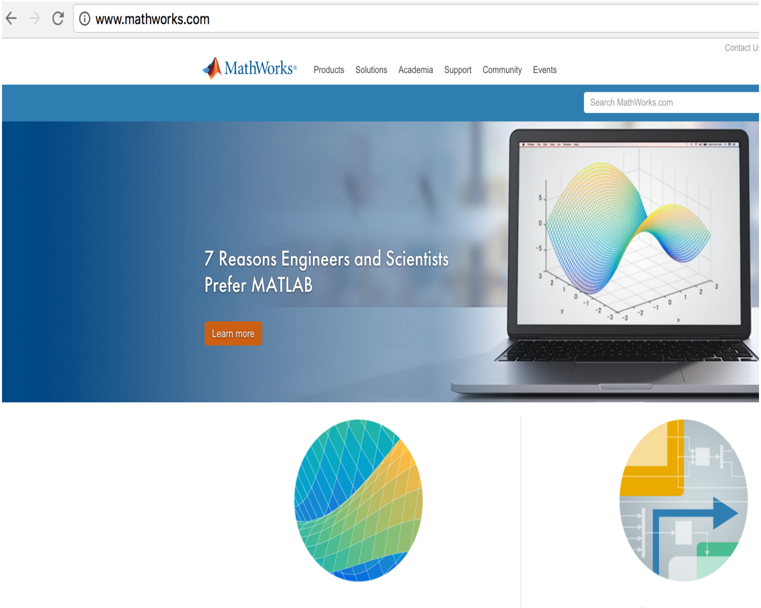
Self Introduction

- **Name:** Ming Gu
- **Office:** 861 Evans
- **Email:** mgu@berkeley.edu
- **Office Hours:** TuWTh 1:30-3:00PM
- **Class Website:**
  math.berkeley.edu/∼mgu/MA128ASpring2017

Text Book

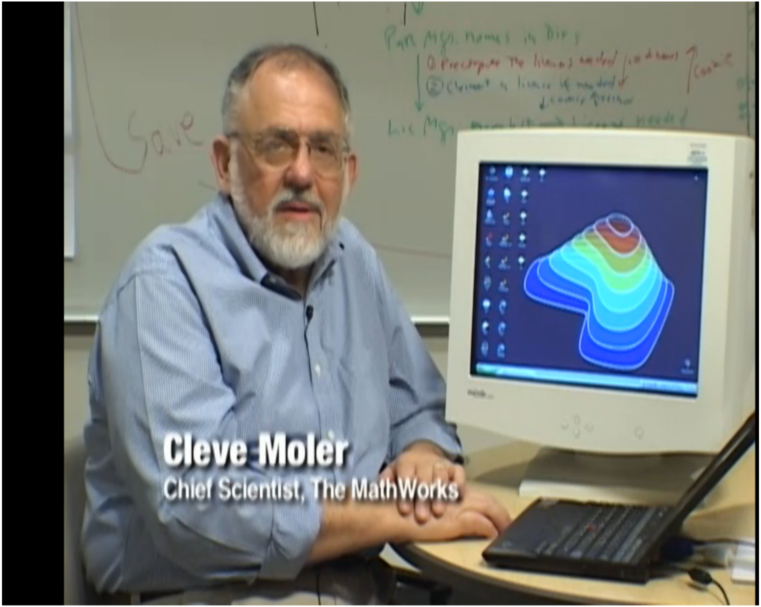
- Burden and Faires, **Numerical Analysis**. Required.

Matlab



www.mathworks.com
Contact U
MathWorks
Products
Solutions
Academia
Support
Community
Events
Search MathWorks.com
7 Reasons Engineers and Scientists Prefer MATLAB
Learn more

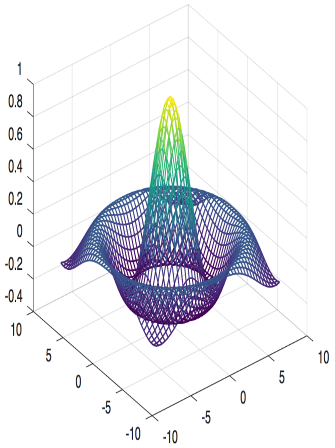
Cleve Moler



and maybe Octave

s://www.gnu.org/software/octave/

GNU Octave About Donate Download Get Involved News Support/Help Docs



GNU Octave

Scientific Programming Language

- Powerful mathematics-oriented syntax with built-in plotting and visualization tools
- Free software, runs on GNU/Linux, macOS, BSD, and Windows
- Drop-in compatible with many Matlab scripts

