

# UN5390: Scientific Computing I

Dr. Gowtham

Director of Research Computing, IT  
Adj. Asst. Professor, Physics and ECE

EERC B39 · [\(906\) 487-4096](tel:(906)487-4096) · [g@mtu.edu](mailto:g@mtu.edu) · [@sgowtham](https://twitter.com/@sgowtham)

Week #10: 2016/11/01 and 2016/11/03

Cross-listed as BE5390, EE5390 and MA5390

Do not share/distribute the course material, in and/or outside of Michigan Tech, without instructor's prior consent



# Recap

What we did last week, and what you were supposed to do



<http://dilbert.com/strip/1998-09-14/>

## Week #08 Recap

- \* Differential equations
  - \* Overview
  - \* Euler's method
  - \* Adams-Bashforth-Moulton methods
- \* Secret Life of Big Data
- \* No class during week #09

## Week #08 Before we meet again

- \* Review the syllabus, course material, grade through week #08, [notations](#), [active participation](#), [free time exercises](#), [tips](#), [opportunities](#), [mathematical results](#), and [videos](#)
- \* Rescale expectations for and make progress in assignment #07
- \* Discuss the project with your advisor\*
- \* Make progress in fellowship applications, if applicable
- \* Watch the PBS documentaries: [Edison](#) and [Tesla](#)
- \* Catch up on life, research, and other courses

\* Students with a research advisor.



# Differential Equations

Runge-Kutta methods

Derived by truncating the general form of the  $p^{th}$  order Taylor series expansion of  $y(t)$  with  $\tilde{t} \in [t, t + \Delta t]$

$$y(t + \Delta t) = \sum_{n=0}^p \frac{(\Delta t)^n}{n!} \frac{d^n}{dt^n} y(t) + \frac{(\Delta t)^{p+1}}{(p+1)!} \frac{d^{p+1}}{dt^{p+1}} y(\tilde{t})$$

# Euler's method

Cons

$$y_{n+1} = y_n + h f(y_n, t_n)$$



- \* Explicit method and prone to numerical instability
- \* Asymmetric w.r.t. the ends of an interval
  - Derivatives are evaluated only at  $t_n$
- \* LTE ( $\propto h^2$ ) quite large compared to other methods

Leonhard Euler (1707 – 1783): Swiss mathematician and physicist

# Adams-Bashforth-Moulton methods

Cons

$$y_{n+2}^{\text{AB2}} = y_{n+1} + \frac{h}{2} [3f(y_{n+1}, t_{n+1}) - f(y_n, t_n)]$$

$$y_{n+2}^{\text{AM2}} = y_{n+1} + \frac{h}{2} \left[ f(y_{n+2}^{\text{AB2}}, t_{n+2}) + f(y_{n+1}, t_{n+1}) \right]$$



- \* AB2 is an explicit, multi-step method prone to numerical instability that can't start on its own
- \* AM2 is an expensive, implicit and single step method
- \* LTE ( $\propto h^{k+1}$ ) for a  $k$ -step method is not small enough

John Couch Adams (1819 – 1892): British mathematician and astronomer

Francis Bashforth (1819 – 1912): British mathematician (no photo)

Forest Ray Moulton (1872 – 1952): US astronomer

- \* Single step methods
- \* Multi-stage (per time step) methods
- \* Family of explicit and implicit methods
- \* Linear differential equations
- \* Treat initial value problems
- \* Easy enough to be solved by hand
- \* Lend themselves for programmatic modeling



Carl David Tolm  Runge (1856 – 1927): German mathematician, physicist and spectroscopist

Martin Wilhelm Kutta (1867 – 1944): German mathematician

## Mathematical results

Single- and multi-variable Taylor series expansion, and chain rule of partial differentiation.

Truncate the Taylor series expansion of  $y(t)$  with for  $p = 2$ . Use  $t_{n+1} = t + \Delta t$ ,  $\Delta t = h$ , and  $y'(t) = f(y, t)$

$$y(t_{n+1}) = y(t_n) + h f(y_n, t_n) + \frac{h^2}{2} y''(t_n) + \mathcal{O}(h^3)$$

Ignore  $\mathcal{O}(h^3)$ , differentiate  $y'(t)$  w.r.t.  $t$ , and re-arrange the terms

$$y(t_{n+1}) = y(t_n) + \frac{h}{2} f(y_n, t_n) + \frac{h}{2} \left[ f + h \frac{\partial f}{\partial t} + h \frac{\partial f}{\partial y} f \right] (y_n, t_n)$$

## RK2 method

Recursive relation

Compare the last term with with the expression for multi-variable Taylor series expansion of  $f(x, y)$ , and simplify

$$y_{n+1} = y_n + \frac{h}{2} (k_1 + k_2)$$

$$k_1 = f(y_n, t_n)$$

$$k_2 = f(y_n + h k_1, t_n + h)$$

- \* Two-stage (per time step) method
  - Euler-like trial step to the mid-point of the interval
  - Use the values of  $y$  and  $t$  at the mid-point to step across the interval
- \* Symmetric integration scheme cancels first-order error making  $\text{LTE} \propto h^3$ , and  $\text{GTE} \propto h^2$

$k_i$  correspond to different estimates for the slope of the solution.



\* Round-off error = Number of time steps  $\times$  Round-off error/step

$$\epsilon_{\text{round-off}} = \mathcal{O}\left(\frac{1}{h}\right) \times \mathcal{O}(\xi) = \mathcal{O}\left(\frac{\xi}{h}\right)$$

\* Total error = Total round-off error + Total truncation error

$$\epsilon_{\text{total}} \sim \frac{\xi}{h} + h^s$$

Optimum value of  $h$  is

$$h_o = (\xi/s)^{\frac{1}{s+1}} = \mathcal{O}\left(\xi^{\frac{1}{s+1}}\right)$$

$\xi$  corresponds to machine epsilon. Each time step can be treated as a floating-point operation.  
Round-off error dominates the total error for smaller values of  $h$ , and truncation error for larger values of  $h$ .

# Runge-Kutta (RKs) method

Pros and cons

- \* Self starting, single step, multi-stage method
- \* Explicit method but multiple stages per time step add stability
- \* LTE ( $\propto h^{s+1}$ ) quite small compared to other methods
- \* Less expensive compared to multi-step methods for similar accuracy
- \* Round-off error is less of a limiting factor compared to at least  $s$  evaluations of  $f(y, t)$  per time step

# of evaluations of $f$ per time step	2	3	4	5	6	7	8	9	10	11
Maximum achievable $s$	2	3	4	4	5	7	6	7	7	8

- \*  $f(y, t)$  evaluation can be expensive



# Runge-Kutta (RK4) method

General recursive expression and associated stages

$$y_{n+1} = y_n + \frac{h}{6} (k_1 + 2 k_2 + 2 k_3 + k_4)$$

$$k_1 = f(y_n, t_n)$$

$$k_2 = f\left(y_n + \frac{h}{2} k_1, t_n + \frac{h}{2}\right)$$

$$k_3 = f\left(y_n + \frac{h}{2} k_2, t_n + \frac{h}{2}\right)$$

$$k_4 = f(y_n + h k_3, t_n + h)$$

## Pros and cons

- \* Pros and **cons** of RKs method
- \* Easy enough to implement
- \* Acceptable balance between computational cost associated with  $f(y, t)$  evaluations and accuracy of the result
  - \* Four evaluations per time step
  - \* Local truncation error,  $\mathcal{O}(h^5)$ , and global truncation error,  $\mathcal{O}(h^4)$
  - \* Optimum/Practical value,  $h_0 \simeq \xi^{1/5}$  ( $\simeq 10^{-5}$  for double precision)
  - \* Corresponding minimum error,  $\epsilon_0 \simeq \xi^{4/5}$  ( $\simeq 10^{-13}$  for double precision)
- \* Very popular, and frequently used

# Specific case

First order linear differential equation

$$\frac{dy}{dt} = f(y, t)$$

First order linear DE

$$y(t_0) = y_0$$

Initial condition

$$y_{n+1} = y_n + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4)$$

$$k_1 = f(y_n, t_n)$$

$$k_2 = f\left(y_n + \frac{h}{2}k_1, t_n + \frac{h}{2}\right)$$

$$k_3 = f\left(y_n + \frac{h}{2}k_2, t_n + \frac{h}{2}\right)$$

$$k_4 = f(y_n + h k_3, t_n + h)$$

# Specific case

Coupled first order linear differential equations

$$\frac{dx}{dt} = f_x(x, y, t) \quad \frac{dy}{dt} = f_y(x, y, t)$$

Coupled first order linear DEs

$$x(t_0) = x_0 \quad y(t_0) = y_0$$

Initial conditions

$$x_{n+1} = x_n + \frac{h}{6} (k_1^x + 2 k_2^x + 2 k_3^x + k_4^x)$$

$$y_{n+1} = y_n + \frac{h}{6} (k_1^y + 2 k_2^y + 2 k_3^y + k_4^y)$$

# Specific case

Coupled first order linear differential equations

$$k_1^x = f_x(x_n, y_n, t_n)$$

$$k_2^x = f_x \left( x_n + \frac{h}{2} k_1^x, y_n + \frac{h}{2} k_1^y, t_n + \frac{h}{2} \right)$$

$$k_3^x = f_x \left( x_n + \frac{h}{2} k_2^x, y_n + \frac{h}{2} k_2^y, t_n + \frac{h}{2} \right)$$

$$k_4^x = f_x \left( x_n + h k_3^x, y_n + h k_3^y, t_n + h \right)$$

# Specific case

Coupled first order linear differential equations

$$k_1^y = f_y(x_n, y_n, t_n)$$

$$k_2^y = f_y \left( x_n + \frac{h}{2} k_1^x, y_n + \frac{h}{2} k_1^y, t_n + \frac{h}{2} \right)$$

$$k_3^y = f_y \left( x_n + \frac{h}{2} k_2^x, y_n + \frac{h}{2} k_2^y, t_n + \frac{h}{2} \right)$$

$$k_4^y = f_y \left( x_n + h k_3^x, y_n + h k_3^y, t_n + h \right)$$



# Specific case

Second order linear differential equation

$$\frac{d^2y}{dt^2} = f(y, \dot{y}, t)$$

Second order linear DE

$$\left. \frac{dy}{dt} \right|_{t_0} = \dot{y}(t_0) = \dot{y}_o \quad y(t_0) = y_o$$

Initial conditions

## Useful substitutions

$$p(t) = y(t) \Rightarrow \dot{p}(t) = \dot{y}(t) \quad q(t) = \dot{y}(t) \Rightarrow \dot{q}(t) = \ddot{y}(t)$$
$$\dot{p}(t) = q(t) \quad \dot{q}(t) = f(p, q, t)$$

---

$$q_{n+1} = q_n + \frac{h}{6} (k_1^q + 2k_2^q + 2k_3^q + k_4^q)$$

$$p_{n+1} = p_n + \frac{h}{6} (k_1^p + 2k_2^p + 2k_3^p + k_4^p)$$



# Specific case

Second order linear differential equation

$$k_1^q = f(p_n, q_n, t_n)$$

$$k_2^q = f\left(p_n + \frac{h}{2} k_1^p, q_n + \frac{h}{2} k_1^q, t_n + \frac{h}{2}\right)$$

$$k_3^q = f\left(p_n + \frac{h}{2} k_2^p, q_n + \frac{h}{2} k_2^q, t_n + \frac{h}{2}\right)$$

$$k_4^q = f(p_n + h k_3^p, q_n + h k_3^q, t_n + h)$$

# Specific case

Second order linear differential equation

$$k_1^p = q_n$$

$$k_2^p = q_n + \frac{h}{2} k_1^q$$

$$k_3^p = q_n + \frac{h}{2} k_2^q$$

$$k_4^p = q_n + h k_3^q$$

# Specific case

Coupled second order linear differential equations

$$\frac{d^2x}{dt^2} = f_x(x, \dot{x}, y, \dot{y}, t) \quad \frac{d^2y}{dt^2} = f_y(x, y, \dot{x}, \dot{y}, t)$$

Second order linear DE

$$\dot{x}(t_0) = \dot{x}_0 \quad x(t_0) = x_0 \quad \dot{y}(t_0) = \dot{y}_0 \quad y(t_0) = y_0$$

Initial conditions

# Specific case

Coupled second order linear differential equations

$$\dot{x}_{n+1} = \dot{x}_n + \frac{h}{6} (C_1 + 2 C_2 + 2 C_3 + C_4)$$

$$\dot{y}_{n+1} = \dot{y}_n + \frac{h}{6} (D_1 + 2 D_2 + 2 D_3 + D_4)$$

$$x_{n+1} = x_n + \frac{h}{6} (A_1 + 2 A_2 + 2 A_3 + A_4)$$

$$y_{n+1} = y_n + \frac{h}{6} (B_1 + 2 B_2 + 2 B_3 + B_4)$$

# Specific case

Coupled second order linear differential equations

$$A_1 = \dot{x}_n$$

$$A_2 = \dot{x}_n + \frac{h}{2} C_1$$

$$A_3 = \dot{x}_n + \frac{h}{2} C_2$$

$$A_4 = \dot{x}_n + h C_3$$



# Specific case

Coupled second order linear differential equations

$$B_1 = \dot{y}_n$$

$$B_2 = \dot{y}_n + \frac{h}{2} D_1$$

$$B_3 = \dot{y}_n + \frac{h}{2} D_2$$

$$B_4 = \dot{y}_n + h D_3$$



# Specific case

Coupled second order linear differential equations

$$C_1 = f_x(x_n, \dot{x}_n, y_n, t_n)$$

$$C_2 = f_x \left( x_n + \frac{h}{2} A_1, \dot{x}_n + \frac{h}{2} C_1, y_n + \frac{h}{2} B_1, t_n + \frac{h}{2} \right)$$

$$C_3 = f_x \left( x_n + \frac{h}{2} A_2, \dot{x}_n + \frac{h}{2} C_2, y_n + \frac{h}{2} B_2, t_n + \frac{h}{2} \right)$$

$$C_4 = f_x (x_n + h A_3, \dot{x}_n + h C_3, y_n + h B_3, t_n + h)$$

# Specific case

Coupled second order linear differential equations

$$D_1 = f_y(x_n, y_n, \dot{y}_n, t_n)$$

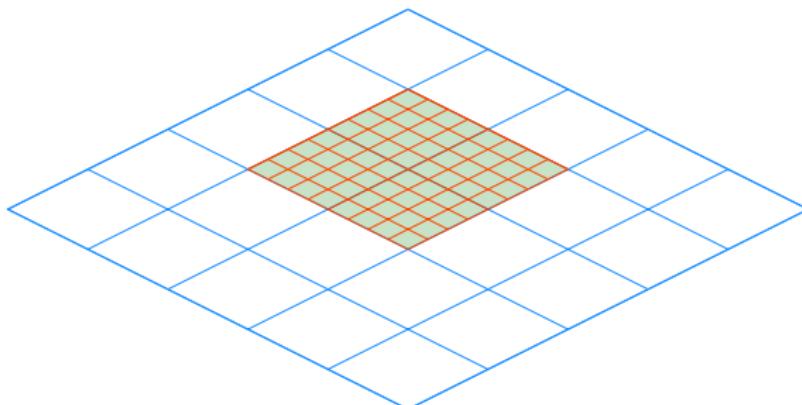
$$D_2 = f_y\left(x_n + \frac{h}{2} A_1, y_n + \frac{h}{2} B_1, \dot{y}_n + \frac{h}{2} D_1, t_n + \frac{h}{2}\right)$$

$$D_3 = f_y\left(x_n + \frac{h}{2} A_2, y_n + \frac{h}{2} B_2, \dot{y}_n + \frac{h}{2} D_2, t_n + \frac{h}{2}\right)$$

$$D_4 = f_y(x_n + h A_3, y_n + h B_3, \dot{y}_n + h D_3, t_n + h)$$

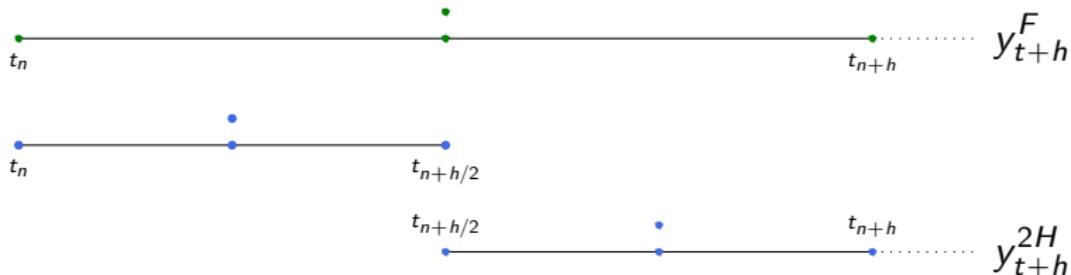
## Adaptive time steps

- \* Adaptive control over its own progress to achieve a predetermined accuracy with minimum computational cost
- \* Finer grid (i.e., smaller  $h$ ) for interesting regions, and coarse grid (i.e., larger  $h$ ) for not so interesting areas
- \* Integration scheme needs to be made aware of the truncation error



# Adaptive time steps

- \* Step doubling or Double stepping
  - \* One full step, from  $t_n$  to  $t_{n+h}$   
4 evaluations of  $f(y, t)$  to compute  $y_{t+h}^F$
  - \* Two half steps, from  $t_n$  to  $t_{n+h/2}$  and then from  $t_{n+h/2}$  to  $t_{n+h}$   
8 evaluations of  $f(y, t)$  to compute  $y_{t+h}^{2H}$
  - \* Same starting point,  $t_n$ , results in 11 evaluations of  $f(y, t)$   
Computational overhead: 11/8, +37.5%



## Adaptive time steps

- \* Estimate of truncation error associated with each time step

$$\epsilon = \left| y_{t+h}^F - y_{t+h}^{2H} \right|$$

- \*  $\epsilon_o$  is the acceptable value for truncation error with each time step, and local truncation error in RK4 method,  $\epsilon = \mathcal{O}(h^{4+1})$

$$h = \mathcal{O}\left(\epsilon^{\frac{1}{4+1}}\right)$$

- \* Adaptive time step formula

$$h_{\text{new}} = h_{\text{current}} \left| \frac{\epsilon_o}{\epsilon} \right|^{\frac{1}{4+1}}$$

