Computational Numerical Methods

CS 374

Prosenjit Kundu

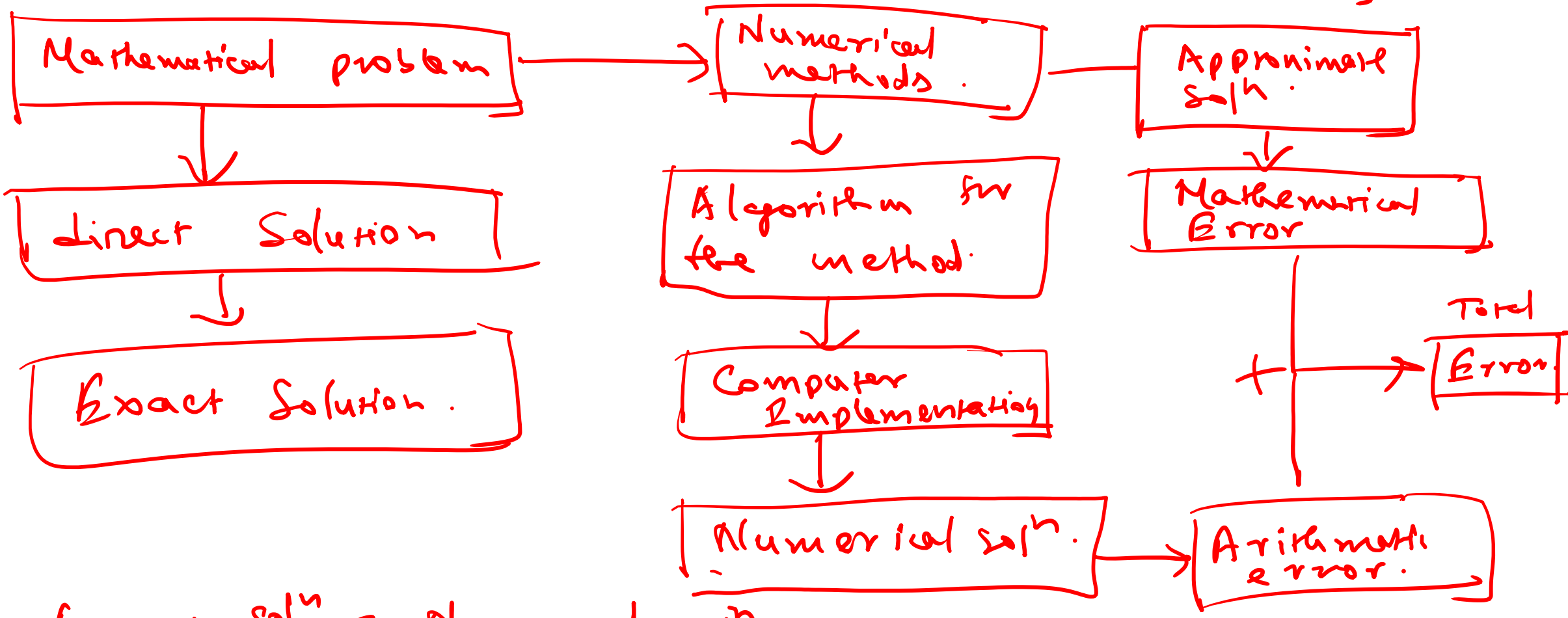
Computer Arithmatic & Errors.

1. ~~floating point representation of numbers~~

$\frac{1}{3} =$

0.33333

0.333333



Exact soln = Numerical soln + error.

Floating point representation of numbers

Let ~~$B=2$~~ Let $B \in \mathbb{N}$ & $B \geq 2$

Any real number can be represented exactly in in

in base B as

$$(-1)^S \times (.d_1 d_2 \cdots d_n d_{n+1} \cdots)_B \times B^e .$$

with $d_i \in \{0, 1, \cdots B-1\}$ $d_1 \neq 0$. or $d_1 = d_2 \cdots d_n = \cdots 0$

$S = 0$ or 1 in called the sign.

an appropriate integer e is called exponent.