Download | Docs

Class Work

- Up to 14 weekly home work sets;
  Count best 10, total 10 points.
- 5 Quizzes;
  Count best 4, total 10 points.
- 2 Programming Assignments, total 20 points;
- 1 Midterm exam, 25 points;
- 1 Final exam, 35 points.
- FINAL WORTH 60 POINTS IF MIDTERM MISSING.

Quiz and Exam Schedule

- **Quiz**: Jan. 31/Feb. 1 in discussion
- **Quiz**: Feb. 14/Feb. 15 in discussion
- **Programming Assignment 1**: 11:59PM, Feb. 22
- **Quiz**: Mar. 7/Mar. 8 in class
- **Midterm**: Mar. 21 in class
- **Quiz**: Apr. 4/Apr. 5 in discussion
- **Quiz**: Apr. 18/Apr. 19 in discussion
- **Programming Assignment 2**: 11:59PM, Apr. 26
- **Final Exam**: May 12, 7:00-10:00PM (Exam Group 20)

Numerical Analysis = Calculus on a computer

- First 6 Chapters of Text Book.
- **Chapter 1**: Calculus, Computer Math.
- **Chapter 2**: Solve $f(x) = 0$.
- **Chapter 3**: Approximate given functions.
- **Chapter 4**: Derivatives, integrals.
- **Chapter 5**: Initial value ODEs.
- **Chapter 6**: Solve $Ax = b$.

Fibonacci's Problem in 1224, with Emperor Frederick II

Solve

$$f(x) = x^3 + 2x^2 + 10x - 20 = 0.$$

Fibonacci's Solution

$$x = 1 + 22\left(\frac{1}{60}\right) + 7\left(\frac{1}{60}\right)^2 + 42\left(\frac{1}{60}\right)^3 + 33\left(\frac{1}{60}\right)^4 + 4\left(\frac{1}{60}\right)^5 + 40\left(\frac{1}{60}\right)^6.$$

Fibonacci's Problem in 1224, with Emperor Frederick II

Solve

$$f(x) = x^3 + 2x^2 + 10x - 20 = 0.$$

Fibonacci's Solution

$$x = 1 + 22\left(\frac{1}{60}\right) + 7\left(\frac{1}{60}\right)^2 + 42\left(\frac{1}{60}\right)^3 + 33\left(\frac{1}{60}\right)^4 + 4\left(\frac{1}{60}\right)^5 + 40\left(\frac{1}{60}\right)^6.$$

The computer has a better solution

$$x = 1 + 22\left(\frac{1}{60}\right) + 7\left(\frac{1}{60}\right)^2 + 42\left(\frac{1}{60}\right)^3 + 33\left(\frac{1}{60}\right)^4 + 4\left(\frac{1}{60}\right)^5 + 39\left(\frac{1}{60}\right)^6.$$

Fibonacci's Cubic Root

```
>> format long e;
>> h = [1 2 10 -20];
>>  r = roots(h)

r =

     -1.684404053910685e+00 + 3.431331350197691e+00i
     -1.684404053910685e+00 - 3.431331350197691e+00i
      1.368808107821373e+00

>> Fibonacci = (((((40/60+4)/60+33)/60+42)/60+7)/60+22)/60+1

Fibonacci =

     1.368808107853224e+00

>>  r(3)-Fibonacci

ans =

    -3.185118835347112e-11

>> Better = ((((((31/60+38)/60+4)/60+33)/60+42)/60+7)/60+22)/60+1

Better =

     1.368808107821430e+00

>> r(3)-Better

ans =

    -5.795364188543317e-14
```

Roots of a Random Quintic Polynomial: No closed form formula

```
>>  format short g
>>  hrand = randn(1,6)

hrand =

     -1.3499       3.0349       0.7254    -0.063055      0.71474     -0.20497

>> rrand = roots(hrand)

rrand =

      2.4872
    -0.70735
       0.105 +    0.56831i
       0.105 -    0.56831i
      0.2584
```

Simple Numerical Integration (I)