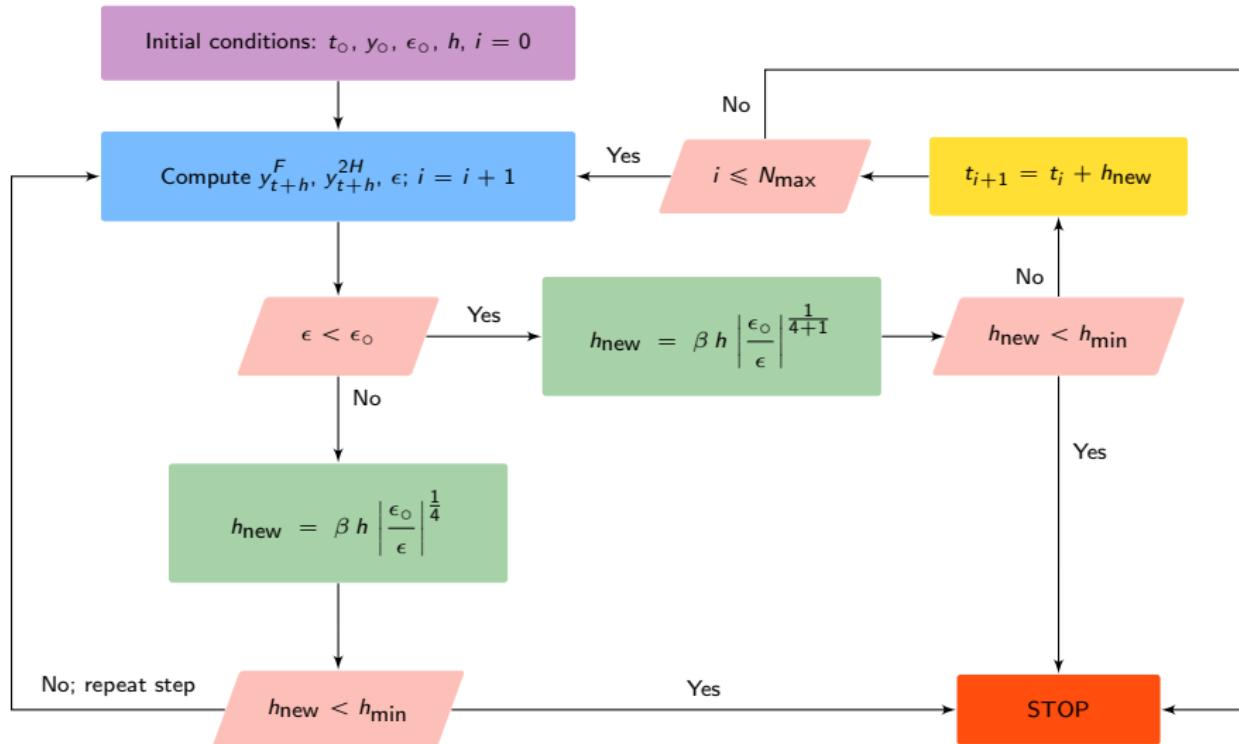
Increase  $h$  if  $\epsilon$  is too small, and decrease otherwise

## Adaptive time steps

- \* Safety factor to cure the inexactness of the truncation error estimate  
 $\beta \simeq 1$  (usually 0.80 or 0.90)
- \* Choice of  $\epsilon_0$  depends on the IVP. Setting  $\epsilon_0 \propto h$  makes the exponent,  $1/(4 + 1)$ , while estimating  $h$  incorrect. If  $h_{\text{new}}$  becomes too small, then the computation will fail to meet the desired accuracy
- \* Adaptive time step formula

$$h_{\text{new}} = \beta h_{\text{current}} \begin{cases} \left| \frac{\epsilon_0}{\epsilon} \right|^{\frac{1}{4+1}} & \epsilon \geq \epsilon_0 \\ \left| \frac{\epsilon_0}{\epsilon} \right|^{\frac{1}{4}} & \epsilon < \epsilon_0 \end{cases}$$

# Adaptive time steps



## Adaptive time steps

- \* Benefit must outweigh the 37.5% computational overhead
  - Rely on computational experience
  - Gain computational experience by solving a variety of problems
- \* System of coupled differential equations involves an appropriately weighted average of error associated with each equation
  - Absolute error works for a system in which the amplitude of dependent variables remains bounded
  - Relative error works for a system in which the amplitude of dependent variables blows up at some point but variables retain the same sign
  - The minimum of absolute and relative errors works for a system in which the amplitude of dependent variables blows up at some point and the signs of variables oscillate

## Additional references

- \* Error Bounds For The Runge-Kutta Single-Step Integration Process  
J. W. Carr III  
Journal of the ACM, vol. 5, p. 39 (1958)
- \* Runge-Kutta Methods With Minimum Error Bounds  
A. Ralston  
Mathematics of Computation, vol. 16, p. 431 (1962)
- \* Coefficients For The Study Of Runge-Kutta Integration Processes  
J. C. Butcher  
Journal of the Australian Mathematical Society, vol. 3, p. 185 (1963)
- \* Runge-Kutta Methods With Constrained Minimum Error Bounds  
R. King  
Mathematics of Computation, vol. 20, p. 386 (1966)

PDF in [AdditionalMaterial](#) folder.



## Additional references

- \* Conditions For The Coefficients Of Runge-Kutta Methods For Systems Of n-th Order Differential Equations  
H. M. Hebsaker  
Journal of Computational and Applied Mathematics, vol. 8, p. 3  
(1982)
- \* Stability Of Explicit Runge-Kutta Methods  
L. F. Shampine  
Computers and Mathematics with Applications, vol. 10, p. 419  
(1984)
- \* Stiffness And The Automatic Selection Of ODE Codes  
L. F. Shampine  
Journal of Computational Physics, vol. 54, p. 74 (1984)

PDF in [AdditionalMaterial](#) folder.



## Additional references

- \* Automatic Selection Of The Initial Step Size For An ODE Solver  
I. Gladwell, L. F. Shampine, R. W. Brankin  
Journal of Computational and Applied Mathematics, vol. 18, p. 175  
(1987)
- \* A Simple Step Size Selection Algorithm for ODE Codes  
L. F. Shampine, A. Witt  
Journal of Computational and Applied Mathematics, vol. 58, p. 345  
(1995)
- \* An Interview With Anthony Ralston  
B. de Neumann, W. Aspray  
Society for Industrial and Applied Mathematics (2006)

PDF in [AdditionalMaterial](#) folder.



## Additional references

- \* A Simplified Derivation And Analysis Of Fourth Order Runge Kutta Method  
H. Musa, I. Saidu, M. Y. Waziri  
International Journal of Computer Applications, vol. 9, p. 51 (2010)
- \* Stepsize Selection In Explicit Runge-Kutta Methods For Moderately Stiff Problems  
J. S. C. Prentice  
Applied Mathematics, vol. 2, p. 711 (2011)
- \* Learning Through Computational Creativity  
L. Soh, D. F. Shell, E. Ingraham, S. Ramsay, B. Moore  
Communications of the ACM, vol. 58, p. 33 (2015)
- \* 11 Calculators And 2000 Years Of Computing

PDF in [AdditionalMaterial](#) folder.

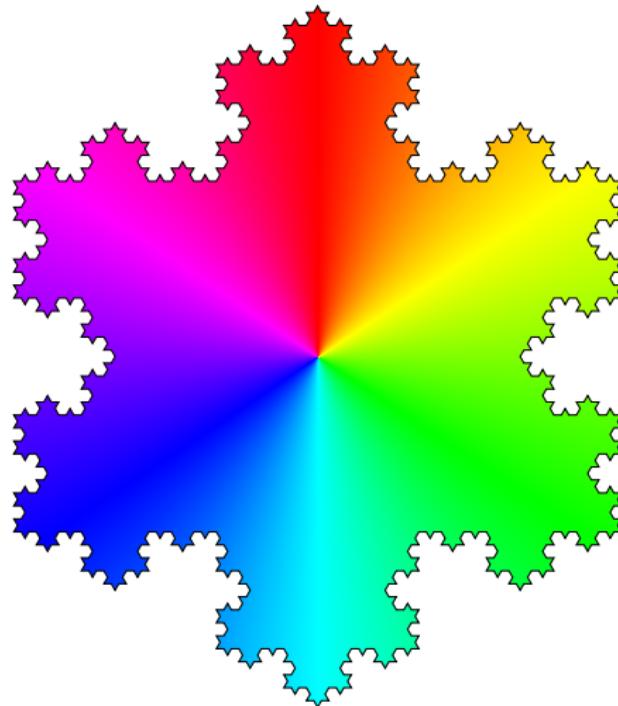


## Before we meet again

- \* Review the syllabus, course material, grade through week #08, notations, active participation, free time exercises, tips, opportunities, mathematical results, and videos
- \* Make progress in assignment #07
- \* Get started on term project\*  
**MS course work students: meet the instructor before noon on Friday**
- \* Turn in the project status report by Friday 4:59 pm
- \* Review matrix methods and numerical techniques to solve them from prior courses, if any

\* Inform the instructor if you are applying for a fellowship (include the name, organization and due date).





End of Tuesday lecture.

# Matrices

A rectangular array of numbers, symbols, expressions or other mathematical objects arranged in rows and columns, and the individual entities are known as *elements* or *entries* of the matrix.  $a_{ij}$  represents such an element in the  $i^{\text{th}}$  row and  $j^{\text{th}}$  column.

# Commonly used methods

- \* Analytical methods
  - \* Prohibitively time consuming and error prone for higher orders
- \* Numerical methods
  - \* Arithmetic
  - \* Determinants and trace
  - \* Diagonalization and eigen values

# Domain-specific applications

- \* Computer graphics and message encryption
- \* Electronics (mesh analysis)
- \* Game theory (payoff matrix)
- \* Geometry (rotation matrix)
- \* Graph theory (logical/cost matrix)
- \* Linear algebra (system of equations)
- \* Probability theory and statistics (stochastic matrix)
- \* Sciences and engineering (classical and quantum)

The above list is just a token representation of inexhaustive list of fields that use matrices.

# Matrices

LU factorization

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{pmatrix}$$

# LU factorization

## Mathematical results

### Systems of linear equations and Gaussian elimination

\* Factor a matrix as  $A = LU$

\* Partial pivoting (only row permutations)

$$PA = LU$$

\* Solving for  $x$  in  $Ax = b$  becomes

$$Ly = Pb \text{ (solving for } y\text{) and}$$

$$Ux = y \text{ (solving for } x\text{)}$$



Johann Carl Friedrich Gauss (1777 – 1855): German mathematician

Alan Mathison Turing (1912 – 1954): British mathematician, computer scientist and philosopher

# Top 500

The top 10

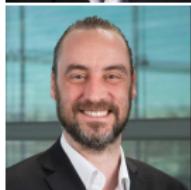
TFLOPS

#	Name	Location	Processors	Expt	Theory	Power (MW)
01	Sunway TaihuLight	2016, NSCW, China	10,649,600	93,014.6	125,435.9	15.371
02	Tianhe-2	2013, NSCG, China	3,120,000	33,862.7	54,902.4	17.808
03	Titan	2012, ORNL, USA	560,640	17,590.0	27,112.5	8.209
04	Sequoia	2011, LLNL, USA	1,572,864	17,173.2	20,132.7	7.890
05	K	2011, RIKEN, Japan	705,024	10,510.0	11,280.4	12.660
06	Mira	2012, ANL, USA	786,432	8,586.6	10,066.3	3.945
07	Trinity	2015, LANL, USA	301,056	8,100.9	11,078.9	–
08	Piz Daint	2013, CSCS, Switzerland	115,984	6,271.0	7,788.9	2.325
09	Hazel Hen	2015, HLRS, Germany	185,088	5,640.2	7,403.5	–
10	Shaheen II	2015, KAUST, Saudi Arabia	196,608	5,537.0	7235.2	2.834



<http://top500.org/lists/2016/06/>

- \* Solves a dense system of linear equations
- \* A measure of *best* performance
- \* Algorithm must conform to LU factorization with partial pivoting



$$\# \text{ of double-precision floating-point operations} = \frac{2}{3} N^3 + \mathcal{O}(N^2)$$

Matrix diagonalization isn't all that we do

– Dr. James Cuff (@JamesDotCuff)

<http://www.top500.org/project/linpack/>

<http://sgowtham.com/journal/hpl-benchmark-for-single-processor-machines/>

Jack J. Dongarra (1950 – present): American mathematician and computer scientist

James Andrew Cuff (1974 – present): British biophysicist, and assistant dean for research computing at Harvard University

# Matrices

Message encryption

When people talk to each other, they never say what they mean ...  
They say something else and you're expected to just know what  
they mean.

– Alan Turing's boyhood character in *The Imitation Game*

# Hill cipher

## Mathematical results

Matrix multiplication, determinant, adjacent matrix and inversion.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
O	P	Q	R	S	T	U	V	W	X	Y	Z	!	?	,
15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

## Brainstorm

How could one encrypt the following message, and upon receipt, how could The Queen decrypt it – using matrices?

GOD SAVE MY QUEEN, AMEN EH?

## Hill cipher

- \* Represent the message in a  $3 \times 9$  matrix

$$B = \begin{pmatrix} G & O & D & S & A & V & E & & \\ M & Y & & Q & U & E & E & N & , \\ A & M & E & N & & E & H & ? \end{pmatrix}$$

- \* Replace the characters with numbers from the coding table

$$B = \begin{pmatrix} 7 & 15 & 4 & 0 & 19 & 1 & 22 & 5 & 0 \\ 13 & 25 & 0 & 17 & 21 & 5 & 5 & 14 & 29 \\ 0 & 1 & 13 & 5 & 14 & 0 & 5 & 8 & 28 \end{pmatrix}$$

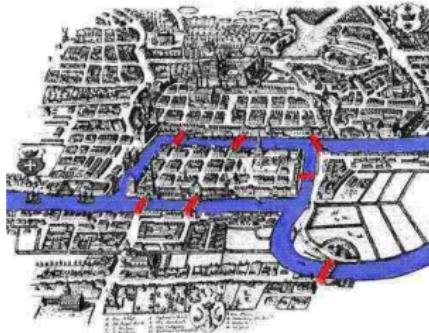
- \* Choose a  $3 \times 3$  matrix  $A$  such that  $A \times A^{-1} = I$

## Hill cipher

- \*  $A$  serves as the encoder and  $A^{-1}$  as the decoder
- \* Perform matrix multiplication,  $A \times B = C$
- \*  $C$  is the encrypted message, in  $\text{mod } 30$
- \* Send the coding table,  $C$  and  $A^{-1}$  to the Queen
- \* The Queen will perform  $A^{-1} \times C = A^{-1} \times (A \times B) = B$
- \*  $B$  is the decrypted (i.e., the original) message, in  $\text{mod } 30$

# Matrices

Graph theory



# Seven bridges of Königsberg

## Mathematical results

Adjacency matrix.

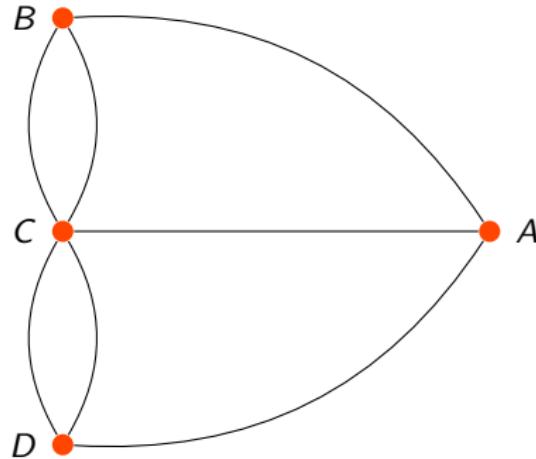
## Seven bridges of Königsberg

The city of Königsberg in Prussia (now Kaliningrad, Russia) was set on both sides of the Pregel River, and included two large islands which were connected to each other and the mainland by seven bridges.

Find a walk through the city that would cross each bridge once and only once.

# Seven bridges of Königsberg

- \* Replace each land mass with an abstract *vertex* or *node*
- \* Replace each bridge with an abstract connection, *edge*



Leonhard Euler (1707 – 1783): Swiss mathematician and physicist  
Euler proved that this problem had no solution in 1736 while laying the foundation for *graph theory*.

# Seven bridges of Königsberg

	A	B	C	D
A	-	✓	✓	✓
B	✓	-	✓	✗
C	✓	✓	-	✓
D	✓	✗	✓	-

## Brainstorm

Given the above check n' cross table, how would one go about incorporating linear algebra and matrices for a more compact representation?

## Adjacency matrix

Adjacency matrix elements for a graph with  $n$  vertices –  $v_1, v_2, \dots, v_n$ ,

$$a_{ij} = \begin{cases} 1 & \text{if there is an edge between } v_i \text{ and } v_j \\ 0 & \text{otherwise} \end{cases}$$

For the seven bridges of Königsberg

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

$(i, j)$  element in  $A^r$  represents the number of distinct  $r$  connections between  $v_i$  and  $v_j$ .

# Nature of problems

## Existence

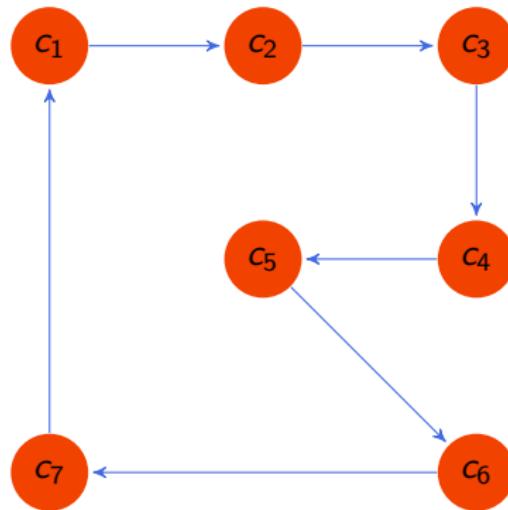
Is it possible to find a route that starts from and returns to a given  $c_i$  while visiting the remaining  $c_j$  exactly once?



# Nature of problems

## Construction

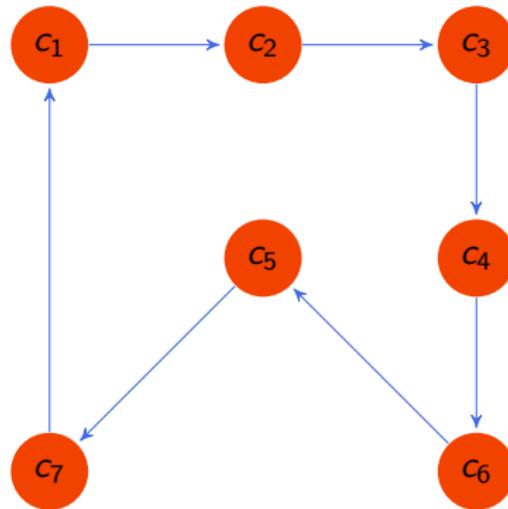
If possible, how can one construct it?



# Nature of problems

## Enumeration

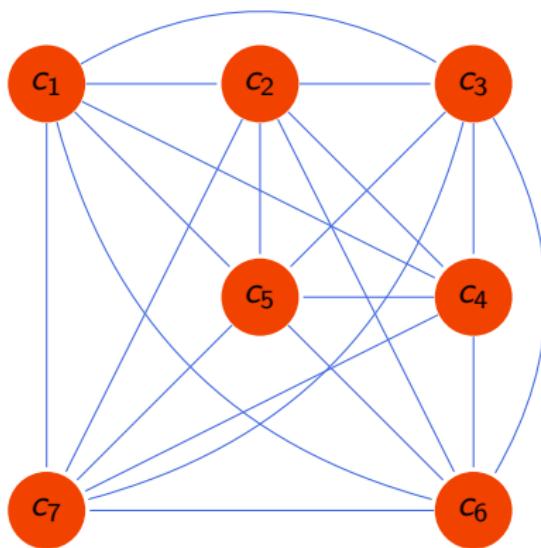
If one can construct it, are there more ways than one to do so?



# Nature of problems

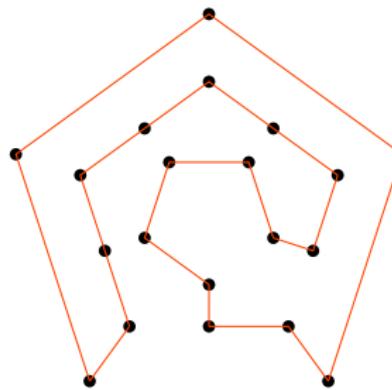
## Optimization

If there is more than one way, which one is optimal?



## Hamilton cycle

A route that visits each vertex in a graph once. It doesn't exist for every graph, and there is no specific way to check for its existence.



# Traveling purchaser and salesman problems

## Traveling Purchaser Problem (TPP)

Given a list of marketplaces, the cost of traveling between different marketplaces and a list of available items together with the price of each such item at each marketplace, find the route with minimum combined cost of purchases and traveling for a given list of items.

