Storing numbers in the computer

("real numbers")

Basic idea: scientific notation.

Note: Computers store numbers in binary form, and the only numbers that can be stored exactly are rational numbers of the form $\frac{p}{2^N}$, where $p \in \mathbb{Z}$.
Otherwise - we just get approximations to other real numbers by numbers of the form $\frac{p}{2^N}$.

Floating Point Numbers:

ingredients: b = base of the number system
(b=10 for decimals, b=2 for binary #s)

f = fraction / significand / mantissa

e = exponent

s = sign

t = radix point placement
(eg decimal point)

B = bias.

Examples -4537.28 Decimal $b=10$ base

$= -4.5372800 \times 10^{3}$

$s = $ sign $= 1$

sign $= (-1)^1$

$e - B = 3$ exponent.

$f = 45372800$

Sp. $B = 7$

$\Rightarrow e = 10$

$t = $ # of places $= 7$

To store in the computer $\longrightarrow$ put $f\,e\,s$ into one number. (for a particular system, b, B, t are fixed ahead of time.

$$fes = \underset{f}{45372800}\,|\,\underset{e}{0010}\,|\,\underset{s}{1}$$

$\uparrow$ negative

0 would mean positive

$\Rightarrow$

The corresponding number that this means is $x = -4.5372800 \times 10^{10-7}$

In general $\underline{x = (-1)^s f \cdot b^{-t}\, b^{e-B}}$

In reality, we use binary #s in the computer, & numbers are displayed as decimals using a conversion program.

How does this work with binary #s?

For reference: IEEE754 Standard

precision type	f f_s	e	s	bias
half-precision (16 bit)	10	5	1	15
single precision (32 bit)	23	8	1	127
double precision (64 bit)	52	11	1	1023
quadruple precision (128 bit)	112	15	1	16383

16 bit example f e s

1011101001 | 01110 | 1

$B = 15$

$e = 1110_2$ $s = 1$

TRICK for binary — for most #s, assume the first digit is 1, and don't store it.

$f = 11011101001$

assumed first #. $t = 10$ for most #s.

$\Rightarrow$

$$X = -1.1011101001 \times \underbrace{2^{14-15}}_{2^{-1}} = \boxed{-0.11011101001_2}$$

When $e = 00000$, then we take $f =$ (exactly the 10 digits) $\Big\}$ $t = 9$

No extra 1 in the front.

That way, $0 = 0000000000|00000|0$.

$X = 0\,1\;1\,1\,0\,0\,0\,0\,0\,0\,|\,0\,0\,0\,0\,0\,0\,|\,0$

$= 0.1111 \times 2^{0-15} = 0.1111 \times 2^{-15}$

$= 1.11 \times 2^{-16}.$

Another special exponent:

when $e = 11111$, $f = \cancel{1111111111}$ 0000000000, $s = 0$

reserved for $+\infty$

$e = 11111$, $f = \cancel{1111111111}$ 0000000000, $s = 1$

$-\infty$.

$e = 11111$, $f =$ anything else

$\rightarrow$ NaN "Not a number"