- $$I_1 = \int_{-1}^{1} \sqrt{1+x}\, dx = 4/3\sqrt{2}.$$
- $$I_2 = \int_{-1}^{1} \sqrt{1+x^2}\, dx = ?.$$

Simple Numerical Integration (I)

```
>> format long g
>> I2 = quad(@(x) sqrt(1+x.^2), -1 ,1 )

I2 =

          2.29558701441275

>> I1 = quad(@(x) sqrt(1+x), -1 ,1 )

I1 =

          1.88561089016424

>> I1 - (4/3)*sqrt(2)

ans =

     -7.19299988971578e-06
```

Simple Numerical Integration (II)

```
>> I1tol = quad(@(x) sqrt(1+x), -1 ,1,1e-12 )

I1tol =

          1.88561808316058

>> I1tol - (4/3)*sqrt(2)

ans =

      -3.5500491435414e-12
```

Simple Numerical Integration (III)

```
>> [I1,fcnt1] = quad(@(x) sqrt(1+x), -1 ,1, 1e-4);disp([I1-(4/3)*sqrt(2),fcnt1])
  -4.5143e-04   2.1000e+01

>> [I1,fcnt1] = quad(@(x) sqrt(1+x), -1 ,1, 1e-8);disp([I1-(4/3)*sqrt(2),fcnt1])
  -4.0728e-08   1.1300e+02

>> [I1,fcnt1] = quad(@(x) sqrt(1+x), -1 ,1, 1e-12);disp([I1-(4/3)*sqrt(2),fcnt1])
  -3.5500e-12   6.9700e+02

>> [I1,fcnt1] = quad(@(x) sqrt(1+x), -1 ,1, 1e-16);disp([I1-(4/3)*sqrt(2),fcnt1])
  -4.4409e-16   4.3930e+03

>> [I1,fcnt1] = quad(@(x) sqrt(1+x), -1 ,1, 1e-20);disp([I1-(4/3)*sqrt(2),fcnt1])
Warning: Maximum function count exceeded; singularity likely.
> In quad at 106
  -5.6018e-06   1.0017e+04
```

Trajectory with ODE

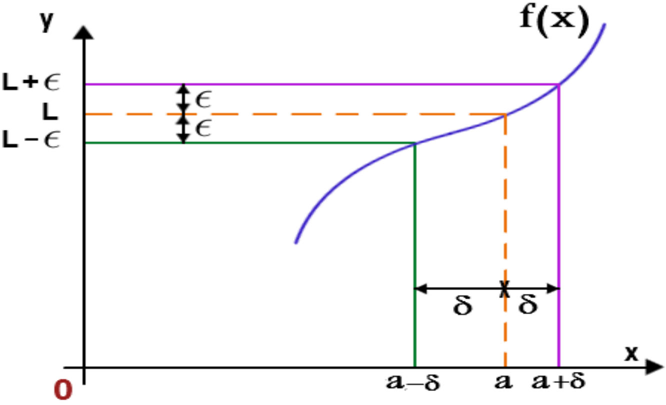
Problem to solve: find function $y(t)$ so that

$$y'(t) = f(y(t), t), \quad y(t_0) = y_0$$

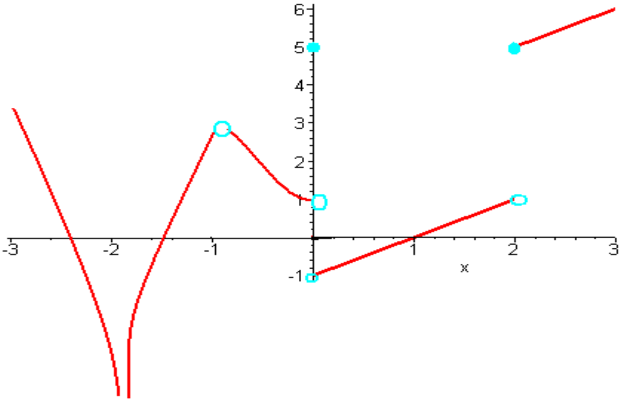
$y(t)$ could be trajectory of a flying bullet.