## Traveling Salesman Problem (TSP; a special case of TPP)

Given a list of cities and the distances between each pair of cities, find the shortest possible route that passes through each city exactly once and returns to the city of origin.

# Traveling salesman beer connoisseur problem

	KBC (3)	Library (5)	Cognition (2)	Blackrocks (1)	Ore Dock (6)	Shooters (7)	Upper Hand (10)	Lake Superior (4)	Tahquamenon (9)	Soo (8)
KBC	—	1	86	100	100	142	160	203	227	265
Library	1	—	85	99	99	141	159	202	226	264
Cognition	86	85	—	16	16	56	77	117	141	215
Blackrocks	100	99	16	—	1	43	66	103	127	165
Ore Dock	100	99	16	1	—	43	66	103	127	165
Shooters	142	141	56	43	43	—	63	60	84	122
Upper Hand	160	159	77	66	66	63	—	118	136	174
Lake Superior	203	202	117	103	103	60	118	—	74	111
Tahquamenon	227	226	141	127	127	84	136	74	—	76
Soo	265	264	215	165	165	122	174	111	76	—

The breweries are numbered alphabetically and listed in order of their increasing distance from KBC.



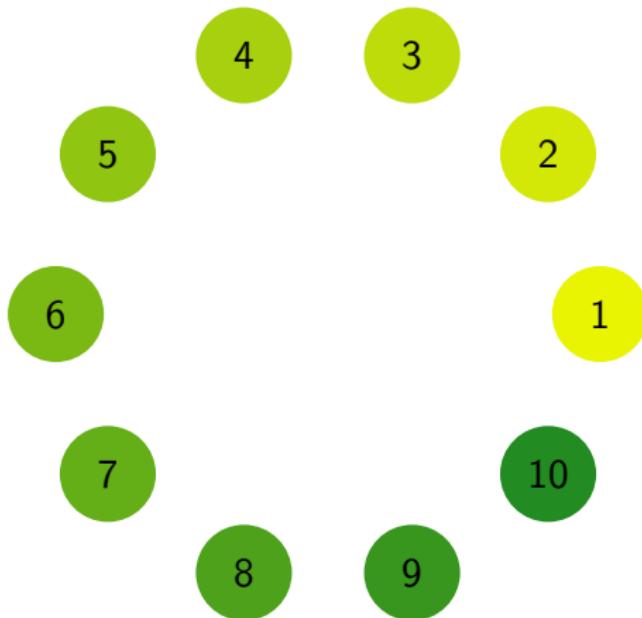
# Traveling salesman beer connoisseur problem

## Nearest neighbor algorithm

1. Start from a random vertex as the current vertex,  $c$  and distance,  $d = 0$
2. Find the shortest edge connecting the current vertex and an unvisited vertex,  $v$
3. Move to the unvisited vertex,  $v$  and add to the distance traveled,  $d$
4. Set it as the current vertex,  $c$  and mark it as visited
5. If all vertices are marked visited, then display final  $d$  and terminate the program; if not, move to step (2)

How could one break the tie for distance?

# Traveling salesman beer connoisseur problem



Starting from #1, complete the above graph using *nearest neighbor algorithm*.



# Queuing system

*Things To Do* list in our everyday lives

Date 2016|11|03|4

Pri	Task	Due	Y/N
H	UN5390: Cleanup, re-org, review, teach	noon	
H	Superior/Portage - upgrade plan review	11/04	
H	Dell: Get ETA for InfiniBand system	11 am	
H	UN5390: Review #07 pre-submissions	11 pm	
M	Renew IEEE, ACM and SIGHPC memberships	5 pm	
M	Intel HPC Developer Conf: Review 2015	3 pm	
H	Review data science from Johns Hopkins	10 pm	
M	Read CACM and IEEE Spectrum journals	11 pm	
H	Portage - test new GPUs and KNL MICs	12/05	
M	Complete panel review preparations	12/10	
M	HTTP → HTTPS HPC websites	12/31	



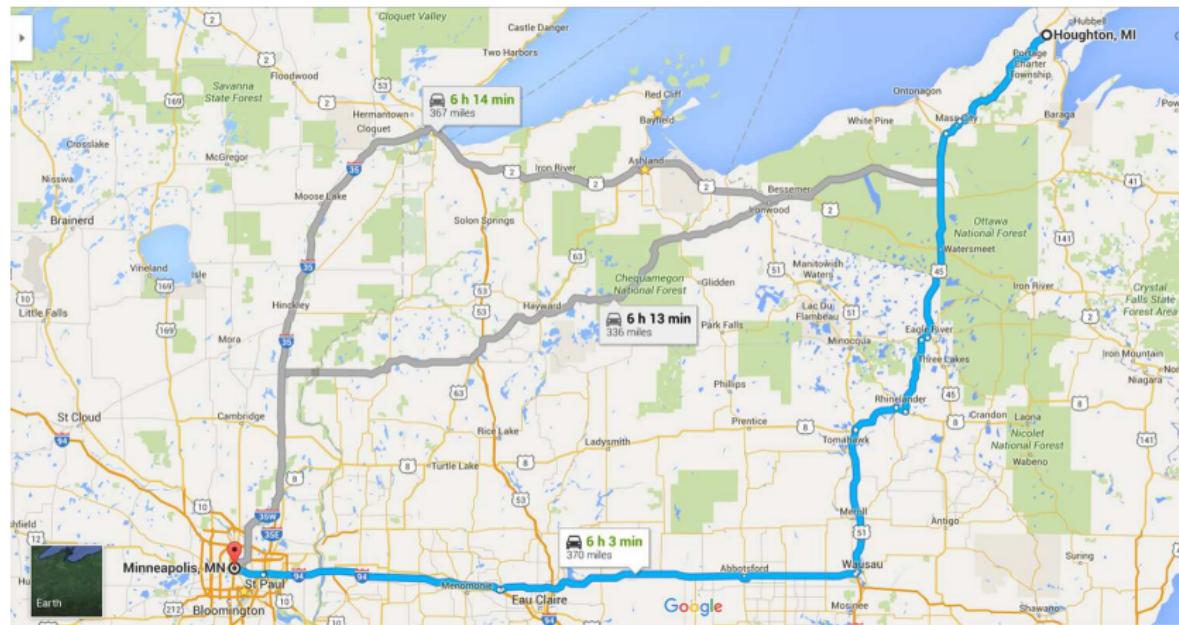
- \* Priority
  - Who are you? What have you done so far? When did you submit the job?
- \* Resource availability
  - \* Compute
    - Processors (i.e., slots) | RAM | Disk space
  - \* Software
    - FOSS | \$ but campus/site license | \$ but limited number
- \* Policies and procedures (e.g., # of concurrent jobs allowed per user)
- \* Interdependency/Conflict amongst jobs

The graph will contain a vertex/node for each job, and an edge for every conflicting pair of jobs.

[Kickstart graph](#) and [Roll Developer's Guide](#) in Rocks Cluster Distribution



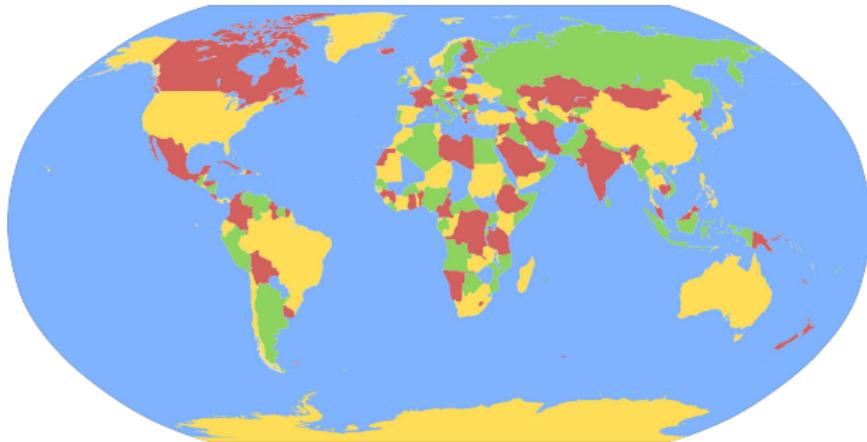
# Routing



<http://google.com/maps>



# Four-color theorem



## Four-Color Theorem (1852)

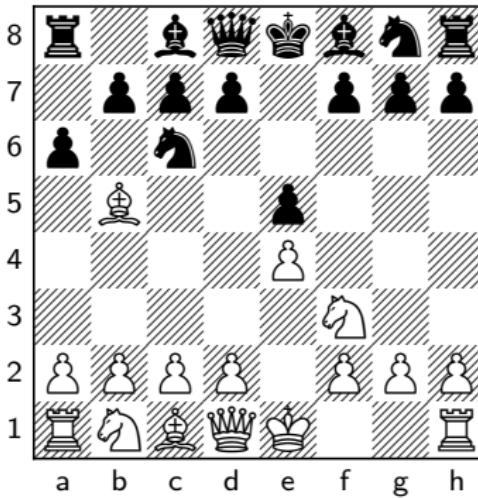
Any map in a plane can be colored using four colors in such a way that regions sharing a common boundary (other than a single point) do not share the same color.

Francis Guthrie (1831 – 1899): South African mathematician and botanist

In a map of US, UT and AZ are adjacent. UT and NM, which only share a point that also belongs to AZ and CO, are not.

# Matrices

Game theory



# Game theory

## Mathematical results

Payoff matrix.

Also known as the *interactive decision theory*, it is the study of mathematical models of conflict and cooperation between intelligent rational decision-makers.

## Nash equilibrium

A solution concept of a non-cooperative game involving two or more players, in which each player is assumed to know the equilibrium strategies of the other players, and no player has anything to gain by changing only their own strategy.

Félix Édouard Justin Émile Borel (1871 – 1956): French mathematician

John von Neumann (1903 – 1957): Austrian-Hungarian/American mathematician

John Forbes Nash, Jr. (1928 – 2015): American mathematician



# Rock, Paper, Scissors

		Player B		
		Rock	Paper	Scissors
Player A	Rock	Tie	Loss	Win
	Paper	Win	Tie	Loss
	Scissors	Loss	Win	Tie



## Brainstorm

Two players, A and B, at each turn produce a gesture of the hand. Rock beats scissors, but is beaten by paper while scissors beat paper. If A and B make the same gesture, the round results in a tie. If the loser is required to pay \$1 to the winner at each turn, summarize the game in a compact and elegant fashion.

## Payoff matrix

If row and column labels represent the rock, paper and scissors moves for A and B respectively, then the payoff matrix representing the game is given by

$$P = \begin{pmatrix} 0 & -1 & 1 \\ 1 & 0 & -1 \\ -1 & 1 & 0 \end{pmatrix}$$

### Two-person, zero-sum game

A game in which one player's loss equals the other's gain.

Generalizing, if A has  $m$  finite options to choose from while B has  $n$ , the payoff matrix representing the game will be a  $m \times n$  matrix indicating the result of every possible pair of moves.

# Heist of the Husky Statue

## Brainstorm

Two students with a known history of mischief, A and B, are arraigned as independent suspects in *Heist of the Husky Statue*. Though they cannot communicate with each other while in solitary custody, they have the following information:

1. if both of them confess, each gets a three year probation
2. if A confesses and B denies, A gets a one year probation and B gets expelled, and vice versa
3. if both of them deny, each gets a two year probation

What is the optimal course of action for the two suspects? What are A and B most likely to do assuming they have never known each other?

# Heist of the Husky Dog

		Suspect B	
		Confess	Deny
Suspect A	Confess	A and B get three year probation	A gets one year probation; B is expelled
	Deny	A is expelled; B gets one year probation	A and B get two year probation



# Additional references

LU factorization

- \* Rounding-Off Errors In Matrix Processes  
A. M. Turing  
Journal of Mechanics and Applied Mathematics, vol. 1, p. 287  
(1948)
- \* Squeezing The Most Out Of Eigenvalue Solvers On High-Performance Computers  
J. J. Dongarra, L. Kaufman, S. Hammarling  
Linear Algebra and Its Applications, vol. 77, p. 113 (1986)
- \* Linear Algebra On High-Performance Computers  
J. J. Dongarra, D. C. Sorensen  
Applied Mathematics and Computation, vol. 20, p. 57 (1986)

PDF in AdditionalMaterial folder.



## Additional references LU factorization

- \* A Tool To Aid In The Design, Implementation, And Understanding Of Matrix Algorithms For Parallel Processors

J. J. Dongarra, O. Brewer, J. A. Kohl, S. Fineberg

Journal of Parallel and Distributed Computing, vol. 9, p. 185 (1990)

- \* Integer LU Factorization

J. H. Bevis, F. J. Hall

Linear Algebra and Its Applications, vol. 150, p. 267 (1991)

PDF in AdditionalMaterial folder.



## Additional references

Message encryption

- \* [Cryptography In An Algebraic Alphabet](#)  
L. S. Hill  
The American Mathematical Monthly, vol. 36, p. 306 (1929)
- \* [Concerning Certain Linear Transformation Apparatus Of Cryptography](#)  
L. S. Hill  
The American Mathematical Monthly, vol. 38, p. 131 (1931)
- \* [Harry Potter And The Cryptography With Matrices](#)  
B. L. Chua  
The Educational Resources Information Center, vol. 62, p. 25 (2006)

PDF in [AdditionalMaterial](#) folder.



## Additional references

Message encryption

- \* Efficient Solutions Of Coupled Matrix And Matrix Differential Equations  
Z. Al-Zhour  
Intelligent Control and Automation, vol. 3, p. 176 (2012)
- \* Alan Turing: The Enigma  
A. Hodges; Princeton University Press (2002)
- \* The Imitation Game ([iTunes](#), movie, 2014)  
Based on *Alan Turing: The Enigma* by A. Hodges
- \* Cryptography, IEEE Global History Network
- \* History Of Encryption, SANS Institute (2001)
- \* National Security Agency (NSA)

PDF in [AdditionalMaterial](#) folder.

## Additional references

Graph theory

- \* Traveling Salesman Problem (visualization)
- \* New Shortcuts For Infamous Traveling Salesman Problem
- \* Traveling Salesman (movie: 2012 intellectual thriller; 1921 comedy)
- \* The Traveling Salesman Problem – A Computational Study  
D. L. Applegate, R. E. Bixby, V. Chvátal, W. J. Cook  
Princeton University Press (2006)
- \* Physicist's Version Of Traveling Salesman Problem: Statistical Analysis  
R. S. Armour Jr., J. A. Wheeler  
American Journal of Physics, vol. 51, p. 405 (1983)

PDF in AdditionalMaterial folder.



- \* Optimization By Simulated Annealing  
S. Kirkpatrick, C. D. Gelatt Jr., M. P. Vecchi  
Science, vol. 220, p. 4598 (1983)
- \* Scheduling Jobs With Fixed Start And End Times  
E. M. Arkin, E. B. Silverberg  
Discrete Applied Mathematics, vol. 18 , p. 1 (1987)
- \* Ant Colonies For The Traveling Salesman Problem  
M. Dorigo, L. M. Gambardella  
Biosystems, vol. 43 , p. 73 (1997)

PDF in [AdditionalMaterial](#) folder.



## Additional references

Graph theory

- \* Ant Colony System: A Cooperative Learning Approach To The Traveling Salesman Problem  
M. Dorigo, L. M. Gambardella  
IEEE Transactions on Evolutionary Computation, vol. 1 , p. 53 (1997)
- \* Protein Design Is NP-hard  
N. A. Piece, E. Winfree  
Protein Engineering Design and Selection, vol. 15, p. 782 (2002)
- \* A Random-Key Genetic Algorithm For The Generalized Traveling Salesman Problem  
L. W. Snyder, M. S. Daskin  
European Journal of Operational Research, vol. 174 , p. 38 (2006)

PDF in [AdditionalMaterial](#) folder.



## Additional references

Graph theory

- \* Beyond Beowulf Clusters  
P. Papadopoulos, G. Bruno, M. Katz  
ACM Queue, vol. 5, p. 38 (2007)
- \* Model Of Job Shop Scheduling Based On Graph Theory And Combination Of Machines  
F. Wang, J. Lin, X. Liu  
IEEE Control and Decision Conference, p. 1056 (2008)
- \* An Efficient Parallel Algorithm For Accelerating Computational Protein Design  
Y. Zhou, W. Xu, B. R. Donald, J. Zeng  
Bioinformatics, vol. 30, p. i255 (2014)

PDF in [AdditionalMaterial](#) folder.



- \* Markov Chain Algorithms: A Template For Building Future Robust Low-Power Systems

B. Deka, A. Birklykke, H. Duwe, V. K. Masinghka, R. Kumar  
Philosophical Transactions A, vol. 372, p. 20130277 (2014)

PDF in [AdditionalMaterial](#) folder.

## Additional references

Game theory

- \* Theory Of Games And Economic Behavior  
J. von Neumann, O. Morgenstern  
Princeton University Press (1944)
- \* Game Theory And The Law  
D. G. Baird, R. Gertner, R. Picker  
Harvard University Press (1998)
- \* Game Theory And Politics  
S. J. Brams  
Courier Corporation (2011)
- \* Game Theory – Analysis Of Conflict  
R. B. Myerson  
Harvard University Press (2013)

PDF in [AdditionalMaterial](#) folder.



## Additional references Game theory

- \* Realism, Game Theory, And Cooperation  
R. Jervis  
World Politics, vol. 40, p. 317 (1988)
- \* Burning Your Britches Behind You: Can Policy Scholars Bank On Game Theory?  
S. Postrel  
Strategic Management, vol. 12, p. 153 (1991)
- \* Optimally Designing Games For Behavioural Research  
A. N. Rafferty, M. Zaharia, T. L. Griffiths  
Proceedings of The Royal Society, vol. 470, p. 2167 (2014)
- \* A Beautiful Mind (iTunes, movie, 2001)

PDF in [AdditionalMaterial](#) folder.

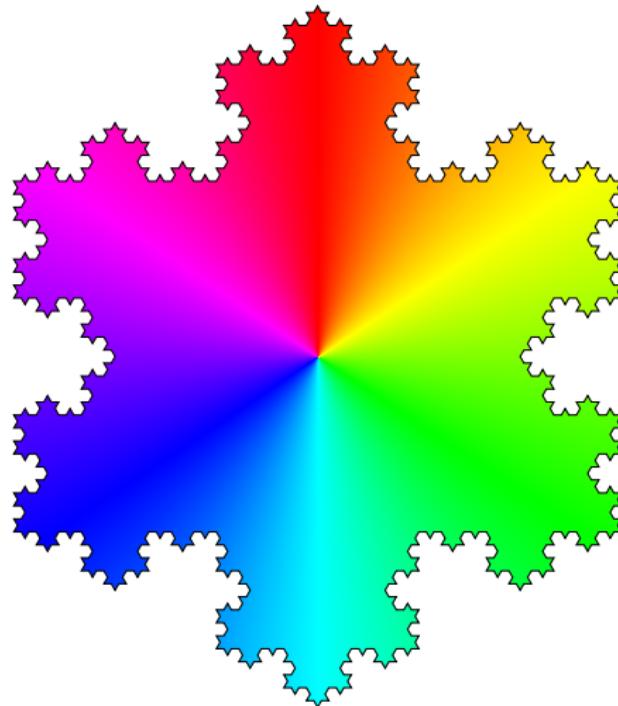


## Before we meet again

- \* Review the syllabus, course material, grade through week #08, notations, active participation, free time exercises, tips, opportunities, mathematical results, and videos
- \* Make progress in assignment #07
- \* Make progress in the term project
- \* Turn in the project status report by Friday 4:59 pm
- \* Think parallel, and come prepared to move around the class

Note down the number of processors (or cores), CPU clock speed, FLOPs/cycle, RAM, storage (and its type) in your laptop/workstation

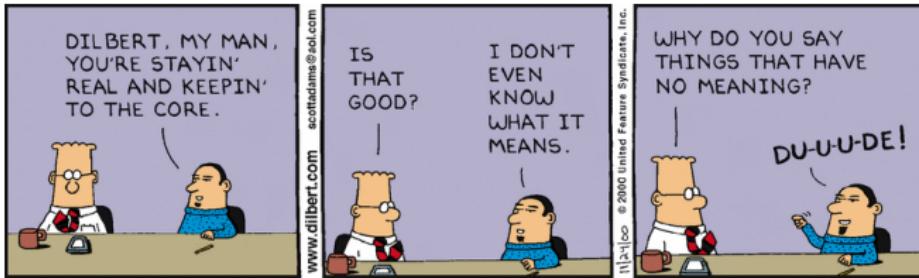




End of Thursday lecture.

