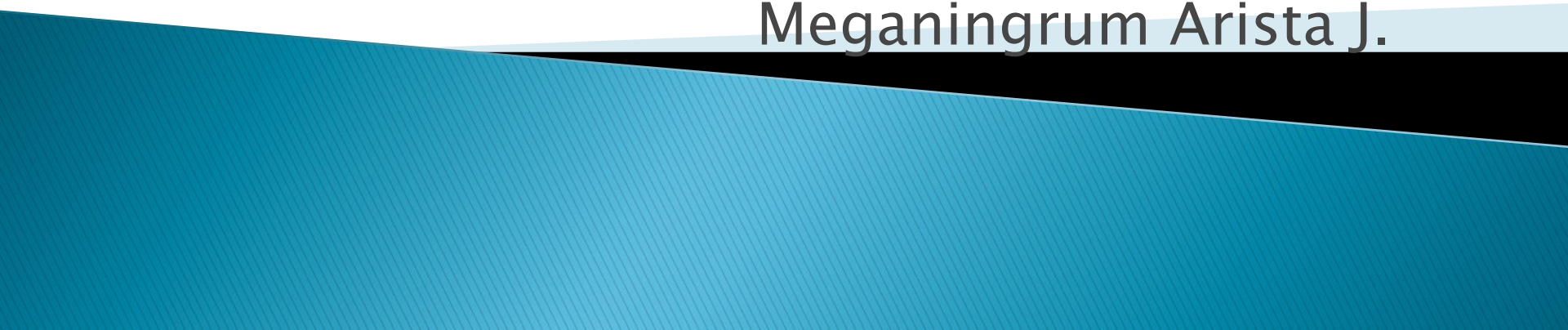


Scientific Computing (a.k.a Numerical Analysis)

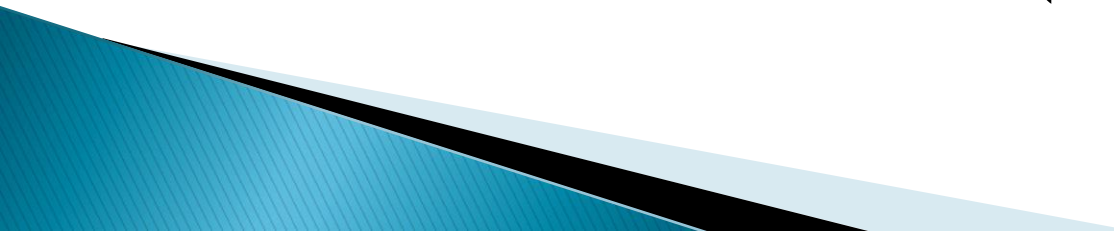
Feb – June 2015

Lecturer:

T. Basaruddin, Heru Suhartanto,
Meganingrum Arista J.



Course Outline

- Scientific Computing & Computer Arithmetic
 - System of linear equations
 - Least Square Problems
 - Nonlinear equations
 - Optimization
 - Interpolation
 - Numerical Differentiation and Integration
 - Initial Value Problems (IVP)
- 

Scientific Computing

- ▶ Aims at solving real world problems
- ▶ By solving related mathematical models
- ▶ Using computer (esp. numerically)
- ▶ Major issues:
 - Accuracy
 - Efficiency (speed, space)
- ▶ Source of error:
 - Data error
 - Computational error

Classification of errors

▶ Computational errors

- Truncation error: approximation using truncated formula
- Rounding error: limitation on machine capacity to store numbers
- The two can occur simultaneously: e.g. Taylor Approximation

▶ Two ways of measuring error

- Absolute error
- Relative error

▶ Two basic approaches in error analysis

- Forward error analysis: directly measure the difference between the computed and the actual
- Backward error analysis: use proxy representation of error
 - E.g. 1.9 is used as solution to $\sqrt{4}$