Shooting Method for ODEs

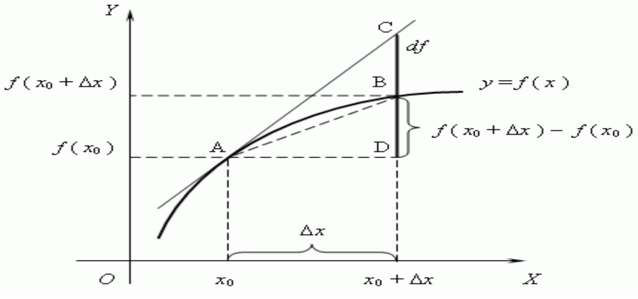
Def: Limit



Def: Continuity

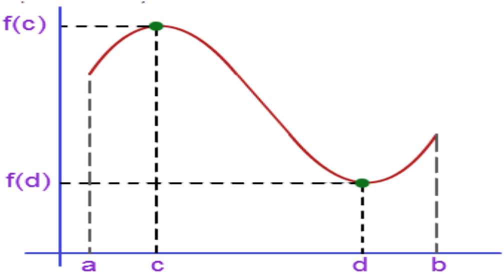


Def: Differentiability



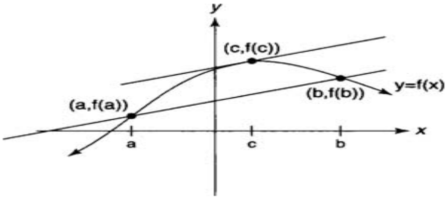
$$f'(x_0) = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}.$$

Extreme Value Theorem



- Maximum $f(c)$ and minimum $f(d)$ attainable in $[a, b]$ if $f(x)$ continuous.
- Basis of much of data analysis, artificial intelligence.

Mean Value Theorem

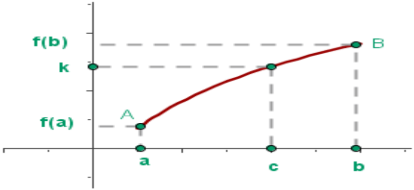


- If $f(x)$ continuous, then c exists in $[a, b]$ so

$$f'(c) = \frac{f(b) - f(a)}{b - a}.$$

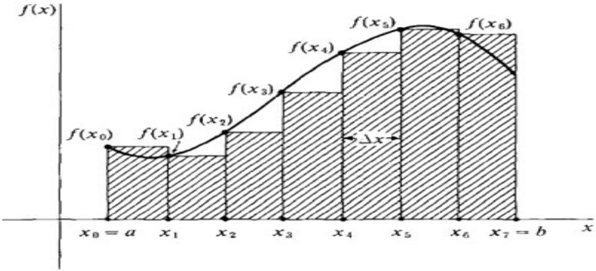
- Basis of much of theoretical analysis.

Intermediate Value Theorem



- If $f(x)$ continuous, then c exists in $[a, b]$ so $f(c) = k$ for any k between $f(a)$ and $f(b)$.
- Basis of methods for solving $fx) = 0$.

Riemann Sum



- $$\int_a^b f(x)\,dx = \lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} f(x_k).$$

Machine Precision

- Computer numbers (floating point numbers) are a **finite** subset of rational numbers.
- There is a smallest positive computer number ϵ so that

$$1 + \epsilon > 1$$

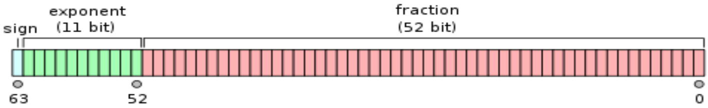
Machine Precision

```
>>  eps

ans =

     2.220446049250313e-16

>>  x = 1 + eps; disp([eps, (x-1)/eps])
     2.220446049250313e-16     1.000000000000000e+00

>>  delta = 0.75*eps

delta =

     1.665334536937735e-16

>>  x = 1 + delta; disp([delta, (x - 1)/delta])
     1.665334536937735e-16     1.333333333333333e+00

>> Delta = 0.5*eps

Delta =

     1.110223024625157e-16

>>  x=1+Delta;disp([Delta, (x - 1)/Delta])
     1.110223024625157e-16                         0
```