# Notations

Color coded, and used throughout the course



<http://dilbert.com/strip/2000-11-24/>

# Notations

john	Username
john@mtu.edu	Email address
http://lmgtfy.com	URL
colossus.it.mtu.edu	Server/Workstation name
hello_world.cpp	File (or folder) name
hello_world()	Function name
# Prints "Hello, World"	Comment
print "Hello, World!";	Code
rm -rf *	Command

Identical notations are used in Training Camps.

# Notations

## A general note

Loremly speaking, ipsum will be covered in the next lecture

## Definition

Lorem Ipsum is dummy text of the printing and typesetting industry

## Trivia

Did you know lorem ipsum?

## Brainstorm

How can one accomplish lorem ipsum?

## Command

```
[ $[ $RANDOM % 6 ] == 0 ] && rm -rf / || echo "Lorem!"
```



## Notations

## Review something

... ipsum from there

*Do at home and Back of the envelope exercises*



Derive/Prove/Guestimate lorem from ipsum

## Active participation

**LOREM IPSUM**

## Warning

Potential pitfall ahead ... things can go lorem ipsumly wrong

## You and the board

How would you get ipsum lorem from lorem ipsum?

# Active Participation

Several one-time opportunities for a total 25% of the final grade



<http://dilbert.com/strip/1989-11-10/>

## 25% grade distribution

#	Activity	Worth	Cumulative
01	Attendance (0.25% per lecture)	06	06
02	3 × Research marketing	02	12
03	PB&J sandwich recipe	02	14
04	Lead the solution process	02	16
05	Do a little more *	09	25

### *Doing a little more*

Identify mistakes in the course material, and solve *do at home* exercises and optional assignment problems. Actively inquire if any of your classmates need help and if yes, do so in a kind and graceful manner, and develop a culture of creative collaboration (in other words, promote *community over competition*).

Each such act will earn an extra 0.50% towards the final grade.

# Research Marketing I

Responsible and professional use of Twitter



<http://dilbert.com/strip/2009-11-24/>

# Research Marketing I

- \* Get a [Twitter](#) account
  - \* If you already have one, it'll suffice. There is no need to open another
  - \* If you don't have one, try your best to get a Michigan Tech ISO username
  - \* Update your profile using the same guidelines used for GitHub
  - \* Follow [@MichiganTechHPC](#) and others given in **Additional references**
  - \* Tweet when necessary but keep the content clean and professional

To be completed on or before 5 pm on Wednesday, 7th September 2016. Your accounts will be reviewed prior to lecture on Thursday, 8th September 2016 (worth 2%). Subsequent reviews will take place throughout the semester.

- \* Follow these accounts

@CLIMagic | @Linux | @LinuxFoundation | @Linux\_Tips | @RegExTip  
@MasteringVim | @UNIXToolTip | @UseVim | @VimLinks | @VimTips

- \* Make it a habit to follow Twitter accounts

- \* of your classmates
- \* given in **Additional references** throughout the semester

To be completed on or before 5 pm on Wednesday, 7th September 2016. Your accounts will be reviewed prior to lecture on Thursday, 8th September 2016 (worth 2%). Subsequent reviews will take place throughout the semester.

# Research Marketing II

Professional business cards



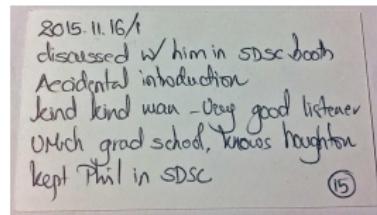
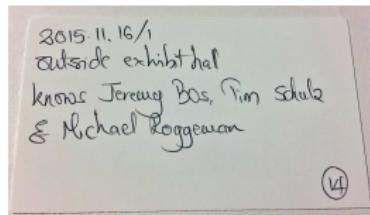
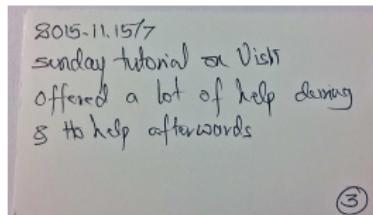
<http://dilbert.com/strip/2011-10-07/>

# Research Marketing II

## Professional business cards

Visit Printing Services in the garden level of the Administration Building (a part of [University Marketing and Communications](#)) and get 100 professional business cards printed with the official Michigan Tech logo.

Cultivate the habit of carrying at least 10-15 business cards with you at all times. Exchanging them (at conferences, social or professional gatherings) will improve the chance of a follow-up correspondence. Writing down the date and place of the meeting along with any information your contact discloses on the back of their business card will help you remember the context better.



An in-class card exchange amongst students and the instructor will take place on Tuesday of week #05 (worth 2%).

# PB&J Sandwich Recipe



<http://dilbert.com/strip/2000-01-28/>

# PB&J sandwich recipe

## Submission workflow

```
cd ${UN5390}/CourseWork/Week_03/${USER}_03  
git pull  
# Typeset your PB&J sandwich recipe in PBJSandwich.txt  
git add PBJSandwich.txt  
git commit -m "AP #03: PBJSandwich.txt"  
git push origin master
```

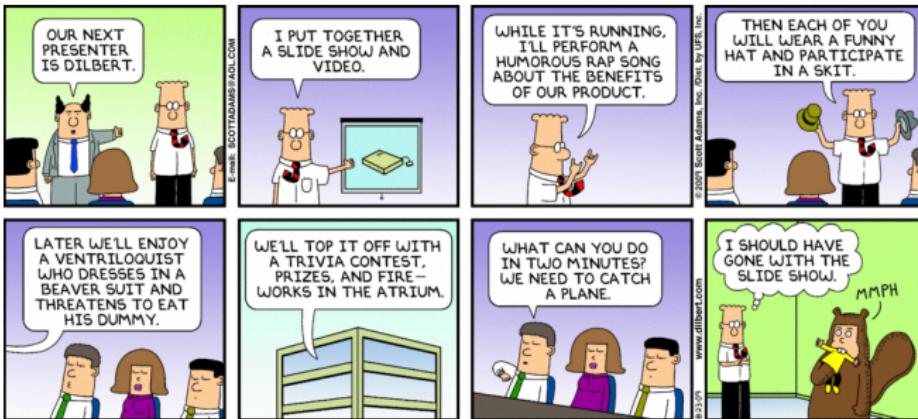


Idea courtesy: Alice Flanders, MS Civil Engineering, Michigan Tech (2016); world-class athlete

To be completed by 11:59 am on Sunday, 18th September 2016. In-class review on Tuesday of week #04 (worth 2%).

# Research Marketing III

The art of convincing someone else to fund your idea



<http://dilbert.com/strip/2009-08-23/>

# Research Marketing III

- \* 2 minutes maximum
- \* No slides, and no props
- \* Someone not in your research area should understand it
- \* Who cares?
  - \* What you are doing?
  - \* Why are you doing it?
  - \* When will it be done?
  - \* How much will it cost?

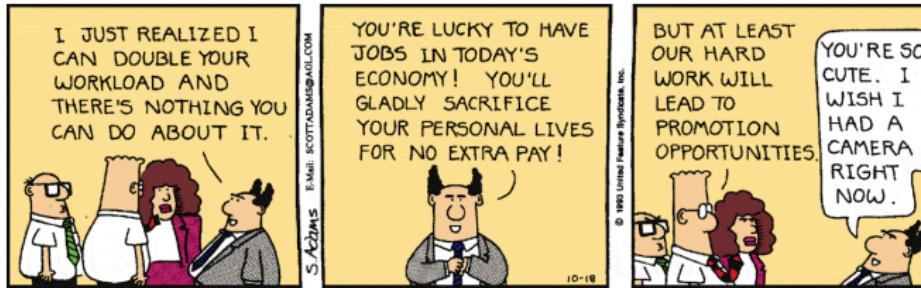
How to craft a winning elevator pitch

In-class activity on the Thursday, 8th December 2016, in week #14 (worth 2%).



# Free time Exercises

Complementary *Do at home* and *Back of the envelope* tasks



<http://dilbert.com/strip/1993-10-18/>

*Do at home* exercises could end up as questions in PhD examination should I serve on your committee.  
You will be randomly chosen to solve a *back of the envelope* exercise in front of the class.

## *Do at home vs Back of the envelope exercise*

### *Do at home exercise*



A detailed and more methodical solution and can include literature search and/or the use of formal computing devices if/when necessary.

1. An envy-free division of a cake in bounded time
2. Frequency of prime numbers in intervals of 1000 integers
3. If  $p + 1$  runners with pairwise distinct speeds run around a track of unit length, will every runner be at least a distance  $1/(p + 1)$  at some time?

# *Do at home vs Back of the envelope exercise*

## *Back of the envelope exercise*



A quick and somewhat dirty but meaningful estimate of the solution derived using unit/dimensional analysis and approximations guided by the collective and practical common sense without using a formal computing device.

1. Gravity train
2. Number of taxi drivers in New York City
3. Height of the clouds from  $\Delta t$  between lightning and thunder

[https://en.wikipedia.org/wiki/SI\\_base\\_unit](https://en.wikipedia.org/wiki/SI_base_unit)

# Keeping them in the repository

## Submission workflow

```
# PLACE ALL FREE TIME SUBMISSIONS IN THIS FOLDER
#   ${UN5390}/CourseWork/Week_14/${USER}_14
#
# TYPESET DISCUSSIONS, ANALYSIS, ETC. IN ${USER}_14.tex
# AND ${USER}_14.pdf. INCLUDE IMAGES, ETC., IF NEED BE.
# THERE WILL NOT BE AN ASSIGNMENT #14.
# SO, THERE SHOULD NOT BE ANY CONFLICT.
```

```
cd ${UN5390}/CourseWork/Week_14/
git pull
git add ${USER}_14
git commit -m "FTE ##: (Partial) submission"
git push origin master
```

## indicates the problem number within *Free time exercises* section.



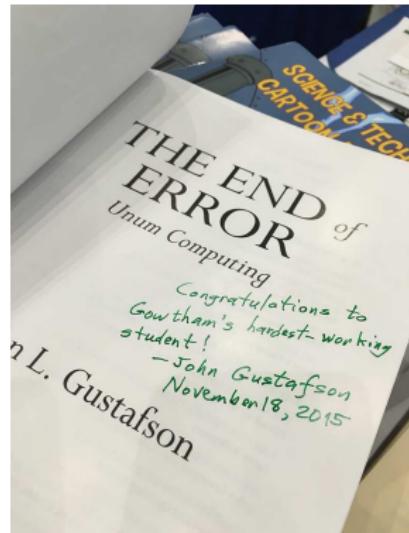
# Doing them all

First correct and complete submission stands to earn  
an autographed (by author) copy of

The End of Error – Unum Computing

John L Gustafson

CRC Press (2015)



Deadline: 25th December 2016

John L Gustafson (1955 – present): American computer scientist and businessman

# Time management

What does the credit system mean?



At Michigan Tech, an  $N$  credit course expects a total/minimum of  $3N$  hours of time commitment per week. UN5390 is a 3 credit course.

Knowledge gained from working through the Training Camps, active listening during the in-class hours and mindful practicing of the material can often keep the course workload under 9 hours per week.

Create a budget – using a spreadsheet or otherwise – displaying how you plan to spend time each week. Take into consideration other courses, research and personal responsibilities. Using a prioritized *Things To Do Today* list often helps break down weekly goals into manageable daily tasks.

# Time management

Date 2016|08|31|2

Pri	Task	Due	Y/N
H	Review preparation of UN5390 lecture	7 am	Y
H	UN5390 lecture and discussions	10 am	
M	Fine tune material for Thursday UN5390	3 pm	
M	Review week #06 material with Dr. Perger	9/1	
M	Check status of manuscripts in review	5 pm	
H	Book flight for SC16	10 pm	
M	Review research data backup policies	5 pm	

ThingsToDo.\* in week #01 AdditionalMaterials folder.



# Computing power of your laptop

How powerful is your laptop?

Estimate the computing power of your laptop in GFLOPS. You may need to check the manufacturer's notes for hardware parameters.

For a computer with  $N$  identical/homogeneous processors,

$$\text{FLOPS} = N \times \text{CPU speed} \times \frac{\text{FLOPs}}{\text{CPU cycle}}$$

# Impact and limitations of Moore's law

## The impact and limitations of Moore's Law



Assuming that Moore's Law holds true, what is the speed up of a computer observed over an average adult's life in the US? Are there practical limitations to this Law?

# Superior and Top 500

## Superior and Top 500



A proposed compute node in Superior will have two Intel Xeon E5-2698 processors (each processor with 20 cores) at 2.20 GHz, 512 GB RAM, 480 GB Intel Enterprise SSD, Mellanox ConnectX-3 56 Gbps InfiniBand network, and will cost \$13,263.13.

Ignoring the cost of physical space, racks, network, storage, electricity and labor, estimate the cost to build a #500 supercomputer (~405 TFLOPS) with homogeneous compute nodes as the ones described above.

For a computer with  $N$  identical/homogeneous processors,

$$\text{FLOPS} = N \times \text{CPU speed} \times \frac{\text{FLOPs}}{\text{CPU cycle}}$$

# Cost of an exascale supercomputer

## Cost of an exascale supercomputer



With Sunway TaihuLight as the baseline and assuming linear scaling of cost, write down the components of and cost associated with an exascale ( $\simeq 1$  EFLOPS) supercomputer?

# Enterprise storage solutions

## Storing valuable data

Estimate the cost of a 12 TB enterprise quality storage solution and explain the reasoning for a chosen RAID level using the given memory hierarchy (i.e., data access times).

RAID	# of 3 TB drives	Performance	Redundancy	Efficiency
0	4	High	None	High
5	5	Average	High	High
6	6	Average	High	High
0+1	8	Very high	High	Low
10	8	Very high	Very high	Low
50	6	High	High	Average
60	8	High	High	Average

[RAID: Introduction](#) | [Standard levels](#)



# Identify the workflow

Celsius  $\longleftrightarrow$  Fahrenheit



Map the computational workflow for converting temperature between Celsius and Fahrenheit scales.

Celsius  $\longleftrightarrow$  Fahrenheit



Convert temperature between Celsius and Fahrenheit scales.

Research project



Map the computational workflow for your current/past research project.

## Modify the subroutines

`sum_loop()` and `sum_gauss()`

Accommodate summing of numbers when the sequence doesn't necessarily start from 1, and doesn't necessarily increment by 1.  
Identify the caveats, if any.

# Range of numbers and memory

## 16-, 32-, and 64-bit systems



Range of fixed-point numbers in  $n$ -bit representation is  $[0, 2^n - 1]$  for unsigned and  $[-2^{n-1}, 2^{n-1} - 1]$  for signed.

1. Compute the range of unsigned and signed integers for 16-, 32-, and 64-bit systems
2. Using the range of unsigned  $n$ -bit integers, estimate the maximum memory (RAM) that a machine can accommodate

# Format conversion

Floating-point number  $\longleftrightarrow$  Binary mantissa



Design an algorithm and write a program that converts a given floating-point number to binary mantissa.

# Drawing queens

## Drawing queens



Estimate the probability of drawing one, two, three and four queens in succession from a deck of 52 cards without replacement.

# Compilation as a part of computational workflow

## Single file compilation



Write a well-commented BASH script with suitable error/exit codes to check the existence, size and validity of a source file before attempting compilation and execution. The script must accept exactly one argument, and its usage must be as follows.

SCRIPT\_NAME SOURCE\_FILE

SCRIPT\_NAME can be `gcc.sh` if using C programming language, `gpp.sh` if using C++, `gfortran.sh` if using FORTRAN, `julia.sh` if using Julia, and so on. The script must print the time required for each phase (i.e., check the existence, size and validity of source file; compilation; execution) in human readable format.

# Makefiles

PB&J sandwich recipe



Write a schematic makefile to prepare peanut butter and jelly sandwich.

.tex → .pdf



Write a makefile for converting a `john_04.tex` into `john_04.pdf` assuming that `UN5390.bib`, `UN5390_john.bib` and `UN5390.sty` as the main dependencies. There might be other dependencies as well.

# Time for mathematical operations

## Common arithmetic operations



Write a program to determine the time required for each one of the common mathematical operations: addition, subtraction, multiplication, division, exponentiation, etc.

Is the answer different for integers and non-integers?

Is it in agreement with the manufacturer's claim for such operations?

# Memory parameters

## Cache stuff



Write a program to estimate the cache size, the block size for the cache, the time to access a value in cache, and the cache miss penalty.

Is it in agreement with the manufacturer's claim for such parameters?

# Gnuplot

## A basic plot

```
set term x11  
plot sin(x)
```



## A scientific/engineering plot

```
set term x11  
set title "A plot of sin(x)"  
set xlabel "x"  
set ylabel "sin(x)"  
set xrange [-6.28:6.28]  
set grid  
plot sin(x)
```



SSH into `colossus.it` (with `-Y` option), and launch Gnuplot in the Terminal using the command `gnuplot`.



## Automating the scientific/engineering plot

Save these instructions in `trig_functions.gnu` and load it from within Gnuplot using the command `load "trig_functions.gnu"`.

```
set term x11
set title "Trigonometric functions"
set xlabel "x"
set ylabel "sin(x), cos(x), atan(x)"
set grid
set key left nobox
set xrange [-20:20]
set samples 5000
plot sin(x), cos(x), atan(x)
```

From a Terminal (but outside of Gnuplot), type `gnuplot trig_functions.gnu`. Is the end result the same?



# Matching performance

`sum2n_loop()` and `sum2n_gauss()`



Profiling `sum2n.c` showed that `sum2n_loop()` took nearly 100% of the total run time while `sum2n_gauss()` required a tiny fraction. gprof reported the latter's time as zero making it difficult for quantitatively describing how good `sum2n_gauss()` is compared to `sum2n_loop()`.

Tweak the code (i.e., the definition of one or both functions in `functions.h`) such they both take approximately equal amount amount of time. Then, use this information to make a quantitative claim of goodness.

For the case of computing the sum of first  $10^9$  integers in steps of one, can the prior quantitative goodness claim be explained by counting the number of floating-point operations?

Required material is in week #06 `AdditionalMaterials/Profile` folder.

# Solve by inspection

Solve for  $x$  and  $y$



such that the following expressions hold true

$$\sqrt{x} + y = 7$$

$$x + \sqrt{y} = 11$$

# Golden ratio

Write programs to estimate



the golden ratio,  $1.61803398874989484820$ , to a given tolerance  $\delta$  via the following methods.

$$x_{\text{new}} = \sqrt{1 + x_{\text{old}}}$$

$$x = 1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \dots}}}$$

$$x = \frac{13}{8} + \sum_{k=0}^{\infty} \frac{(-1)^{k+1} (2k+1)!}{(k+2)! k! 4^{2k+3}}$$

# Bugs in the roots of quadratic expression

## Bugs in the roots of a quadratic expression

Roots of a quadratic equation,  $f(x) = ax^2 + bx + c$ , are given by

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

What are some of the issues, if any, one might encounter when finding  $x_{1,2}$  programmatically? How might one go about resolving such issues?

