Floating Point Arithmetic

Real numbers are called "float" values in computing. Floating point-arithmetic is the method computers use to handle real numbers.

Floats are represented using the IEEE 754 standard, which improves accuracy based by storing numbers using the principles of normalized scientific notation.

IEEE represents floating-point numbers using 3 components: a sign bit, an exponent, and a mantissa. To store numbers effectively, IEEE 754 enforces normalized form, meaning the first digit before the decimal is always 1 in binary ($1.xxxxx \cdot 2^n$)

Since this leading 1 is always present, IEEE 754 does not store it explicitly (this is called the hidden bit). Instead, only the fractional part is stored, allowing the mantissa to hold more precision within the available bits.

Exponential Notation

- The following are equivalent representations of 1,234

123,400.0	x 10^{-2}
12,340.0	x 10^{-1}
1,234.0	x 10^{0}
123.4	x 10^{1}
12.34	x 10^{2}
1.234	x 10^{3}
0.1234	x 10^{4}

The representations differ in that the decimal place – the "point" -- "floats" to the left or right (with the appropriate adjustment in the exponent).

IEEE 754 Standard

- Single precision: 32 bits, consisting of...
  - Sign bit (1 bit)
  - Exponent (8 bits)
  - Mantissa (23 bits)

Normalized binary significand with *hidden* bit (1): 1.M

Floating Point: Normalized Scientific Notation

- The mantissa must be normalized: 1.xxxxxx * 2^{yyyyy}
- Always has a 1 in front of the binary point
- This 1 does not need to be stored

- Floating point numbers have an implied "1" on left of the decimal place
  - Mantissa → 10100000000000000000000
  - Represents → $1.101_2 = 1.625_{10}$

This differs from standard scientific notation where the leading digit is always explicitly written (e.g. 1.23×10^3 in decimal). By enforcing normalization and using a hidden bit, IEEE ensures efficient storage and greater accuracy in floating point percision.

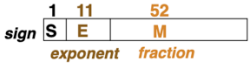
A floating-point number is always composed of:

- Sign bit (S): determines wether the number is positive (0) or negative (1).
- Exponent (E): determines how much to shift the binary point (stored using a biased format)
- Mantissa (M): stores the significant digits of the number (with an assumed 1)

IEEE 754 Standard

- Single precision: 32 bits, consisting of...
  - Sign bit (1 bit)
  - Exponent (8 bits)
  - Mantissa (23 bits)

- Double precision: 64 bits, consisting of...
  - Sign bit (1 bit)
  - Exponent (11 bits)
  - Mantissa (52 bits)



Normalized binary significand with *hidden* bit (1): 1.M

$(E-bias)$