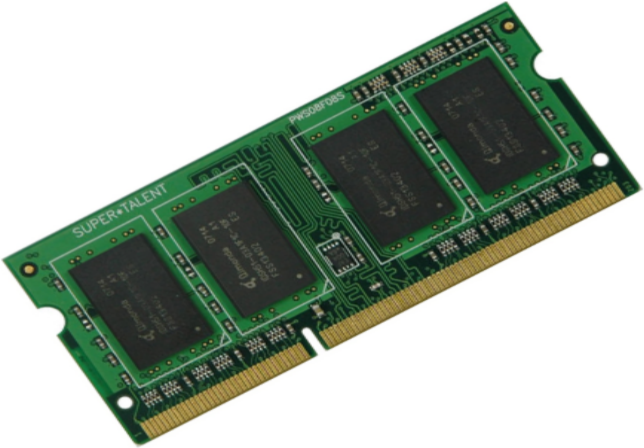
Overflow

```
>>  x=2^1023
 x=2^1023

x =

  8.9885e+307

>> x = 2* x
x = 2* x

x =

   Inf

>> y = 2^(-1023)
y = 2^(-1023)

y =

  1.1125e-308

>> y = y/(2^51)
y = y/(2^51)

y =

  4.9407e-324

>> y = y / 2
y = y / 2

y =

     0
```

IEEE 754 Double Precision Format



Apple Memory Chip



Round-off Errors and Computer Arithmetic

- **Binary Machine Numbers:** any double precision non-zero *floating point number* has form

$$x = (-1)^s \, 2^{c-1023} \, (1 + f) \,,$$

using 64 bits
  - s = sign bit: 0 for $x > 0$ and 1 for $x < 0$.
  - c = characteristic, with 11 bits:

$$c = c_1 \cdot 2^{10} + c_2 \cdot 2^9 + c_3 \cdot 2^8 + c_4 \cdot 2^7 + c_5 \cdot 2^6 + c_6 \cdot 2^5 + c_7 \cdot 2^4 + c_8 \cdot 2^3 + c_9 \cdot 2^2 + c_{10} \cdot 2^1 + c_{11} \cdot 2^0,$$

    with each $c_j = 0$ or 1.
  - f = mantissa with 52 bits

$$f = f_1 \cdot \left(\frac{1}{2}\right) + \cdots + f_{52} \cdot \left(\frac{1}{2}\right)^{52} = \sum_{j=1}^{52} f_j \cdot \left(\frac{1}{2}\right)^j,$$

    and each $f_j = 0$ or 1.

Round-off Errors and Computer Arithmetic

- **Binary Machine Numbers:** Example binary string

  0 10000000011 1011100100010000000000000000000000000000000000000000

- $s = 0,\ c = (10000000011)_2 = 1024 + 2 + 1 = 1027$, and

$$f = 1\cdot\left(\frac{1}{2}\right)^1 + 1\cdot\left(\frac{1}{2}\right)^3 + 1\cdot\left(\frac{1}{2}\right)^4 + 1\cdot\left(\frac{1}{2}\right)^5 + 1\cdot\left(\frac{1}{2}\right)^8 + 1\cdot\left(\frac{1}{2}\right)^{12}.$$

$$(-1)^s 2^{c-1023}(1+f) = (-1)^0 \cdot 2^{1027-1023}\left(1 + \left(\frac{1}{2} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{256} + \frac{1}{4096}\right)\right)$$

$$= 27.56640625.$$

Round-off Errors and Computer Arithmetic

- **Binary Machine Numbers:** any double precision non-zero *floating point number* has form

$$x = (-1)^s \, 2^{c-1023} \, (1+f) \, ,$$

using 64 bits

  - s = sign bit: 0 for $x > 0$ and 1 for $x < 0$.
  - c = characteristic, with 11 bits:

$$c = c_1 \cdot 2^{10} + c_2 \cdot 2^9 + c_3 \cdot 2^8 + c_4 \cdot 2^7 + c_5 \cdot 2^6 + c_6 \cdot 2^5 + c_7 \cdot 2^4 + c_8 \cdot 2^3 + c_9 \cdot 2^2 + c_{10} \cdot 2^1 + c_{11} \cdot 2^0,$$

with each $c_j = 0$ or 1.

  - f = mantissa with 52 bits

$$f = f_1 \cdot \left(\frac{1}{2}\right) + \cdots + f_{52} \cdot \left(\frac{1}{2}\right)^{52} = \sum_{j=1}^{52} f_j \cdot \left(\frac{1}{2}\right)^j ,$$

and each $f_j = 0$ or 1.