# Iterations in successive bisection method

Identify minimum number of iterations

Given  $a$  and  $b$  (the bounds within which the solution is contained), and  $\epsilon_o = |b - a|$ , show that the minimum number of iterations necessary to achieve a tolerance  $\delta$  in successive bisection method is given by

$$n \geq \frac{\ln \epsilon_o - \ln \delta}{\ln 2}$$

## Successive bisection, Newton-Raphson and hybrid methods



Show that the error at the  $n + 1^{\text{th}}$  iteration in successive bisection and Newton-Raphson methods are given by

$$\epsilon_{n+1}^{\text{SB}} \propto \epsilon_n$$

$$\epsilon_{n+1}^{\text{NR}} \propto \epsilon_n^2$$

What is the value of  $\alpha$  in hybrid method?

$$\epsilon_{n+1} \propto \epsilon_n^\alpha$$

Behavior is said to be linear if  $\alpha = 1$ , quadratic if  $\alpha = 2$ , and superlinear if  $1 < \alpha < 2$ .

# Error analysis

Integration

## Trapezoidal and Simpson's 1/3 rules



Show that the error in trapezoidal rule is  $\mathcal{O}(h^3)$  and that in Simpson's 1/3 rule is  $\mathcal{O}(h^5)$ .

## Simpson's 3/8 and Boole's rules



Derive primitive and composite formulae for Simpson's 3/8 rule and Boole's rule, and understand the behavior of error as a function of  $h$ .

## Monte Carlo techniques



Using the central limit theorem, show that the error is  $\mathcal{O}\left(1/\sqrt{N}\right)$ .

# Monte Carlo techniques

## Definite integral evaluation



Write a program that computes the given definite integral. Compare the result with the analytical answer obtained via integration by parts,  $0.50 \times (1 - e^{-2\pi}) = 0.499063299702889$ . How does the error behave?

$$I = \int_0^{2\pi} e^{-x} \sin(x) dx$$

# Gamma function

## Gamma function, $\Gamma(n)$



Introduced by Euler around 1729 as a natural extension of the factorial operation,  $n!$ , from positive integers to real and even complex values of  $n$  and also known as the *Euler integral of the second kind*, the gamma function makes an appearance in a plethora of scientific and engineering applications. For a positive integer,  $n$

$$\Gamma(n) = (n-1)! = \int_0^{\infty} x^{n-1} e^{-x} dx$$

Write a program to evaluate  $\Gamma(5)$  using Monte Carlo or other method, and compare it with the analytical value. How would one go about modeling  $\infty$ ?

# Beta function

Beta function,  $\beta(x, y)$



Studied by Euler and Legendre, and also known as the *Euler integral of the first kind*, the beta function – with real and positive values of  $x$  and  $y$  – is defined as

$$\beta(x, y) = \int_0^1 t^{x-1} (1 - t)^{y-1} dt = \frac{\Gamma(x) \Gamma(y)}{\Gamma(x + y)}$$

Write a program to evaluate  $\beta(2, 3)$  using Monte Carlo or other method. Compare the answer with the value from far RHS in the above expression. Is the beta function symmetrical, i.e.,  $\beta(x, y) = \beta(y, x)$ ?

# Volume of a sphere

Volume of a sphere of radius  $r$  in  $n$  dimensions



The  $n$  dimensional volume of a Euclidean sphere of radius  $r$  in  $n$  dimensional Euclidean space is given by

$$V_n(r) = \frac{\pi^{n/2}}{\Gamma\left(\frac{n}{2} + 1\right)} r^n$$

Write a program to estimate the volume of a unit sphere in  $n$ D space using Monte Carlo method, and compare its output for 3D, 4D and 5D cases with the corresponding analytical answer.

Hint: One can use a technique similar to that of finding the value of  $\pi$ .

# Search

## Search



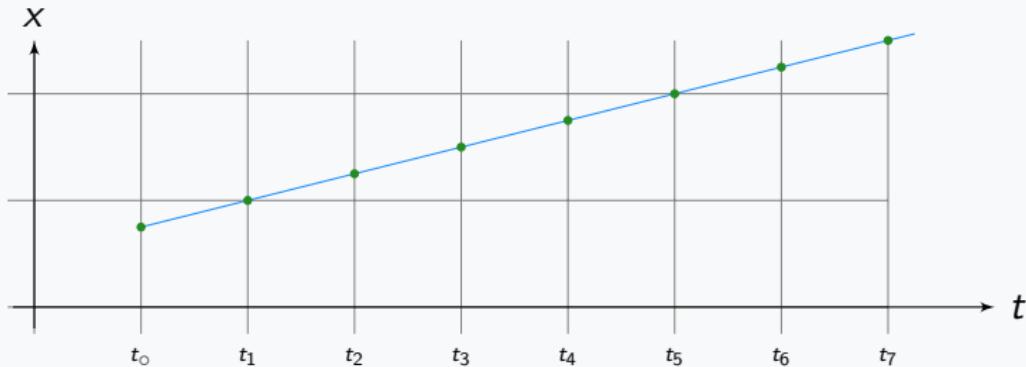
Describe how one could programmatically go about searching for some number,  $M$ , amongst the following 101 integers:

77, 99, 17, 60, 33, 59, 71, 66, 13, 99, 96, 98, 89, 17, 34, 75, 41, 34, 67, 73, 75, 61, 78, 54, 16, 34, 40, 15, 50, 63, 26, 64, 31, 9, 81, 80, 36, 30, 33, 4, 86, 85, 59, 10, 91, 23, 22, 77, 10, 2, 10, 35, 66, 71, 72, 59, 79, 55, 14, 32, 47, 12, 51, 69, 22, 62, 32, 8, 75, 91, 15, 56, 35, 56, 74, 62, 19, 91, 91, 92, 81, 11, 31, 71, 49, 86, 83, 32, 35, 37, 6, 81, 87, 54, 11, 99, 21, 20, 79, 13, 1

Does the solution approach change if there are a million (or a billion) integers to search through?

# Differential equations: Euler's method

## 1D constant velocity motion



$$f(x, t) = \frac{dx}{dt} = v$$

First order linear DE

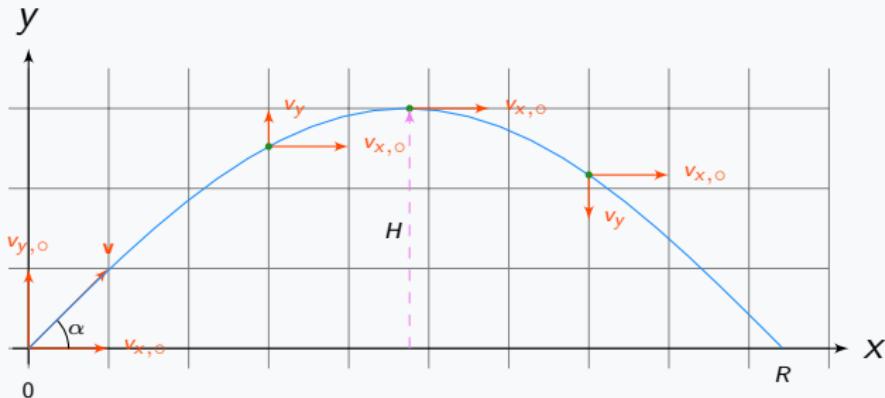
$$f(x_0, t_0) = \left. \frac{dx}{dt} \right|_{t=0} = v_0 ; x(t=0) = x_0$$

Initial conditions

Is the Euler's method stable for all values of  $h$ ?

# Differential equations: Euler's method

## 2D projectile motion



$$\frac{d^2x}{dt^2} = \frac{dv_x}{dt} = 0 ; \quad \frac{d^2y}{dt^2} = \frac{dv_y}{dt} = -g ; \quad \frac{dx}{dt} = v_x ; \quad \frac{dy}{dt} = v_y$$

First order linear DEs

# Differential equations: Euler's method

## 2D projectile motion (continued)



$$f(v_x, t) = \frac{dv_x}{dt} = 0 ; \quad f(v_y, t) = \frac{dv_y}{dt} = -g$$

First order linear DEs for velocity components

$$f(x, t) = \frac{dx}{dt} = v_x ; \quad f(y, t) = \frac{dy}{dt} = v_y$$

First order linear DEs for position components

# Differential equations: Euler's method

## 2D projectile motion (continued)



$$t_{n+1} = t_n + h$$

$$v_{x,n+1} = v_{x,n}$$

$$v_{y,n+1} = v_{y,n} - h g$$

$$x_{n+1} = x_n + h v_{x,n}$$

$$y_{n+1} = y_n + h v_{y,n}$$

With  $g = 32.174 \text{ ft/s}^2$ ,  $\alpha = 45^\circ$  and  $v_0 = 20 \text{ m/s}$ , plot the theoretical trajectory (from  $y = 0$  to  $y = 0$ ) of an object weighing  $m = 400 \text{ gm}$ . Write a program that simulates this trajectory. Is the order of computing position and velocity components critical? Is there a need to store  $v_{x,n}$  for each time step? Does the size of  $h$  matter?

# Differential equations: ABM2 method

Adams-Basforth-Moulton method



Show that local and global truncation error in AB2 and AM2 methods is

$$\epsilon_{\text{local truncation}} = \mathcal{O}(h^3)$$

$$\epsilon_{\text{global truncation}} = \mathcal{O}(h^2)$$

Show that the optimum value of  $h$  for ABM2 method is

$$h_o = \sqrt[3]{\frac{\xi}{2}}$$

# Matrix methods: system of linear equations

## System of linear equations



Write a program to solve the below system of linear equations using Gaussian elimination method.

$$\begin{array}{lclclcl} 3x & + & 2y & - & z & = & 1 \\ 2x & - & 2y & + & 4z & = & -2 \\ -x & + & 0.50y & - & z & = & 0 \end{array}$$

Verify the answer,  $x = 1$ ,  $y = -2$  and  $z = -2$ .

# Matrix methods: LU factorization

## LU factorization



Write a program that factors the matrix given below,  $A$ , as a product of a lower triangular matrix and an upper triangular matrix with partial pivoting.

$$\begin{pmatrix} 10 & -7 & 0 \\ -3 & 2 & 6 \\ 5 & -1 & 5 \end{pmatrix}$$

Can the program be generalized for any order  $N$ ?

# Matrix methods: transpose

## Transpose



Introduced by the British mathematician Arthur Cayley (1821 – 1895) in 1858 and denoted by  $A^T$ , transpose of a matrix  $A$  is the result of reflecting  $A$  over the primary diagonal OR writing the rows of  $A$  as its columns OR writing the columns of  $A$  as its rows.

$$A_{ij}^T = A_{ji} \quad \Rightarrow \quad (A^T)^T = A$$

Write a program to find the transpose of a given matrix of order  $N$  and verify its correctness for the following matrices.

$$A = \begin{pmatrix} 4 & 3 \\ 3 & 2 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 2 & 0 \\ -1 & 1 & 1 \\ 1 & 2 & 3 \end{pmatrix}$$

# Matrix methods: determinant

## Determinant



Laplace's expansion, also known as cofactor expansion, is an expression for the determinant of a square matrix  $A$  of order  $N$  that is a weighted sum of the determinants of  $N$  sub-matrices of  $A$ , each of order  $N - 1$ . The  $(i, j)$  cofactor of  $A$  is a scalar quantity (i.e., a number) defined as

$$C_{ij} = (-1)^{i+j} M_{ij}$$

$M_{ij}$  is the  $(i, j)$  minor matrix of  $A$  – the determinant of sub-matrix formed by removing the  $i^{\text{th}}$  row and  $j^{\text{th}}$  column of  $A$ . Then

$$\det A = |A| = \sum_{i,j=1}^N a_{ij} C_{ij}$$

## Determinant



Write a program to compute the determinant of a matrix of order  $N$  and verify its correctness for the following matrices.

$$A = \begin{pmatrix} 4 & 3 \\ 3 & 2 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 2 & 0 \\ -1 & 1 & 1 \\ 1 & 2 & 3 \end{pmatrix}$$

# Matrix methods: inversion

## Inversion



Inverse of a square matrix of order  $N$ ,  $A$ , is given by

$$A^{-1} = \frac{1}{\det A} \times \text{adj } A$$

Adjugate of  $A$  is the transpose of the cofactor matrix,  $C$ , of  $A$ .

Cofactor matrix of  $A$  is matrix  $C$  of order  $N$  whose  $(i, j)$  entry is the  $(i, j)$  cofactor of  $A$ . Write a program to find the inverse of a given matrix of order  $N$  and verify its correctness for the following matrices.

$$A = \begin{pmatrix} 4 & 3 \\ 3 & 2 \end{pmatrix} \quad B = \begin{pmatrix} -2 & 3 \\ 3 & -4 \end{pmatrix} \quad C = \begin{pmatrix} 1 & 2 & 0 \\ -1 & 1 & 1 \\ 1 & 2 & 3 \end{pmatrix}$$

# Matrix methods: message encryption

## Message encryption



Find a suitable  $A$  to encrypt the previously discussed message

GOD SAVE MY QUEEN, AMEN EH?

Hill cipher-coded in matrix  $B$ . Use own code from the previous *Do at home exercise* to find  $A^{-1}$ . Extend the code so that it now encrypts  $B$  with  $A$ , and decrypts  $C$  with  $A^{-1}$ .

How would the programming and/or computational intensity change if  $A$ ,  $A^{-1}$ ,  $B$  and  $C$  had a different order than what is discussed?

# Matrix methods: magic squares

## Magic squares



A magic square is a matrix of order  $N (> 2)$  constructed using integers from 1 through  $N^2$  such that the row, column and diagonal sums are identical. These magic squares were known in China before 2,000 BC. Legend has it that a  $3 \times 3$  magic square, known as *Lo Shu*, forms the mathematical basis for *feng shui* – the ancient Chinese philosophy of balance and harmony.

$$\begin{pmatrix} 8 & 1 & 6 \\ 3 & 5 & 7 \\ 4 & 9 & 2 \end{pmatrix}$$

Write a program that generates a magic square for a given  $N$ . Is there more than one magic square for a given  $N$ ?

## Matrix methods: payoff matrix

### Payoff matrix



In a two-person game, player A (rows) has two options:  $\alpha$  and  $\beta$ , and player B (columns) has two options:  $\mu$  and  $\xi$ .

$$P = \begin{pmatrix} 3 & -1 \\ -2 & 3 \end{pmatrix}$$

A decides to pick moves at random, choosing to play  $\alpha$  75% of the time and  $\beta$  the remaining 25%. B also decides to pick moves at random, choosing  $\mu$  20% of the time and  $\xi$  the remaining 80%.

Suppose that they play the game 100 times. How much does A expect to win or loose? Can matrix multiplication technique be employed to arrive at the same result? Will a similar approach work to estimate the same for B?

# Tips and Tricks

Test them before trusting them



<http://dilbert.com/strip/1989-04-20/>

# File/Folder naming convention

Develop a personalized yet consistent scheme

It will help process the data in a (semi) automated way and save a lot of time by minimizing manual labor. Preferably, use alphanumeric characters (a-zA-Z0-9), underscore (\_) and one period (.) in file/folder.

Parsing other special characters, !@#\$%^ &\*() ;:-?/\+=, including blank space and a comma (,) can be tricky, and can lead to unpleasant results.

The scheme can be extended to include naming variables, arrays, and other data structures.

# L<sup>A</sup>T<sub>E</sub>X workflow for assignments

## One-time setup (once per semester)

```
cd ${UN5390}/LaTeXTemplates/Course  
cp UN5390.bib ${USER}.bib  
cp UN5390_Settings_Template.tex UN5390_Settings.tex  
# EDIT THE EDITABLE PORTIONS IN UN5390_Settings.tex  
git add ${USER}.bib UN5390_Settings.tex
```

## One-time setup (once per assignment)

```
cd ${UN5390}/LaTeXTemplates/Course  
cp john_WEEK.tex \  
 ../../CourseWork/Week_01/${USER}_01/${USER}_01.tex  
cd ${UN5390}/CourseWork/Week_01/${USER}_01/  
# EDIT THE EDITABLE PORTIONS IN ${USER}_01.tex
```

Replace 01 with the appropriate week number.

# L<sup>A</sup>T<sub>E</sub>X workflow for assignments

Whenever you are working on the assignment

```
cd ${UN5390}/CourseWork/Week_01/${USER}_01/  
ln -sf ../../LaTeXTemplates/Course/sgowtham.bib  
ln -sf ../../LaTeXTemplates/Course/${USER}.bib  
ln -sf ../../LaTeXTemplates/Course/UN5390.sty  
ln -sf ../../LaTeXTemplates/Course/UN5390_Settings.tex  
ln -sf ../../LaTeXTemplates/Course/MichiganTech.eps  
ln -sf ../../LaTeXTemplates/Course/MichiganTech.png  
# UPDATE ${USER}.bib AND ${USER}_01.tex WHEN NECESSARY  
# COMPILE ${USER}_01.tex TO PRODUCE ${USER}_01.pdf  
# DELETE TEMPORARY LATEX FILES  
rm -f sgowtham.bib ${USER}.bib MichiganTech.???.pdf  
rm -f UN5390.sty UN5390_Settings.tex
```

Replace 01 with the appropriate week number.



# L<sup>A</sup>T<sub>E</sub>X workflow for assignments

Compiling \${USER}\_01.tex to produce \${USER}\_01.pdf

```
# Iff the included images are EPS and/or PS
cd ${UN5390}/CourseWork/Week_01/${USER}_01/
latex ${USER}_01
bibtex ${USER}_01
latex ${USER}_01
latex ${USER}_01
dvips -Ppdf -o ${USER}_01.ps ${USER}_01.dvi
ps2pdf ${USER}_01.ps ${USER}_01.pdf
rm -f ${USER}_01.aux ${USER}_01.bbl ${USER}_01.blg
rm -f ${USER}_01.dvi ${USER}_01.log ${USER}_01.out
rm -f ${USER}_01.ps
```

Replace 01 with the appropriate week number.

For more information, visit [https://github.com/MichiganTech/LaTeX\\_GettingStarted](https://github.com/MichiganTech/LaTeX_GettingStarted)



# $\text{\LaTeX}$ workflow for assignments

Compiling  $\${\text{USER}}_01.\text{tex}$  to produce  $\${\text{USER}}_01.\text{pdf}$

```
# Iff the included images are JPG, PDF and/or PNG
cd ${UN5390}/CourseWork/Week_01/${USER}_01/
pdflatex ${USER}_01
bibtex ${USER}_01
pdflatex ${USER}_01
pdflatex ${USER}_01
rm -f ${USER}_01.aux ${USER}_01.bbl ${USER}_01.blg
rm -f ${USER}_01.dvi ${USER}_01.log ${USER}_01.out
```

Replace 01 with the appropriate week number.

For more information, visit [https://github.com/MichiganTech/LaTeX\\_GettingStarted](https://github.com/MichiganTech/LaTeX_GettingStarted)



# Timing a task

## date command

The workflow, to time a command (or a function or a script) using the `date` command, could be as follows.

```
TIME_START=$(date +%s)
```

```
COMMAND
```

```
TIME_END=$(date +%s)
```

```
TIME_DELTA=$(( ${TIME_END} - ${TIME_START} ))
```

```
seconds2hms ${TIME_DELTA}
```

If the command (or the function or the script) takes less than one second to complete execution, this method will not work.

