1.3

Computer (floating point arithmetic)

Fixed point decimal representation of real numbers

Ex: $-428.391 = -\left[4\times10^{2}+2\times10+8\times1+3\times10^{-1}+9\times10^{-2}+1\times10^{-3}\right]$

$$x = -.428391 \times 10^{3}$$

(sign: $-$; decimal pt.: $.$; fraction: 428391; base: 10; exponent: 3)

Normalized floating pt. ⇒ written in a standard way

Base 2 Analogue

Any nonzero real number has the representation

$x = \pm q \times 2^{m}$, where $\frac{1}{2} \leq q < 1$.

q has a possibly infinite binary representation

$$q = .1\,d_1 d_2 d_3 \cdots$$

(binary pt.: $.$)

Example

$\pi = +3.14159 \Rightarrow$ base 10

$\pi = 11.0010 \Rightarrow$ base 2 $= .110010\cdots \times 2^{2}$

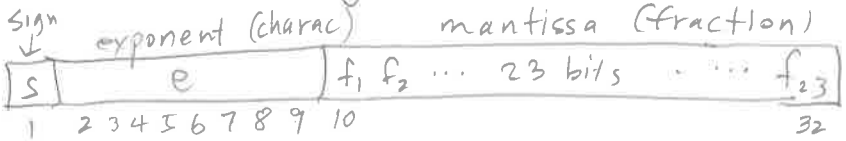
(binary floating point approx) representation of real number) Truncate (or round off) the fraction to a fixed # of digits This determines the relative accuracy.

Note: exponent is also stored in binary form

The number of bits available determines the smallest and largest possible positive numbers that can be represented.

Storage of numbers

Ex: IEEE Single Precision (32 bits)



This represents

$$x = (-1)^s \times 1.f_1 f_2 \cdots f_{23} \times 2^{e-b}$$

Note $(-1)^s = \begin{cases} 1, & s=0 \\ -1, & s=1 \end{cases}$

b = bias for the exponent

for single precision, $b = 2^7 - 1 = 127$

purpose for b is no sign is needed.

Smallest value $\Rightarrow$ fraction $= 0$

$\Rightarrow b = 127$

$\Rightarrow e = 0$

$\Rightarrow 2^{-127}$

$$0 - 127 \le e - b \le 255 - 127$$

$$-127 < e - b < 128$$

A rough est. of the smallest positive and largest pos.

is $2^{-127} = 5.9 \times 10^{-39}$ and $1.\underbrace{111}_{23 \text{ ones}} \times 2^{128} \approx 2 \times 2^{128} = 2^{129}$

$= 6.8 \times 10^{38}$

Other considerations

$e=0, f=0 \Rightarrow$ represents 0

$e=255, f=0 \Rightarrow$ represents $(-1)^s \times \infty = \pm\infty$

$e=255, f\neq 0 \Rightarrow$ represents NaN (not a number)

} Mostly for debugging purposes.

Note: IEEE Double Precision floating pt uses 64 bits.

Ques: How many bits are used for exponent (charac) fraction (mantissa)?

We can find in a computer by obing $1 + eps$, where ε_M is the machine epsilon. ε_M is the smallest pos number such that $fl(1 + x) > 1$.

See handout

DEF: Absolute error

If p^* is an approx to p, then the abs. error is

$$|p - p^*|$$

The relative error is

$$\frac{|p - p^*|}{|p|}, \text{ provided } p \neq 0.$$

---

for floating pt numbers

$$\frac{|fl(x) - x|}{|x|} \leq \varepsilon_M \quad \text{(the bound is attained sometimes)}$$

If $x \neq 0$, $fl(x) = \underset{\substack{\uparrow \\ \text{true} \\ \text{value}}}{x}(1 + r)$, where $|r| \leq \varepsilon_M$

Note relative accuracy of operations $+, -, \times$, or $\div$ are just as accurate, if $x \odot y \neq 0$ and "extended precision" is used for operation, but not storage. (The only error results from storage)

$$\frac{|fl(x \odot y) - x \odot y|}{|x \odot y|} \leq \varepsilon_M$$

problems with computers

Try

```
x=1/3
xvec=rep(0,n)
for (i=1:30) {
  x=(9x+1)x-1
  xvec[i]=x
}
```

$9x^2+x-1 = 9\left(\frac{1}{3}\right)^2+\frac{1}{3}-1$

$= 1+\frac{1}{3}-1 = \frac{1}{3}$

xvec => It's supposed to replace x by itself each time

Look how large the error can be (compound)

Try $2+\underbrace{.2+.2+.2+.2+.2}_{1}-3$ => the answer isn't zero on a computer.

$.2+.2+.2+.2+.2+2-3$ (But $2+.2(5)-3$ does)

Try 1.0000001 (repeatedly square 27 times) Try $1.0000001^{2^{24}}$

real answer $= 674530.4707$

Do floor(100*(4.34-floor(4.34)))