Round-off Errors and Computer Arithmetic

- **k-digit Decimal Machine Numbers:**

$$x = \pm 0.d_1 d_2 \cdots d_k \times 10^n, \quad \text{where} \quad 1 \le d_1 \le 9, \quad 0 \le d_i \le k, i \ge 2.$$

- Any positive real number

$$\begin{aligned} y &= 0.d_1 d_2 \cdots d_k d_{k+1} d_{k+2} \cdots \times 10^n, \\ &\approx 0.d_1 d_2 \cdots d_k \times 10^n \stackrel{def}{=} fl(y) \quad (\textbf{chopping}) \\ &\approx 0.\delta_1 \delta_2 \cdots \delta_k \times 10^n \stackrel{def}{=} fl(y) \quad (\textbf{rounding}), \end{aligned}$$

where

$$\textbf{rounding} = \textbf{chopping on} \quad y + 5 \times 10^{n-(k+1)}.$$

  - If $d_{k+1} < 5$: **rounding = chopping.**
  - If $d_{k+1} \ge 5$: cut off d_{k+1} and below, then add 1 to d_k.

Round-off Errors and Computer Arithmetic

- 5-**digit Decimal Machine Numbers for** π**:**

$$\begin{aligned} \pi &= 0.314159265\cdots \times 10^1 \\ &\approx 0.31415 \times 10^1 = 3.1415 \quad \textbf{(chopping)} \\ &\approx (0.31415 + 0.00001) \times 10^1 = 3.1416 \quad \textbf{(rounding)}. \end{aligned}$$

Absolute error vs. relative error

Suppose that p^* is an approximation to $p \neq 0$.

- **absolute error** $= |p - p^*|$,
- **relative error** $= \frac{|p-p^*|}{|p|}$.

Example

- **absolute errors:**

$$|\pi - 3.1415| \approx 9 \times 10^{-5}, \quad |\pi - 3.1416| \approx 7 \times 10^{-6}.$$

- **relative errors:**

$$\frac{|\pi - 3.1415|}{\pi} \approx 3 \times 10^{-5}, \quad \frac{|\pi - 3.1416|}{\pi} \approx 2 \times 10^{-6}.$$

Relative error for **chopping**

Suppose that $y = 0.d_1 d_2 \cdots d_k d_{k+1} d_{k+2} \cdots \times 10^n$, with $d_1 \geq 1$.

$$\left|\frac{y - fl(y)}{y}\right| = \left|\frac{0.d_1 d_2 \ldots d_k d_{k+1} \ldots \times 10^n - 0.d_1 d_2 \ldots d_k \times 10^n}{0.d_1 d_2 \ldots \times 10^n}\right|$$

$$= \left|\frac{0.d_{k+1} d_{k+2} \ldots \times 10^{n-k}}{0.d_1 d_2 \ldots \times 10^n}\right| = \left|\frac{0.d_{k+1} d_{k+2} \ldots}{0.d_1 d_2 \ldots}\right| \times 10^{-k}.$$

But $0.d_1 d_2 \cdots d_k d_{k+1} d_{k+2} \cdots \geq 0.1$,

$$\left|\frac{y - fl(y)}{y}\right| \leq \frac{1}{0.1} \times 10^{-k} = 10^{-k+1}.$$

Relative error for **rounding**

Suppose that $y = 0.d_1 d_2 \cdots d_k d_{k+1} d_{k+2} \cdots \times 10^n$, with $d_1 \geq 1$.

$$\left| \frac{y - fl(y)}{y} \right| \leq 0.5 \times 10^{-k+1}.$$

Proof: Exercise in text.

Machine addition, subtraction, multiplication, and division

$$x \oplus y = fl(fl(x) + fl(y)), \quad x \otimes y = fl(fl(x) \times fl(y)),$$

$$x \ominus y = fl(fl(x) - fl(y)), \quad x \oplus y = fl(fl(x) \div fl(y)).$$

Cancellation of significant digits

Suppose that x and y do not differ much:

$$\begin{aligned} x &= 0.d_1 \cdots d_p \alpha_{p+1} \cdots \times 10^n \\ &= 0.d_1 \cdots d_p \alpha_{p+1} \cdots \alpha_k \times 10^n + \epsilon_x = fl(x) + \epsilon_x, \\ y &= 0.d_1 \cdots d_p \beta_{p+1} \cdots \times 10^n \\ &= 0.d_1 \cdots d_p \beta_{p+1} \cdots \beta_k \times 10^n + \epsilon_y = fl(y) + \epsilon_y, \end{aligned}$$

with $\epsilon_x, \epsilon_y \approx 10^{n-k}$.