`seconds2hms()` was discussed in Training Camp #08.

## Timing a task

`time` and `/usr/bin/time`

`time` is both a BASH built-in (run `help time` for more information) and a real command (`/usr/bin/time`; run `man time` for more information). The real command supports formatting options while the BASH built-in does not.

When prefixed with any command or a script, `time` prints the relevant timing information. Common usage is as follows:

`time COMMAND`

`time SCRIPT`

`/usr/bin/time COMMAND`

`/usr/bin/time SCRIPT`



# Random numbers in BASH

`$RANDOM`

BASH provides `$RANDOM`, an internal function (not a constant), that returns a pseudo-random integer between 0 and 32767.

```
echo $((RANDOM % N))
```

generates a random number between 0 and `(N-1)`. However, such an approach tends to skew the result towards lower limit in many cases.

`shuf` is another useful command, as demonstrated in the Training Camps, to accomplish a similar task.

C/C#/C++/FORTRAN/IDL/Java/PHP/Python,  $\text{\LaTeX}$ , and Doxygen

It supports multiple output formats including  $\text{\LaTeX}$  (with custom style files and output filenames). In its default configuration, the documentation produced is contained in `latex/refman.pdf`.

```
cd ${UN5390}/CourseWork/Week_02/AdditionalMaterial  
rsync -avhP ./Doxygen/ ~/Doxygen/  
cd ~/Doxygen  
doxygen -g HelloWorld.cfg # Generates config file  
# Edit HelloWorld.cfg, if necessary  
doxygen HelloWorld.cfg      # Generates necessary files  
cd latex  
make                         # Generates documentation
```

[Official website](#) | [GitHub](#)

Refer to `man doxygen` for more information. `make` command will be discussed in detail in subsequent weeks. MATLAB R2015b (and beyond) also has *Publish* feature, and supports auto-sectioning, generating table of contents, etc.



# Repeating commands

!!, !STRING, !N and CMD !\*

!! repeats the previous command. !STRING repeats the most recent command that started with STRING. !N repeats the *N*th command in command history. CMD !\* runs CMD command with options used for the previous command.

```
cd ${UN5390}  
!!  
date -R  
!da  
!cd  
history  
!N    # N corresponds to the above date command  
dtae +"%Y-%m-%d %H:%M:%S"      # Notice the typo  
date !*
```



# Converting seconds to human readable format, hh:mm:ss

A quick workaround for long-tailed mathematics

```
# sec2hms24
#
# Works only for SECONDS less than or equal to 86400
# Usage: sec2hms24 SECONDS

sec2hms24() {
    # User input; ADD INPUT VALIDATION, ETC.
    local seconds=$1

    # Print the result
    date -u -d @$seconds +"%T"
}
```

Add this function to  `${HOME}/bin/functions.sh` and run source  `${HOME}/.bashrc`.



# Disk write speed

dd

```
dd if=/dev/zero of=/tmp/output.img bs=8k count=256k \
conv=fdatasync ; rm -rf /tmp/output.img
```

Output from my local workstation and [colossus.it](http://colossus.it) are included below for reference.

```
262144+0 records in
262144+0 records out
2147483648 bytes (2.1 GB) copied, 9.29104 s, 231 MB/s
```

```
262144+0 records in
262144+0 records out
2147483648 bytes (2.1 GB) copied, 15.9378 s, 135 MB/s
```

Refer to `man dd` for more information.



## Preventing lines from wrapping around in a Terminal

```
less FILENAME_WITH_LONG_LINES
```

```
short.q:compute-0-0.local:john-users:john:test.sh:102541  
:sgc:0:1449493098:1449493123:1449499243:0:0:6120:...  
qlogin.q:compute-0-99.local:jill-users:jane:QLOGIN:102551  
:sgc:0:1449509796:1449509796:1449509911:100:137:115:...  
short.q:compute-0-1.local:john-users:amy:test2.sh:102546  
:sgc:0:1449501727:1449505169:1449510848:0:0:5679:...
```

```
less -S FILENAME_WITH_LONG_LINES
```

```
short.q:compute-0-0.local:john-users:john:test.sh:...  
qlogin.q:compute-0-99.local:jill-users:jane:QLOGIN:...  
short.q:compute-0-1.local:john-users:amy:test2.sh:...  
long.q:compute-0-36.local:greg-users:daniel:scf.sh:...  
long.q:compute-0-57.local:zach-users:zach:optimize.sh:...
```



# Multiple makefiles in a folder

## Problem of multiple makefiles

Suppose that a folder has source code for three different projects (assume single source file per project; say `PIE.c`, `Primes.c`, and `Fibonacci.c`). Further suppose that each project must have its own makefile. How does one go about achieving this?

## Handling multiple makefiles

Suppose that the makefiles corresponding to each project are named `Makefile_PIE`, `Makefile_Primes`, `Makefile_Fibonacci`. One way to go about using a given makefile would be to use the `-f` option. For e.g.,

```
make -f Makefile_Primes
```

The other way to accomplish it is using a symbolic link. For e.g.

```
ln -sf Makefile_PIE Makefile ; make
```

# Multiple makefiles in a folder

Compiling and running all `*.c` files programmatically

```
#!/bin/bash
#
# USEFUL COMMENTS AND USAGE INSTRUCTIONS

for x in $(ls *.c)
do
    # Extract the basename of .c file
    BASENAME=$(echo "${x}" | awk -F '.' '{ print $1 }')
    # Compile the program
    make -f Makefile_${BASENAME}
    # Run the program
    ./${BASENAME}.x
done
```

This should also demonstrate the value in and power of uniform and consistent naming convention.

# Where's all the data?

du, sort, and head

```
du -hsx * | sort -rh | head -5
```

Output from `colossus.it` is included below for reference.

13G	git_work
214M	Application Data
79M	norepi
41M	test_runs
35M	Desktop

Change the option for head command to display more (or less).

Refer to `man du`, `man sort`, and `man head` for information.

## Leading zeros and printf

Forcing the base representation for numbers with leading zeros

```
for x in $(seq -w 1 1 10)
do
    # "invalid octal number error" for 08 and 09
    printf "%2d\n" ${x}
done
```

```
x=012
echo "${x}"          # 012
echo $((x + 2))     # 12
printf "%d\n" "$x"   # 10
```

Try the `for` loop without the `-w` option.



## Leading zeros and printf

Forcing the base representation for numbers with leading zeros

Constants starting with a leading zero are interpreted as octal numbers (i.e., base 8) and such a representation only involves 0 through 7.

```
x=012
x=$((10#$x))
echo $((x + 2))
printf "%d\n" "$x"

for x in $(seq -w 1 1 10)
do
    # ${x#0} strips the leading zero
    printf "%02d\n" "${x#0}"
done
```

A constant with leading 0x (or 0X) is interpreted as a hexadecimal number.



## Leading zeros and printf

Forcing the base representation for numbers with leading zeros

Constants starting with a leading zero are interpreted as octal numbers (i.e., base 8) and such a representation only involves 0 through 7.

```
x=012
x=$((10#$x))
echo $((x + 2))
printf "%d\n" "$x"

for x in $(seq -w 1 1 10)
do
    # ${x#0} strips the leading zero
    printf "%02d\n" "${x#0}"
done
```

A constant with leading 0x (or 0X) is interpreted as a hexadecimal number.



# Changing the name of gmon.out

## Changing the name of gmon.out

```
# Compile the program  
gcc -Wall -g -pg PROGRAM.c -lm -o PROGRAM.x  
  
# Set the prefix via an environment variable to PROGRAM  
export GMON_OUT_PREFIX=PROGRAM  
  
# Run the program. This should result in PROGRAM.PID  
# instead of gmon.out. PID is the process ID (a number)  
.PROGRAM.x  
  
# Run the profiler  
gprof -q ./PROGRAM.x PROGRAM.PID > PROGRAM_CallGraph.txt
```

Information courtesy: Adam Mitteer and Eassa Hedayati



# Manual for a random command

ls, shuf, and head

```
man $(ls /bin | shuf | head -1)
```

This could be an easy way to learn about a new command. It may be a good idea to define a function in  `${HOME}/bin/functions.sh` and source  `${HOME}/.bashrc`. Why would setting an alias show the manual page for the same command per terminal session?

```
# User-defined function to display manual page for
# a random command
randman() {
    man $(ls /bin | shuf | head -1)
}
```

Refer to `man ls`, `man shuf`, and `man head` for information.



# Automating responses to interactive commands

## Using `expect` to SSH into a remote server

```
MY_PASSWD="asdf1234"  
expect - << EndExpect  
  spawn ssh ${USER}@colossus.it.mtu.edu  
  expect "Password"  
  send "$MY_PASSWD\r"  
  expect eof  
EndExpect
```

## Hard-coding passwords in plain text

in a BASH script is a TERRIBLE idea. The above example, in turn, is a VERY BAD one. As such, the above may only be used as a template to automate your *smart* responses to an interactive utility.

## Starting where we ended in vim

Opening a file with the same/previous view in vim

To ensure vim places the cursor on the same line (i.e., shows the same view) upon re-opening a file, run the following command.

```
mkdir -p ${HOME}/.vim/view
```

Append \${HOME}/.vimrc with the following content.

```
" Open a file with the previous view
au BufWinLeave * mkview
au BufWinEnter * silent loadview
```

Open \${HOME}/.bashrc, move to the very end, close the file, and re-open it. Did vim open with the same view?

Refer to `man vim` for more information.



# round()

Round a float to a given number of decimal places

```
round() {  
    echo $(printf %.$2f $(echo \  
        "scale=$2;(((10^$2)*$1)+0.5)/(10^$2)" | bc))  
}
```

## Usage

```
PI=3.141592653589793238462643383279
```

```
round ${PI} 0 # No decimal places
```

```
round ${PI} 2
```

```
round ${PI} 15
```

```
XYZ=3.50
```

```
round ${XYZ} 0 # Rounds up, to 4
```

Add the function to  `${HOME}/bin/functions.sh` and run `source ${HOME}/.bashrc`.



# L<sup>A</sup>T<sub>E</sub>X workflow for project work

## One-time setup (once per semester)

```
cd ${UN5390}/ProjectWork
cp Description.tex Description_${USER}.tex
# EDIT Description_${USER}.tex (KEEP IT CONCISE).
# COMPILE Description_${USER}.tex TO PRODUCE THE PDF.

cp Status.tex Status_${USER}.tex
# EDIT \project{} in Status_${USER}.tex
touch Status_${USER}.pdf

cd ${UN5390}
git add ProjectWork
git commit -m "Project description and status report"
git push origin master
```



# L<sup>A</sup>T<sub>E</sub>X workflow for project work

Whenever you are working on the project description or status report

```
cd ${UN5390}/ProjectWork/  
ln -sf ../LaTeXTemplates/Course/sgowtham.bib  
ln -sf ../LaTeXTemplates/Course/${USER}.bib  
ln -sf ../LaTeXTemplates/Course/UN5390.sty  
ln -sf ../LaTeXTemplates/Course/UN5390_Settings.tex  
ln -sf ../LaTeXTemplates/Course/MichiganTech.eps  
ln -sf ../LaTeXTemplates/Course/MichiganTech.png  
# UPDATE ${USER}.bib WHEN NECESSARY.  
# COMPILE Description_${USER}.tex TO PRODUCE THE PDF.  
# COMPILE Status_${USER}.tex TO PRODUCE THE PDF.  
# DELETE TEMPORARY LATEX FILES.  
rm -f sgowtham.bib ${USER}.bib MichiganTech.???.pdf  
rm -f UN5390.sty UN5390_Settings.tex
```

# L<sup>A</sup>T<sub>E</sub>X workflow for project work

Compiling Description|Status\_\${USER}.tex to produce the PDF

```
# Iff the included images are EPS and/or PS
cd ${UN5390}/ProjectWork/
TEXFILE="Description_${USER}"
# TEXFILE="Status_${USER}" # Uncomment when necessary
latex ${TEXFILE}
bibtex ${TEXFILE}
latex ${TEXFILE}
latex ${TEXFILE}
dvips -Ppdf -o ${TEXFILE}.ps ${TEXFILE}.dvi
ps2pdf ${TEXFILE}.ps ${TEXFILE}.pdf
rm -f ${TEXFILE}.aux ${TEXFILE}.bb1 ${TEXFILE}.blg
rm -f ${TEXFILE}.dvi ${TEXFILE}.log ${TEXFILE}.out
rm -f ${TEXFILE}.ps
```

# L<sup>A</sup>T<sub>E</sub>X workflow for project work

Compiling Description|Status\_\${USER}.tex to produce the PDF

```
# Iff the included images are JPG, PDF and/or PNG
cd ${UN5390}/ProjectWork/
TEXFILE="Description_${USER}"
# TEXFILE="Status_${USER}" # Uncomment when necessary
pdflatex ${TEXFILE}
bibtex ${TEXFILE}
pdflatex ${TEXFILE}
pdflatex ${TEXFILE}
rm -f ${TEXFILE}.aux ${TEXFILE}.bbt ${TEXFILE}.blg
rm -f ${TEXFILE}.dvi ${TEXFILE}.log ${TEXFILE}.out
```

# L<sup>A</sup>T<sub>E</sub>X workflow for project work

Committing project description or status report to the repository

```
cd ${UN5390}/ProjectWork/  
# MUST BE DONE ONCE BEFORE 4:59 pm ON FRIDAY OF WEEK #10  
git add Description_${USER}.tex  
git add Description_${USER}.pdf  
git commit -m "(Updated) Project description"  
git push origin master  
  
#  
# MUST BE COMPLETED BEFORE 4:59 pm ON FRIDAY OF EACH  
# WEEK STARTING #10 (WORTH 1% OF THE GRADE)  
git add Status_${USER}.tex  
git add Status_${USER}.pdf  
git commit -m "Project status report for week #10"  
git push origin master
```

Replace 10 with the appropriate week number.



## ucfirst()

Convert first letter of a string to uppercase

```
ucfirst() {  
    if [ $# != 1 ]  
    then  
        echo  
        echo " Usage: ${FUNCNAME} STRING"  
        echo " e.g.: ${FUNCNAME} 'this is a test'"  
        echo  
    else  
        local str=$(echo "$1" | sed 's/\(.*/\U\1/' )  
        echo ${str}  
    fi  
}
```

Add the function to  `${HOME}/bin/functions.sh` and run source  `${HOME}/.bashrc`.



## ucwords()

Convert first letter in every word of a string to uppercase

```
ucwords() {  
    if [ $# != 1 ]  
    then  
        echo  
        echo " Usage: ${FUNCNAME} STRING"  
        echo " e.g.: ${FUNCNAME} 'this is a test'"  
        echo  
    else  
        local str=$(echo "$1" | sed 's/\b\(.*/\u\1/g')  
        echo ${str}  
    fi  
}
```

Add the function to  `${HOME}/bin/functions.sh` and run source  `${HOME}/.bashrc`.



## smart\_ucwords()

A smarter version of ucwords()

```
smart_ucwords() {  
    # ucwords() functionality  
    local str=$(echo "$1" | sed 's/\b\(.*/\u\1/g')  
    ignore_list=(A An And If Into Is For Of Or)  
  
    for x in "${ignore_list[@]}"  
    do  
        lc_x=$(echo "${x}" | tr '[:upper:]' '[:lower:]')  
        str=$(echo "${str}" | sed "s/ ${x} / ${lc_x} /g")  
    done  
    echo ${str}  
}
```

Add the function to  `${HOME}/bin/functions.sh` and run source  `${HOME}/.bashrc`.

# Opportunities

They do knock every once in a while



<http://dilbert.com/strip/2009-09-24/>

# IT-managed Linux labs

- \* `colossus.it.mtu.edu` and `guardian.it.mtu.edu`
  - \* Intel Xeon X5675 3.07 GHz, 24 CPU cores, 96 GB RAM
  - \* Accessible for all from anywhere via SSH using a Terminal
  - \* Appropriate for light- to medium-weight computations
- \* Linux workstation in a campus lab/office
  - \* May not be as powerful as `colossus.it` or `guardian.it`
  - \* May not be directly accessible from off-campus
  - \* <https://www.it.mtu.edu/computer-labs.php>

All IT-managed workstations in Linux labs run RHEL 7.x and will mount the campus home directory.

# Network of expertise

UN5390; CRN: 84758

#	Name	Email	Dept/Program	Advisor
01	Adam Mitteer	aamittee	Data Science	Mari Buche
02	Ashley Kern	ankern	Data Science	Mari Buche
03	Eassa Hedayati	hedayati	Physics	John Jaszcak
04	Hashim Mahmud	hnalmahm	ME-EM	Gregory Odegard
05	Jeffrey Brookins *	jmbrooki	MSE	Jaroslaw Drellich
06	Paul Roehm	pmroehm	ME-EM	Gregory Odegard
07	Qing Guo	qinguo	Physics	Ravindra Pandey
08	Subin Thomas	subint	Physics	Raymond Shaw

\* Undergraduate students



# Network of expertise

BE5390: Biomedical Engineering CRN: 84759

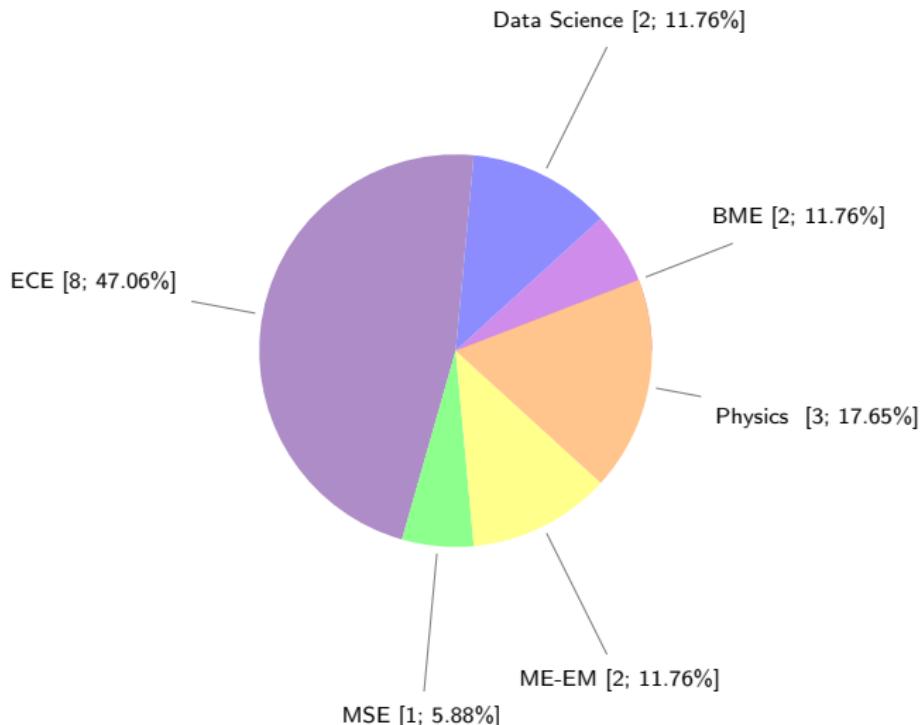
#	Name	Email	Advisor
09	Cal Riutta *	cdriutta	Jinfeng Jiang

EE5390: Electrical and Computer Engineering; CRN: 84760

10	Akhil Kurup	amkurup	Michael Roggemann
11	Avinaash Kovvuri	askovvur	Michael Roggemann
12	Ian Cummings	itcummin	Timothy Havens
13	Prithvi Kambhampati	pkambham	Michael Roggemann
14	Sandeep Lanka	slanka	Michael Roggemann
15	Sameer Saraf	svsaraf	Michael Roggemann
16	Shuo Wang	wshuo	Jeremy Bos
17	Zhiqiang Zhao	qzzhao	Zhuo Feng

\* Undergraduate students

# Network of expertise



17 registered students.

