# CS 357 - 05 Rounding

Boyang Li (boyangl3)

**Rounding in IEEE-754:** Not all real numbers $x$ can be expressed in the floating point format. We need to round it into the nearby machine number, either $x_-$ or $x_+$:

$$x_- = 1.b_1b_2b_3...b_n \times 2^m$$
$$x_+ = 1.b_1b_2b_3...b_n \times 2^m + 0.\underbrace{000000...0001}_{n \text{ bits}} \times 2^m$$

This process is called rounding, the error is called roundoff error. Listed below are the rounding options:

- round towards zero/infinity or up/down (different for positive and negative)
- round to nearest floating point (up/down take the closer one)
- round by chopping (take $x_-$)

| | **positive $x$** | **negative $x$** |
|---|---|---|
| Round up (ceil) | $x_+$ (towards $+\infty$) | $x_-$ (towards 0) |
| Round down (floor) | $x_-$ (towards 0) | $x_+$ (towards $-\infty$) |

The roundoff errors are bounded, as shown below:

$$|x_+ - x_-| = \epsilon_m \times 2^m$$

- **Bound for absolute error:** $|fl(x) - x| \leq \epsilon_m \times 2^m$
- **Bound for relative error:** $\dfrac{|fl(x) - x|}{|x|} \leq \epsilon_m$

**Mathematical properties:**

- **Not necessarily associative:** Because $fl(fl(x + y) + z) \neq fl(x + fl(y + z))$.
- **Not necessarily distributive:** Becuse $fl(z \cdot fl(x + y)) \neq fl(fl(z \cdot x) + fl(z \cdot y))$.
- **Not necessarily cumulative:** repeatedly adding a very small number to a large number may do nothing.

**Floating point addition:** Here the steps to do the addition:

1. Make both numbers into a common exponent
2. Do grade-school addition from left to right, until run out of digits
3. Round the result

Note: There is no loss of significant digits with floating point addition.

**Floating point subtraction and catastrophic cancellation:** Floating point subtraction is similar to addition, however problems occur when you subtract two numbers of similar magnitude. Example from book:

$$\begin{aligned} a &= 1.1011???? \times 2^1 \\ b &= 1.1010???? \times 2^1 \\ a - b &= 0.0001???? \times 2^1 \end{aligned}$$

When we normalize the result, we get $1.???? \times 2^{-3}$. There is no data to indicate what the missing digits should be. Although the floating point number will be stored with 4 digits in the fractional, it will only be accurate to a single significant digit. This loss of significant digits is known as catastrophic cancellation. A method of avoiding loss of significant digits is to eliminate subtraction.