The floating-point form of $x - y$ is

$$fl(fl(x) - fl(y)) = 0.\sigma_{p+1}\sigma_{p+2} \ldots \sigma_k \times 10^{n-p},$$

where

$$0.\sigma_{p+1}\sigma_{p+2} \ldots \sigma_k = 0.\alpha_{p+1}\alpha_{p+2} \ldots \alpha_k - 0.\beta_{p+1}\beta_{p+2} \ldots \beta_k.$$

Roughly, **relative error** is

$$\left| \frac{\text{error in computing } x - y}{x - y} \right| \approx \left| \frac{|\epsilon_x| + |\epsilon_y|}{fl(fl(x) \; - \; fl(y))} \right| \approx \frac{10^{n-k}}{10^{n-p}} = 10^{-(k-p)}.$$

Quadratic formula for $ax^2 + bx + c = 0$

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{and} \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}.$$

One of x_1, x_2 faces cancellation of significant digits if

$$|4ac| \ll b^2.$$

Solving $ax^2 + bx + c = 0$ the better way

- Compute $\delta = \sqrt{b^2 - 4 * a * c}$
- If $b > 0$ then
$$x_1 = \frac{-b - \delta}{2a};$$
if $b \leq 0$ then
$$x_1 = \frac{-b + \delta}{2a}.$$
- Vieta's formula
$$x_2 = \frac{c}{a\,x_1}.$$

Roots to Quadratic to Roots (I)

```
function xx = quadroot(x)
a = 1;
b =-(x(1)+x(2));
c = x(1)*x(2);
del = sqrt(b*b-4*a*c);
xx(1) = (-b+del)/(2*a);
xx(2) = (-b-del)/(2*a);
xx =xx(:);
```

Roots to Quadratic to Roots (II)

```
>> format long e
format long e
>> x = randn(2,1)
x = randn(2,1)

x =

     1.630235289164729e+00
     4.888937703117894e-01

>> xx = quadroot(x)
xx = quadroot(x)

xx =

     1.630235289164729e+00
     4.888937703117894e-01

>> x = [randn*1e5;randn*1e-12]
x = [randn*1e5;randn*1e-12]

x =

     1.034693009917860e+05
     7.268851333832379e-13

>> xx = quadroot(x)
xx = quadroot(x)

xx =

     1.034693009917860e+05
                         0
```

Roots to Quadratic to Roots (III)

```
>> a = randn*1e-5;b = 1; c = - randn*1e-12;
a = randn*1e-5;b = 1; c = - randn*1e-12;
>> roots([a b c])
roots([a b c])

ans =

     3.295534380226372e+05
     2.938714670966580e-13

>> del = sqrt(b*b-4*a*c)
del = sqrt(b*b-4*a*c)

del =

     1

>> x(1) = (-b+del)/(2*a);x(2) = (-b-del)/(2*a)
x(1) = (-b+del)/(2*a);x(2) = (-b-del)/(2*a)

x =

                         0
     3.295534380226372e+05

>> x(2) = (-b-del)/(2*a);x(1)=(c/a)/x(2)
x(2) = (-b-del)/(2*a);x(1)=(c/a)/x(2)

x =

     2.938714670966580e-13
     3.295534380226372e+05
```