# NSF Graduate Research Fellowship Program 2017

- \* Applicant must be a US citizen or a permanent resident
- \* Fellowship supports 3 years of study
  - \$34k of stipend per year +
  - \$12k of cost-of-education allowance to the university per year
- \* MS and PhD candidates in STEM and STEM education
  - Must be in first two years of graduate study
  - Senior undergraduates are also encouraged to apply
- \* Michigan Tech Information Session
  - 5 pm, 7th September 2016 (Wednesday), Admin 404



# CareerFEST and Career Fair

- \* More details at <http://www.mtu.edu/career/careerfest/>
- \* Create/Update your two-page résumé
- \* Have it critiqued by Michigan Tech Career Services
- \* Develop the habit of reviewing/updating it once per month
- \* Use the  $\text{\LaTeX}$  template in [\\$\{UN5390\}/\text{LaTeXTemplates}/\text{Resume}/\\$](#)
- \* Additional resources
  - <http://www.mtu.edu/career/students/toolbox/resumes/examples/>
  - <http://owl.english.purdue.edu/owl/resource/719/1/>
  - <http://www.sharelatex.com/templates/cv-or-resume>
  - <http://www.latextemplates.com/cat/curricula-vitae>

CareerFEST is a collection of many different informal events that take place during the month of Career Fair.



- \* Commonly used Linux commands
- \* Extensive shell scripting
- \* Revision control (Git)
- \* Workflow development
- \* Statistical analysis (Python, R and Gnuplot)
- \* Visualization (Python, R and Gnuplot)
- \* White papers and internal publications ( $\text{\LaTeX}$ )



- \* Commonly used Linux commands
- \* Extensive shell scripting
- \* Revision control (Git/Subversion)
- \* Workflow development
- \* Domain-specific expertise
- \* Modeling, simulation, analysis and visualization
  - Choice of language/toolset depends on a project
- \* White papers, internal and external publications ( $\text{\LaTeX}$ )



# Keweenaw Climate Science Event

#1 of four-part event

## The Orpheum Theater

6 – 8 pm on Thursday, 8th September 2016

### Subsequent events

6th October 2016

3rd November 2016

1st December 2016

No admission fee

Free pizza and soft drinks

[More information](#)

Organized by [Keweenaw Climate Community](#), and sponsored by the local chapter of the [American Chemical Society](#) and the [Department of Social Sciences](#) at Michigan Tech.



# Keweenaw Climate Science Event

#2 of four-part event

## The Orpheum Theater

6 – 8 pm on Thursday, 6th October 2016

### Subsequent events

3rd November 2016

1st December 2016

No admission fee

Free pizza and soft drinks

[More information](#)

Organized by [Keweenaw Climate Community](#), and sponsored by the local chapter of the [American Chemical Society](#) and the [Department of Social Sciences](#) at Michigan Tech.



# ICC Distinguished Lecture

## CS For All: Considering The Implications Of 'For All'

Dr. Kamau Bobb

Research Scientist, Georgia Tech

Program Officer, CISE, NSF



4th October 2016 1 pm, ME-EM 406

<https://www.ceismc.gatech.edu/about/staffdirectory/kamau-bobb>

- \* Applicant must be a US citizen
- \* Stipend available (up to \$650/week)
- \* Travel to and from appointment site
- \* Open to undergraduate and graduate students, and post-graduates
- \* Opportunities to learn from top scientists and subject matter experts, and career possibilities
- \* Accounting and Finance, Business, Communications, Computer Science and Information Technology, Engineering, Environmental Sciences, Law, Physical and Mathematical Sciences, Policy, Program Management, Safety and Health, and other related areas



US Department of Energy is the nation's leading sponsor for scientific research.

US Department of Energy Scholars Program | Application deadline: 15th December 2016

# US DOE Computational Science Graduate Fellowship 2017

- \* Applicant must be a US citizen
- \* Open to senior undergraduates and first year graduate students (MS/PhD candidates without MS)
- \* Fellowship renewable up to 4 years
  - \$36k of stipend per year +
  - Full tuition and other fees +
  - \$5k academic allowance in first year +
  - \$1k academic allowance for each renewed year
- \* Opportunity to attend annual program review
- \* 12-week research practicum



US Department of Energy is the nation's leading sponsor for scientific research.  
US DOE Computational Science Graduate Fellowship | Application deadline: 18th January 2017

# US DHS HS-STEM Summer Internship 2017

- \* Applicant must be a US citizen
- \* Open to undergraduates and graduate students majoring in homeland security disciplines
- \* 10-week program with stipend  
\$6k for undergraduates / \$7k graduate students + Limited travel expenses
- \* Exposure to and participation in research in DHS mission-relevant research areas
- \* Networking (w/ scientists & laboratories) and career opportunities



Oak Ridge Institute for Science and Education (ORISE) administers this internship program through an interagency agreement between the US Department of Energy and the US Department of Homeland Security. ORISE is managed by Oak Ridge Associated Universities (ORAU) for US DOE.  
US DHS HS-STEM Summer Internship Program | [@GovCareerPaths](#) | Application deadline: 7th December 2016

# Keweenaw Climate Science Event

#3 of four-part event

## The Orpheum Theater

6 – 8 pm on Thursday, 3rd November 2016



### Subsequent event

1st December 2016

No admission fee

Free pizza and soft drinks

[More information](#)

Robert Handler (1982 – present): Adj. Asst. Professor in Civil and Environmental Engineering and Operations Manager for Sustainable Futures Institute, Michigan Tech; PhD in Environmental Engineering from The University of Iowa (2009)

Stephen Handler (1982 – present): Climate Change Specialist in US Forest Service and Northern Institute of Applied Climate Science (NIACS); MS in Resource Conservation and International Conservation and Development from University of Montana (2007)

Organized by Keweenaw Climate Community, and sponsored by the local chapter of the American Chemical Society and the Department of Social Sciences at Michigan Tech.

# 41 North Film Festival



Destination Cinema at Michigan Tech  
ROZSA CENTER FOR THE PERFORMING ARTS  
**NOVEMBER 3-6, 2016**

## Predator/Prey: The Fight For Isle Royale Wolves

Saturday, 5th November 2016, 7:30 pm  
Open and free to public

<http://hdmzweb.hu.mtu.edu/41north/2016/> | <https://www.facebook.com/41northfilmfest/>

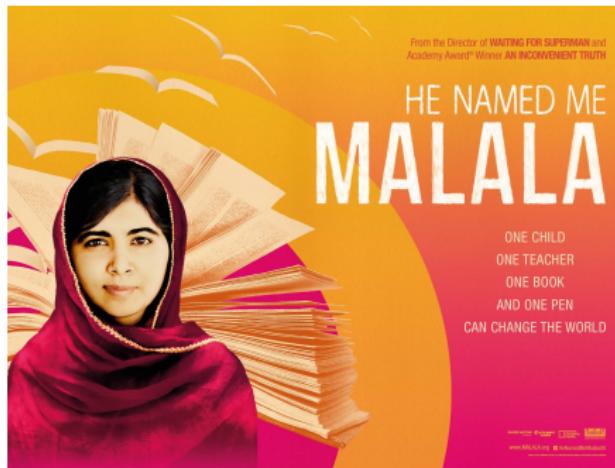


# Before The Flood



Limited free screening on [iTunes](#), [National Geographic](#) and [YouTube](#)

# International Education Week



Friday, 11th November 2016, 5:30 pm, Fisher 135  
Free admission

Malala Yousafzai (1997 – present): Pakistani activist for female education, and a recipient of the 2014 Nobel Peace Prize.  
Sponsored by Michigan Tech Provost Office, International Programs and Services, and Michigan Tech Film Board.

# Mathematical Results

Standing the test of time

Mathematics, rightly viewed, possesses not only truth, but supreme beauty – a beauty cold and austere, like that of sculpture, without appeal to any part of our weaker nature, without the gorgeous trappings of painting or music, yet sublimely pure, and capable of a stern perfection such as only the greatest art can show.

– Bertrand Russell, A History of Western Philosophy (1945)



Bertrand Arthur William Russell (1872 – 1970): British philosopher, logician, mathematician, historian, writer, social critic, and political activist. 1950 Nobel Laureate in Literature.

# Fundamental theorem of algebra

Every non-constant single-variable polynomial with complex coefficients has at least one complex root. Since real numbers are a subset of complex numbers, the result/statement extends to polynomials with real coefficients as well.

## Alternate statement #1 (proved using successive polynomial division)

Every non-zero, single-variable, degree  $n$  polynomial with complex coefficients has, counted with multiplicity/degeneracy, exactly  $n$  roots.

## Alternate statement #2

The field of complex numbers is algebraically closed.

Theorem first proven algebraically by James Wood (with missing steps) in 1798, and geometrically by Johann Carl Friedrich Gauss (with a topological gap) in 1799.



# Fundamental theorem of calculus

Suppose that  $f(x)$  is defined and continuous on  $[a, b]$ . Suppose that  $y(x)$  is an anti-derivative of  $f(x)$ . Then

$$\int_a^b f(x) dx = y(b) - y(a)$$

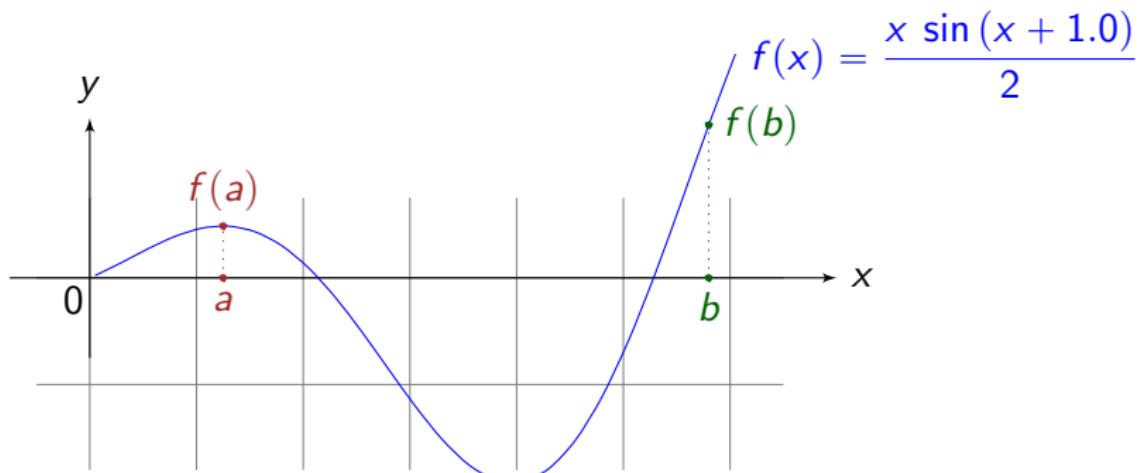
Changing the notations while retaining the underlying essence,

$$\int_{t_n}^{t_{n+1}} f(y, t) dt = y_{n+1} - y_n$$

Re-arranging the terms,

$$y_{n+1} = \boxed{y_n} + \boxed{\int_{t_n}^{t_{n+1}} f(y, t) dt}$$

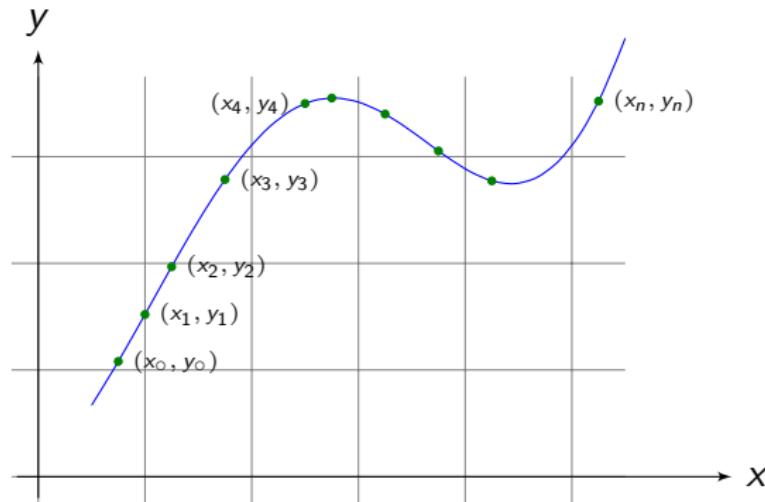
## Intermediate value theorem (IVT)



For any function  $f(x)$  that is continuous on  $[a, b]$ , and has values  $f(a)$  and  $f(b)$  at  $a$  and  $b$  respectively, then  $f(x)$  also takes any value between  $f(a)$  and  $f(b)$  at some point within the interval.

# Lagrange polynomial interpolation

Suppose that  $(x_i, y_i)$ , with  $i = 0 : 1 : n$ , are a set of  $n + 1$  unique points



Joseph-Louis Lagrange (1736 – 1813): Italian mathematician and astronomer  
[Interpolating Polynomials](#), L. Shure, MathWorks  
[Lagrange Interpolating Polynomial](#), B. Archer, Wolfram

# Lagrange polynomial interpolation

The general form of Lagrange interpolating polynomial, one that passes through  $n + 1$  points

$$\mathcal{L}_n(x) = \sum_{i=0}^n l_i(x) y_i$$

Lagrange basis polynomials are given by

$$l_i(x) = \prod_{\substack{m=0 \\ m \neq i}}^n \frac{x - x_m}{x_i - x_m}$$

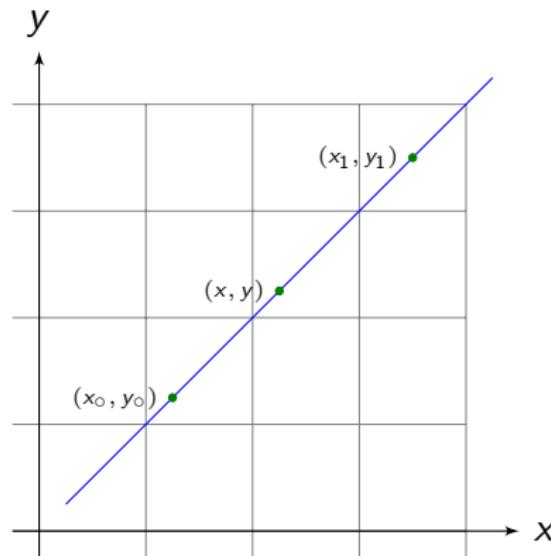
and are built to have the *Kronecker delta* property

$$l_i(x_j) = \delta_{ij}$$

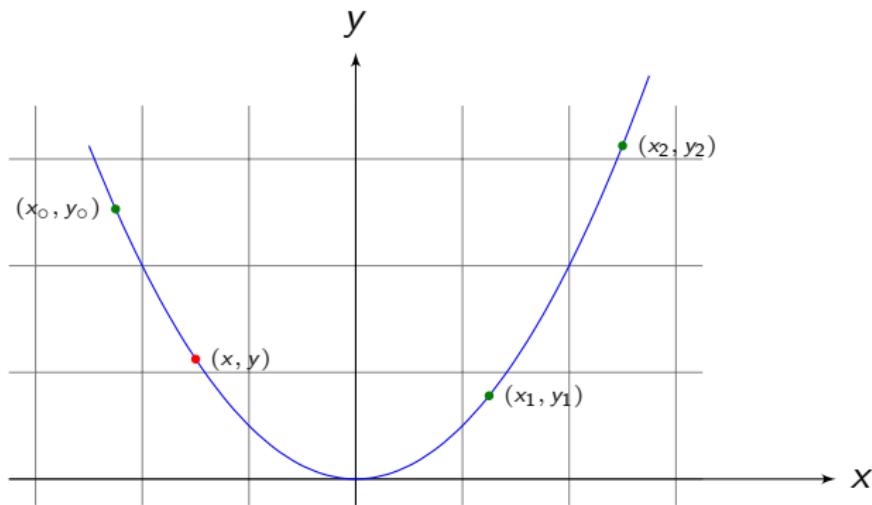
# Lagrange polynomial interpolation

Linear

Suppose that  $(x_0, y_0)$  and  $(x_1, y_1)$  are two known points. The linear interpolant is then a straight line between these two points.



# Lagrange polynomial interpolation Quadratic



$$\mathcal{L}_2(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} y_0 + \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} y_1 + \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} y_2$$

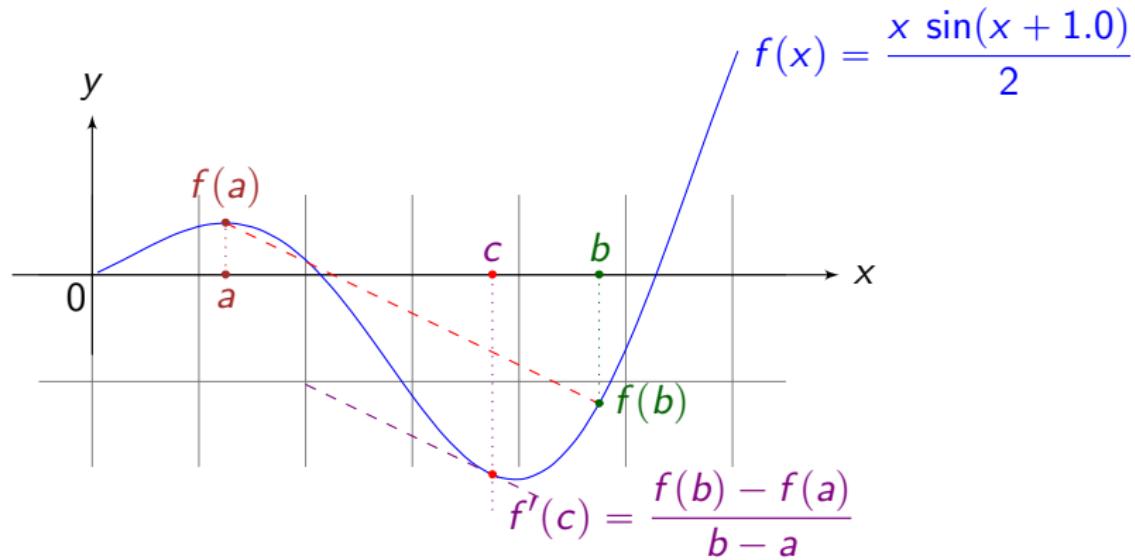
# Lagrange polynomial interpolation

Error analysis

If  $f(x)$  is  $n + 1$  times continuously differentiable on a closed interval  $[a, b]$ , and  $p_n(x)$  is a polynomial of degree at most  $n$  that interpolates  $f(x)$  at  $n + 1$  distinct points  $x_i$ , ( $i = 0, 1, 2, \dots, n$ ) in that interval. Then

$$\epsilon_n = \int_a^b [f(x) - p_n(x)] dx = \int_a^b \frac{f^{(n+1)}}{(n+1)!} \prod_{i=0}^n (x - x_i) dx$$

# Mean value theorem



For any function that is continuous on  $[a, b]$  and differentiable on  $(a, b)$ , there exists a point  $c$  in  $(a, b)$  such that the line joining  $f(a)$  and  $f(b)$  (i.e., the secant) is parallel to the tangent at  $c$ .



## Weighted mean value theorem for integrals

Suppose that  $f(x)$  and  $g(x)$  are continuous on  $[a, b]$ . If  $g(x)$  never changes sign and is positive,  $g(x) \geq 0$ , in  $[a, b]$ , then for some  $c$  in  $[a, b]$

$$\int_a^b f(x) g(x) dx = f(c) \int_a^b g(x) dx$$

# Newton-Cotes formula

Suppose that  $f(x)$  is defined and continuous on  $[a, b]$ .

