Computer representation of numbers (Overton, Sec 3,4)

Binary & decimal representation:

$$(71)_{10} = 7\times 10^1 + 1\times 1$$

$$(1000111)_2 = 1\times 1 + 1\times 2 + 1\times 4 + 0\times 8 + 0\times 16 + 0\times 32 + 1\times 64$$

ith position corresponds to 2^{i-1}

$$(5.5)_{10} = 5\times 1 + 5\times \tfrac{1}{10}$$

$$(101.1)_2 = 1\times 4 + 0\times 2 + 1\times 1 + 1\times \tfrac{1}{2}$$

Some numbers have a finite representation with respect to one basis, but not another one:

$$\frac{1}{10} = \underbrace{(0.1)_{10}}_{\text{finite}} = \underbrace{(0.000110011\,00\ldots)_2}_{\text{infinite}}$$

Fixed point representation (base 2, binary)

Storage 32 bit $= 4$ byte single precision

64 bit $\to$ 8 byte double precision "default"

For example:

1 bit	15 bits	16 bits
±	0....0101	10...

$\simeq \frac{11}{2}$

sign bit

decimal

disadvantage: some numbers cannot be represented at all if they are too small or large.

Floating point representation

base 10: $x = \pm S \times 10^{E}$ ← integer, "exponent"

mantissa (S)

$1 \le S < 10$

2.3456×10^{9}

base 2: $x = \pm S \times 2^{E}$ $\quad 1 \le S < 2 \; (\Rightarrow S = 1.\ldots)$

Table 4.1: IEEE Single Format

$\pm$	$a_1 a_2 a_3 \ldots a_8$	$b_1 b_2 b_3 \ldots b_{23}$

If exponent bitstring $a_1 \ldots a_8$ is	Then numerical value represented is
$(00000000)_2 = (0)_{10}$	$\pm(0.b_1b_2b_3\ldots b_{23})_2 \times 2^{-126}$
$(00000001)_2 = (1)_{10}$	$\pm(1.b_1b_2b_3\ldots b_{23})_2 \times 2^{-126}$
$(00000010)_2 = (2)_{10}$	$\pm(1.b_1b_2b_3\ldots b_{23})_2 \times 2^{-125}$
$(00000011)_2 = (3)_{10}$	$\pm(1.b_1b_2b_3\ldots b_{23})_2 \times 2^{-124}$
$\downarrow$	$\downarrow$
$(01111111)_2 = (127)_{10}$	$\pm(1.b_1b_2b_3\ldots b_{23})_2 \times 2^{0}$
$(10000000)_2 = (128)_{10}$	$\pm(1.b_1b_2b_3\ldots b_{23})_2 \times 2^{1}$
$\downarrow$	$\downarrow$
$(11111100)_2 = (252)_{10}$	$\pm(1.b_1b_2b_3\ldots b_{23})_2 \times 2^{125}$
$(11111101)_2 = (253)_{10}$	$\pm(1.b_1b_2b_3\ldots b_{23})_2 \times 2^{126}$
$(11111110)_2 = (254)_{10}$	$\pm(1.b_1b_2b_3\ldots b_{23})_2 \times 2^{127}$
$(11111111)_2 = (255)_{10}$	$\pm\infty$ if $b_1 = \cdots = b_{23} = 0$, NaN otherwise

	$a_1 a_2 \ldots a_8$	
sign bit	8 bits for E	23 bits for mantissa

$S = (1.b_1 b_2 \ldots)$

IEEE standardized representation in '70–80s: don't store 1, "hidden bit"

round-off, dealing with Inf, −Inf, NaN

Single format

±	$a_1 \; \ldots \; a_8$	$b_1 \; \ldots \; b_{23}$

biased representation
stored is E+127

$$\left(\sum_{i=1}^{23} b_i 2^{-i} + 1 \right) \times 2^{\left[\sum_{i=1}^{\infty} a_i 2^{8-i} - 127 \right]}$$

smallest number:

±	0 ... 01	0 0

$$\simeq 1 \times 2^{-126} \approx 1.2 \times 10^{-38}$$ realmin (single)

large number:

±	1 10	1 1

$$\simeq 10^{38}$$ realmax (single)

special: $a_1 \ldots a_8 = 1$

±	1 ... 1	0 0	→ $\pm\infty$
±	1 ... 1	0 .. 1 0	→ NaN
±	0 ... 0	0 ... 0	→ 0

Table 4.1: IEEE Single Format

±	$a_1a_2a_3 \ldots a_8$	$b_1b_2b_3 \ldots b_{23}$

If exponent bitstring $a_1 \ldots a_8$ is	Then numerical value represented is
$(00000000)_2 = (0)_{10}$	$\pm(0.b_1b_2b_3 \ldots b_{23})_2 \times 2^{-126}$
$(00000001)_2 = (1)_{10}$	$\pm(1.b_1b_2b_3 \ldots b_{23})_2 \times 2^{-126}$
$(00000010)_2 = (2)_{10}$	$\pm(1.b_1b_2b_3 \ldots b_{23})_2 \times 2^{-125}$
$(00000011)_2 = (3)_{10}$	$\pm(1.b_1b_2b_3 \ldots b_{23})_2 \times 2^{-124}$
↓	↓
$(01111111)_2 = (127)_{10}$	$\pm(1.b_1b_2b_3 \ldots b_{23})_2 \times 2^{0}$
$(10000000)_2 = (128)_{10}$	$\pm(1.b_1b_2b_3 \ldots b_{23})_2 \times 2^{1}$
↓	↓
$(11111100)_2 = (252)_{10}$	$\pm(1.b_1b_2b_3 \ldots b_{23})_2 \times 2^{125}$
$(11111101)_2 = (253)_{10}$	$\pm(1.b_1b_2b_3 \ldots b_{23})_2 \times 2^{126}$
$(11111110)_2 = (254)_{10}$	$\pm(1.b_1b_2b_3 \ldots b_{23})_2 \times 2^{127}$
$(11111111)_2 = (255)_{10}$	$\pm\infty$ if $b_1 = \cdots = b_{23} = 0$, NaN otherwise

Subnormals:

$$\boxed{\pm \,|\, 0 \; \ldots \; 0 \,|\, 1 \; 0 \; \ldots \; \ldots \; 0}$$

$$\simeq (0.1)_2 \times 2^{-126} \simeq 2^{-127}$$

$$\boxed{\pm \,|\, 0 \; \ldots \; 0 \,|\, 0 \; 0 \; \ldots \; \ldots \; 01}$$

$$= 2^{-149}$$

Double precision: 64 bits

$$\boxed{\pm \,|\, a_1 \; \ldots \; a_{11} \,|\, b_1 \; \ldots \; b_{52}}$$

11 bits 52 bits

Machine epsilon $(eps(1))$: gap between 1 and next largest number

$$\boxed{\pm \,|\, 0 \; 1 \; \ldots \; 1 \,|\, 0 \; \ldots \; \ldots \; 0}$$

$$1.0 \ldots 0 \times 2^0$$

$$\longrightarrow eps = \underline{2^{-52} \approx 10^{-16}} \quad \text{(double)}$$

$$\approx 10^{-7} \quad \text{(single)}$$