Horner's Method for Fibonacci's Problem in 1224, with his Emperor

Solve

$$f(x) = x^3 + 2x^2 + 10x - 20 = 0.$$

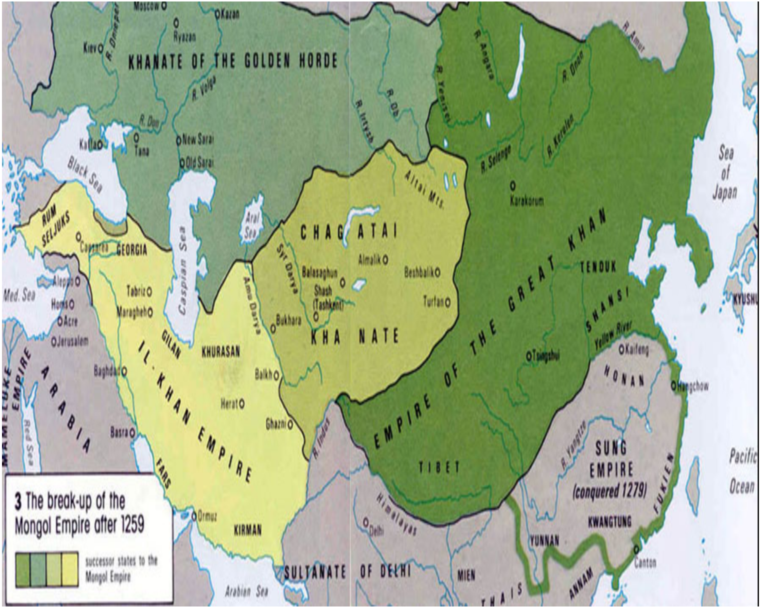
Fibonacci's Solution

$$x = 1 + 22\left(\frac{1}{60}\right) + 7\left(\frac{1}{60}\right)^2 + 42\left(\frac{1}{60}\right)^3 + 33\left(\frac{1}{60}\right)^4 + 4\left(\frac{1}{60}\right)^5 + 40\left(\frac{1}{60}\right)^6.$$

With Horner's nested sum method, let $\tau = \left(\frac{1}{60}\right)$:

$$x = 1 + \tau \cdot (22 + \tau \cdot (7 + \tau \cdot (42 + \tau \cdot (33 + \tau \cdot (4 + 40\tau))))).$$

Fibonacci's Problem 1224, what timing



3 The break-up of the Mongol Empire after 1259

Pseudocode for Horner's Method

```
function SUM = horner(x,a)
%
% horner's method
%
n = length(a);
SUM = a(n)*ones(size(x));
for i=n-1:-1:1
    SUM = a(i) + x .* SUM;
end
return
```

Numerical stability: a second order recursion

For any constants c_1 and c_2,

$$p_n = c_1\left(\frac{1}{3}\right)^n + c_2 3^n,$$

is a solution to the recursive equation

$$p_n = \frac{10}{3}p_{n-1} - p_{n-2}, \quad \text{for } n = 2, 3, \ldots .$$

- $$\lim_{n\to\infty} |p_n| = \begin{cases} \infty & \text{if} \quad c_2 \neq 0, \\ 0 & \text{otherwise.} \end{cases}$$
- $$\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \frac{1}{8}\begin{pmatrix} 9p_0 - 3p_1 \\ 3p_1 - p_0 \end{pmatrix}, \quad \text{given } p_0, p_1.$$
- condition $c_2 = 3p_1 - p_0 = 0$ **hard to satisfy exactly in finite precision computations**.

Numerical values go crazy for $p_0 = 1, p_1 = 1/3$.

With five-digit rounding arithmetic,

n	Computed $\hat{p}_n$	Correct p_n	Relative Error
0	0.10000×10^1	0.10000×10^1	
1	0.33333×10^0	0.33333×10^0	
2	0.11110×10^0	0.11111×10^0	9×10^{-5}
3	0.37000×10^{-1}	0.37037×10^{-1}	1×10^{-3}
4	0.12230×10^{-1}	0.12346×10^{-1}	9×10^{-3}
5	0.37660×10^{-2}	0.41152×10^{-2}	8×10^{-2}
6	0.32300×10^{-3}	0.13717×10^{-2}	8×10^{-1}
7	-0.26893×10^{-2}	0.45725×10^{-3}	7×10^0
8	-0.92872×10^{-2}	0.15242×10^{-3}	6×10^1

Rate of convergence: the Big O

Suppose $\{\beta_n\}_{n=1}^{\infty}$ is a sequence known to converge to zero, and $\{\alpha_n\}_{n=1}^{\infty}$ converges to a number α. If a positive constant K exists with

$$|\alpha_n - \alpha| \le K|\beta_n|, \quad \text{for large } n,$$

then we say that $\{\alpha_n\}_{n=1}^{\infty}$ converges to α with **rate, or order, of convergence** $O(\beta_n)$. (This expression is read "big oh of β_n".) It is indicated by writing $\alpha_n = \alpha + O(\beta_n)$. ■