Computer Arithmetic

Floating point number



$$x = sm\beta^e$$

$$L \leq e \leq U$$

$$m = \left(\sum_{j=0}^{p-1} d_j \beta^{-j} \right); \quad 0 \leq d_i \leq \beta - 1$$

β base or radix

p precision

$[L;U]$ exponent range

Normalization

- ▶ Floating-point system normalized if leading digit d_0 always nonzero unless number represented is zero
- ▶ In normalized system, mantissa m of nonzero floating-point number always satisfies

$$1 \leq m \leq \beta$$

- ▶ Reasons for normalization
 - no digits wasted on leading zeros
 - leading bit need not be stored (in binary system)

Computer Arithmetic-2

System	β	p	L	U
IEEE Single Precision	2	24	-126	127
IEEE Double Precision	2	53	-1022	1023
Cray	2	48	-16383	16384
HP Calculator	10	12	-499	499
IBM Mainframe	16	6	-64	63

Properties of Floating-Point Systems

- ▶ Floating-point number system is finite and discrete
- ▶ Number of normalized floating-point numbers:

$$2(\beta - 1) \beta^{p-1} (U - L + 1) + 1$$

- ▶ Smallest positive normalized number:
underflow level = UFL = β^L

- ▶ Largest floating-point number:
overflow level = OFL = $\beta^{U+1} (1 - \beta^{-p})$

Example: $\beta=2$, $p=3$, $L=-1$, $U=1$

- ▶ Thus there will be 25 numbers that can be represented;
- ▶ OFL = 3.5
- ▶ UFL = 0.5

Rounding Rules & Precision

- ▶ Chopping: simply ignoring the extra digit
- ▶ Rounding to nearest: use the nearest floating point number
- ▶ Accuracy of floating-point system characterized by unit round-off, machine precision, or machine epsilon, denoted by $\epsilon\text{-mach}$
- ▶ With rounding by chopping, $\epsilon\text{-mach} = \beta^{1-p}$
- ▶ With rounding to nearest, $\epsilon\text{-mach} = 0.5 \beta^{1-p}$
- ▶ Alternative definition is smallest number ϵ such that $(1 + \epsilon) > 1$
- ▶ In all practical floating-point systems,
 $0 < \text{UFL} < \epsilon\text{-mach} < \text{OFL}$
- ▶ The general representation of a floating point
$$\text{fl}(x) = x(1+\delta) \quad 0 \leq \delta \leq \epsilon$$

Stability & Conditioning

- ▶ Condition number
$$\kappa = \frac{|f(\hat{x}) - f(x)| / |f(x)|}{|\hat{x} - x| / |x|}$$
$$\approx \left| \frac{xf'(x)}{f(x)} \right|$$
- ▶ Absolute condition number
$$\kappa_{abs} = \frac{|f(\hat{x}) - f(x)|}{|\hat{x} - x|}$$
- ▶ Ill condition if K is large

Issues on FP arithmetic

- ▶ Error propagation
 - A lengthy arithmetic operation
 - E.g. $\text{fl}(x) = x(1 + \delta) \rightarrow \text{fl}(x^2) = x^2(1 + 2\delta)$
- ▶ Lost of significant digit (due to cancellation)
 - E.g. $x = 0.21232145$ (8 significant)
 $y = 0.21232236$ (8 significant)
 $y - x = 0.91 \times 10^{-6}$ (2 significant)

Sample variance

$$s_n^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2; \quad \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

$$s_n^2 = \frac{1}{n-1} \left(\sum_{i=1}^n x_i^2 - \frac{1}{n} \left(\sum_{i=1}^n x_i \right)^2 \right)$$

$$s_n^2 = Q_n / (n-1)$$

$$Q_1 = 0; Q_k = Q_{k-1} + \frac{(k-1)(x_k - M_{k-1})^2}{k}$$

$$M_1 = x_1; M_k = M_{k-1} + \frac{x_k - M_{k-1}}{k}$$