) = 1.01 \times 2^{-1} = \frac{5}{8} = .625$. However, $\frac{1}{2}+\epsilon_M = \frac{1}{2} + \frac{7}{32} = \frac{23}{32} \neq fl(\frac{1}{2}+\epsilon_M)$.