- $d_1 d_2 \cdots d_n d_{n+1} \cdots$ called mantissa.

$$= \frac{d_1}{B} + \frac{d_2}{B^2} + \cdots + \frac{d_n}{B^n} + \cdots .$$

$$0.3145 = (-1)^0 (.3145)_{10} (10)^0$$

$$-0.003145 = (-1)^1 (.3145)_{10} 10^{-2}$$

$$0.3145 = \frac{3}{10} + \frac{1}{100} + \frac{4}{1000} + \frac{5}{10000}$$

$$= \frac{3}{10} + \frac{1}{10^2} + \frac{4}{10^3} + \frac{5}{10^4}.$$

If $\beta = 2$.

$$(-1)^{s} \times (.d_1 d_2 \cdots d_n d_{n+1} \cdots)_2 \times 2^e.$$

is called the binary floating point representation.

For $\beta = 10$

is called decimal floating point representation.

Ex $x = -0.0014$

$= (-1)^{s} \; 0.14 \times 10^{-2}$.

Note There are only finite number of digit that can be in the n digit f.p. representation.

But

$$\frac{1}{3} = 0.3333\ldots$$

$$= (-1)^0 (0.333\ldots) \cdot (10)^0$$

Underflow & Overflow of memory

When the value of the exponent e in floating point representation exceeds the maximum limit of (M) the memory we encounter the overflow of memory.

When the value undergoes below of the minimum (m) range we will encounter underflow.

$$m \leq e \leq M .$$

- If $e > M$ then the number is considered as ∞
  if $e < m$ " " " " " 0

Chopped & Rounded numbers

Let x be a real number given in the FP representation as

$$x = (-1)^s \times (.d_1 d_2 \ldots d_n d_{n+1} \ldots)_\beta \; \beta^e.$$

The chopped approximation of x is:

$$fl(x) = (-1)^s (.d_1 d_2 \ldots d_n)_\beta \; \beta^e.$$

Let x be a real number given in the FP representation

as

$$x = (-1)^s \left(.d_1 d_2 \cdots d_n d_{n+1} \cdots\right)_\beta \beta^e.$$

The rounded approximation of x is given by

$$f(x) = \begin{cases} (-1)^s \left(.d_1 d_2 \cdots d_n\right)_\beta \times \beta^e \\ (-1)^s \left(.d_1 d_2 \cdots (d_n+1)\right)_\beta \times \beta^e \end{cases}$$

$$0 \leq d_{n+1} < \beta/2$$

$$\frac{\beta}{2} \leq d_{n+1} < \beta$$

Performing arithmetic operations

We * denote any of the basic arithmetic operations.

$* = \{+, -, \times, \div\}$

We x & y be two real numbers.

The process of computing $x * y$ is as follows.

Step 1 Enter the n-digit fp approximation

$fl(x)$ & $fl(y)$ of the numbers x, & y respectively

Step 2 Perform the calculations $fl(x) * fl(y)$ using exact arithmetic.

Step 3 : Get the n digit FP representation of $fl(fl(x) * fl(y))$

Example : Consider $f(x) = x\left(\sqrt{x+1} - \sqrt{x}\right)$

Evaluate $f(100000)$ using a 5 in digit rounding.

$fl(100000) = 100000$

$fl(100001)$

$f(100000) = 100000\left(\sqrt{100001} - \sqrt{100000}\right)$

$\sqrt{100001} \approx 316.229347 = (-1)^0 (0.31622\,9347)$

$= (-1)^0\ 0.316229 \times 10^3$

$fl(\sqrt{100000}) = (-1)^0 \times 0.316228 \cdot \times 10^3 \cdot$

$\therefore$ ~~$fl(fl(x)*1) = fl(x)$~~

$fl(fl\sqrt{100001} - fl((100000)) = 0.1 \times 10^{-2}.$

Finally $fl((f(100000)) = fl((100000) \times (0.1 \times 10^{-2})$

$= 100$