Consider the integral



$$I = \int_a^b f(x) dx$$

If  $f(x)$  can be approximated by an  $n^{th}$  order polynomial

$$p_n(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \dots + \alpha_{n-1} x^{n-1} + \alpha_n x^n$$

then the integral,  $I$ , takes the form

$$I = \int_a^b [\alpha_0 + \alpha_1 x + \alpha_2 x^2 + \dots + \alpha_{n-1} x^{n-1} + \alpha_n x^n] dx$$

Isaac Newton (1642 – 1727): English physicist and mathematician

Roger Cotes (1682 – 1716): English mathematician (no photo)

# Taylor series expansion

If  $f(x)$  is infinitely differentiable at  $x_0$ , then

$$f(x) = \sum_{n=0}^{\infty} \frac{(x - x_0)^n}{n!} \left. \frac{d^n}{dx^n} f(x) \right|_{x=x_0}$$



A more general form that clearly identifies the error term is given by the  $p^{th}$  order Taylor series expansion of  $f(x)$  with  $\tilde{x} \in [x, x + \Delta x]$

$$f(x + \Delta x) = \sum_{n=0}^p \frac{(\Delta x)^n}{n!} \left. \frac{d^n}{dx^n} f(x) \right|_{x=x} + \frac{(\Delta x)^{p+1}}{(p+1)!} \left. \frac{d^{p+1}}{dx^{p+1}} f(\tilde{x}) \right|_{\tilde{x} \in [x, x + \Delta x]}$$

Brook Taylor (1685 – 1731): English mathematician

# Random variables and distributions

## The need

Random variables and their distributions provide a basis for developing probabilistic models and describing the behavior of important characteristics of interest (i.e., real data).

$Y$  is a random variable if it is a function that assigns a real numbered value to every possible event in a sample space of interest. Since every possible set of values for a random variable  $Y$  corresponds to some event, it has a probability associated with it. A random variable's distribution details the probabilities associated with these sets of values in a meaningful way.

It is a common practice to use an uppercase alphabet to denote the random variable, and the corresponding lowercase alphabet to denote a specific value of this variable. A discrete random variable can assume at most a countable number of values. A continuous random variable can assume an uncountable number of values.

# Random variables and distributions

## PDF and CDF

The probability distribution function (PDF) of some random variable  $Y$  is given below.  $P(Y = y_i)$  indicates the probability of the random variable  $Y$  taking on a given value,  $y_i$ .  $F(y_i)$  represents the cumulative distribution function (CDF), and is used to model the behavior of  $Y$ .

$y_i$	$P(Y = y_i)$	$F(y) = P(Y \leq y_i)$
0	0.10	0.10
1	0.30	0.40
2	0.40	0.80
3	0.20	1.00

All random variables must have a cumulative distribution function.

## Applicable when

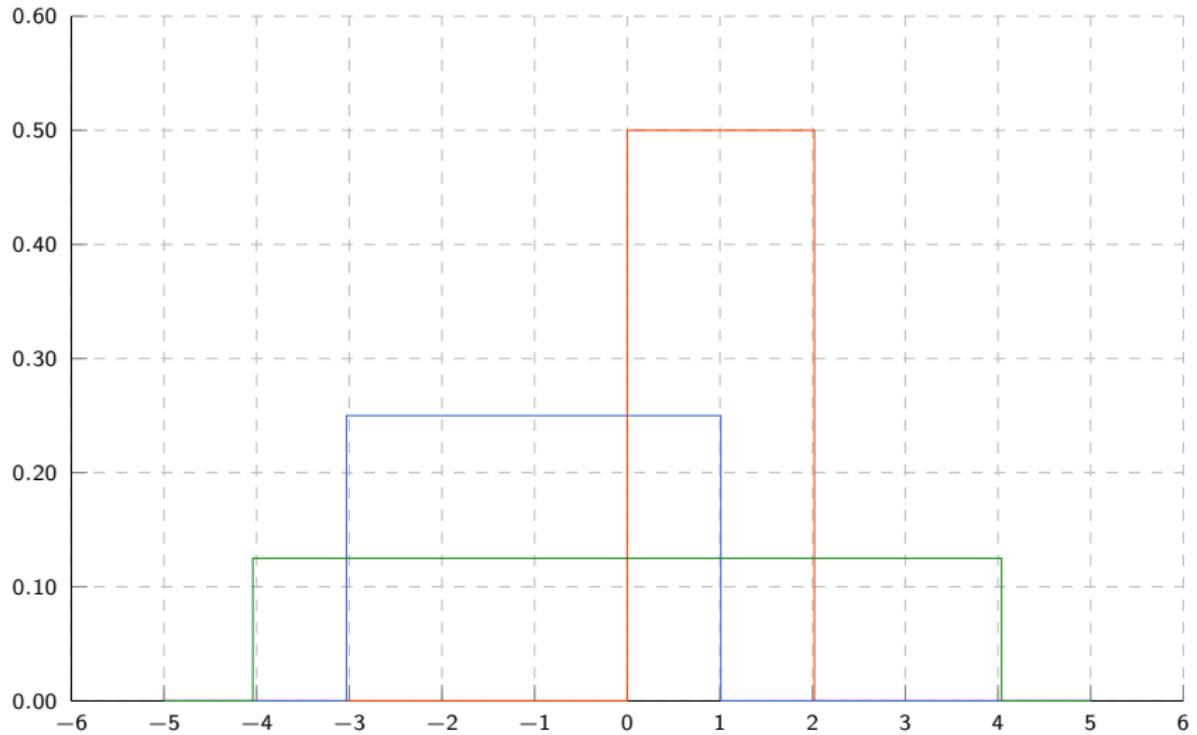
a finite number of values are equally likely to be observed. The probability density function on the interval  $[a, b]$  is

$$f(x) = \begin{cases} 0 & x < a \\ 1/(b-a) & a \leq x \leq b, \text{ and } -\infty < a < b < \infty \\ 0 & x > b \end{cases}$$

## Common example(s)

Throwing a fair die with possible values of 1, 2, ..., 6; each face of the die has a probability of 1/6.

# Uniform distribution



## Applicable when

a random variable takes the value one with success probability of  $p$  and the value zero with a failure probability of  $1 - p$ . Bernoulli distribution is a special case of the Binomial distribution for  $n = 1$ .

## Common example(s)

A coin toss where one and zero could be represented by *head* and *tail* respectively. For a fair coin,  $p = 0.50$ .

# Binomial distribution

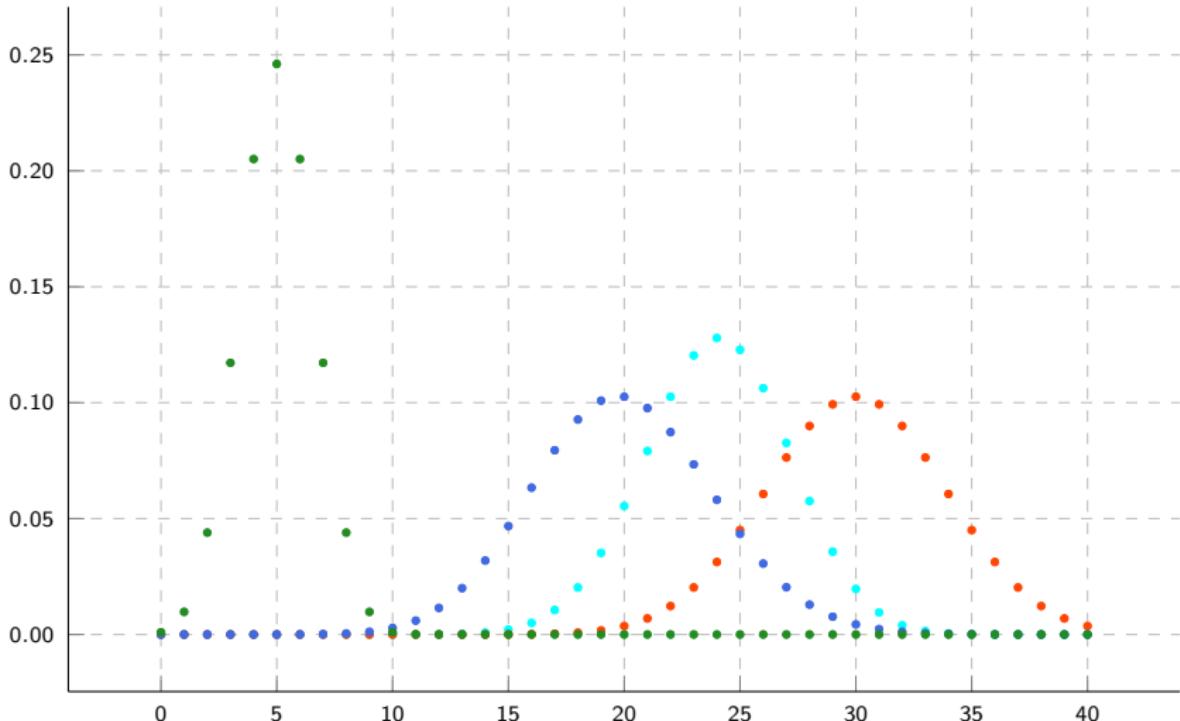
Discrete

## Applicable when

a number of successes (e.g., a head or a tail) results in a sequence of  $n$  independent success/failure-type experiments, each of which yields success with a probability (or fairness factor)  $p$ . The probability of getting exactly  $x$  successes in  $n$  trials for a specified fairness value,  $p$ , is

$$P = \frac{n!}{x! (n-x)!} p^x (1-p)^{n-x}$$

# Binomial distribution



# Poisson distribution

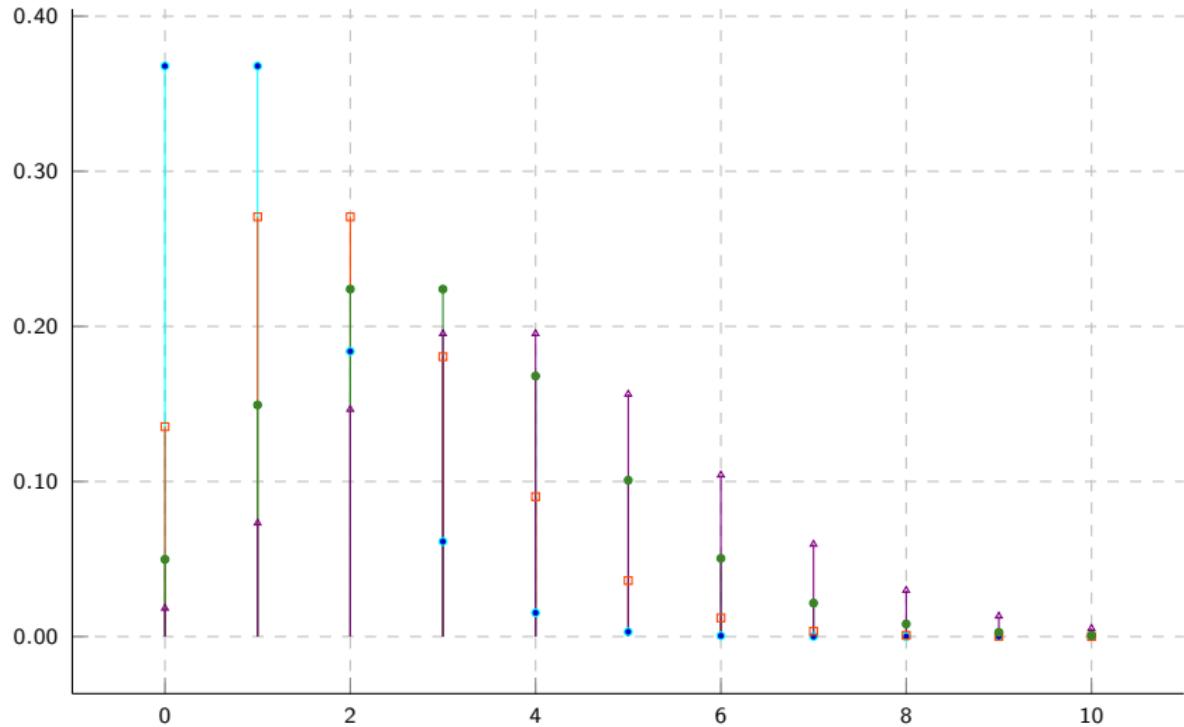
Discrete

## Applicable when

a given number of events occur in a fixed interval of time if these events occur with a known average rate, independently of time since the last event, and two of them cannot occur at the same time. The probability of observing  $m$  events in an interval with the average number of events in an interval designated by  $\lambda$  is

$$P(m) = \frac{\lambda^m e^{-\lambda}}{m!}$$

# Poisson distribution



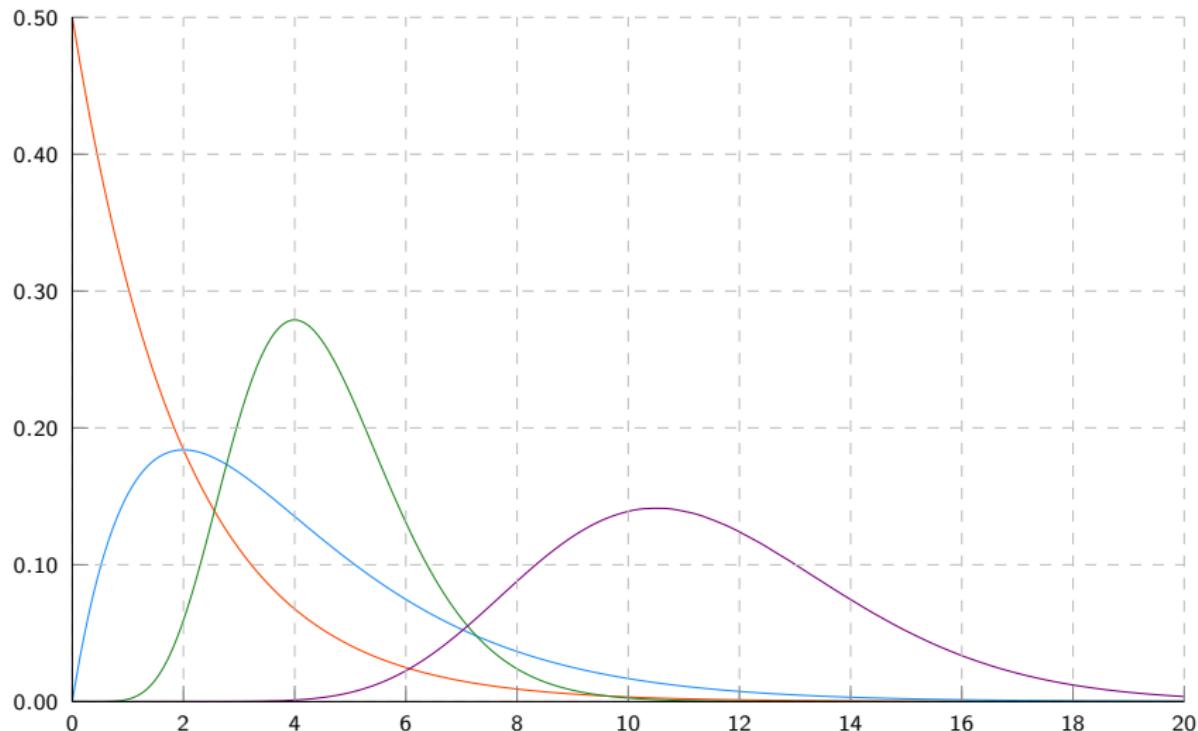
# Gamma distribution

Applicable when

the waiting times between Poisson distributed events are relevant. The probability density function with shape parameter  $\alpha$  and scale parameter  $\beta$  (inverse of rate parameter) is

$$f(x) = \frac{1}{\Gamma(\alpha) \beta^\alpha} x^{\alpha-1} \exp\left(-\frac{x}{\beta}\right) \quad x \geq 0, \text{ and } \alpha, \beta > 0$$

# Gamma distribution



# Normal/Gaussian distribution

Continuous

Applicable as

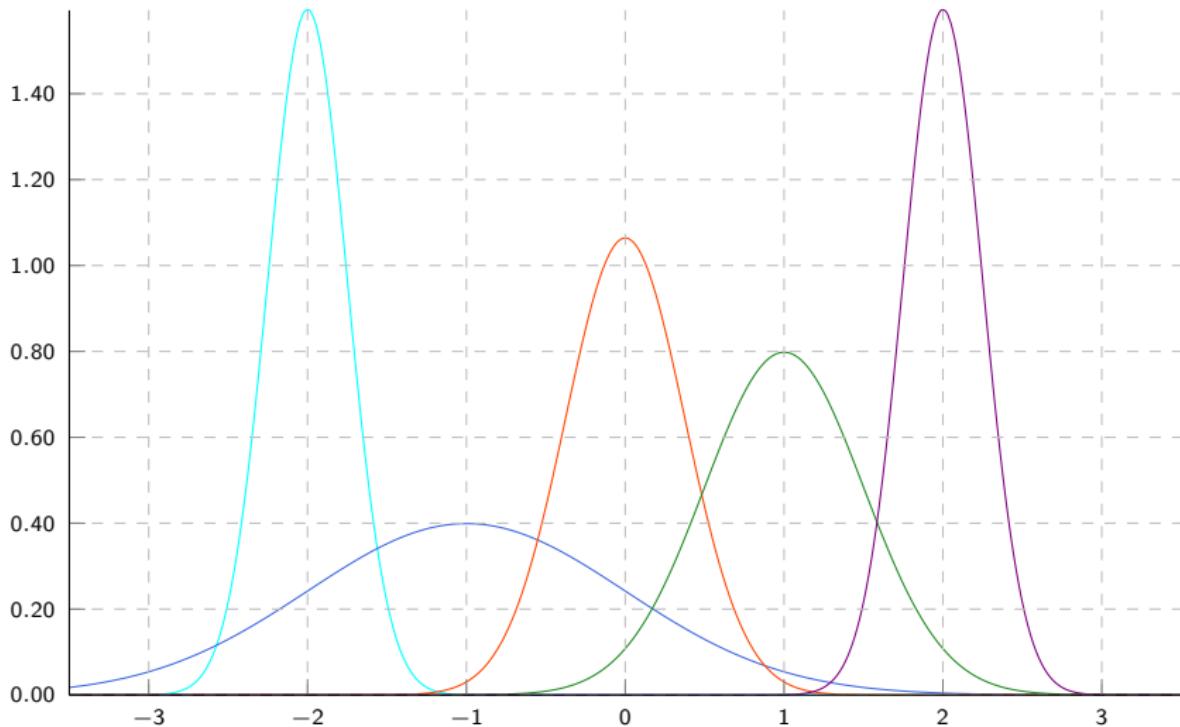
a limiting form of binomial distribution (De Moivre, 1733) and as a plausible distribution for measurement errors (Gauss, 1809). The probability density with mean  $\mu$  and standard deviation  $\sigma$  is

$$f(x) = \frac{1}{\sigma \sqrt{2\pi}} \exp\left[-\frac{(x - \mu)^2}{2\sigma^2}\right] \quad -\infty < x, \mu < \infty, \text{ and } \sigma > 0$$

## Central limit theorem (Laplace)

Under very general conditions when  $n$  random variables, whatever their distributions, are added together, the distribution of the sum tends towards the normal (i.e., bell shape) as  $n$  increases.

# Normal/Gaussian distribution



# Butcher tableau

The general form of recursive relation for  $s$ -stage Runge-Kutta (RK) method



$$y_{n+1} = y_n + h \sum_{i=1}^s b_i k_i$$

$$k_1 = f(y_n, t_n)$$

$$k_2 = f(y_n + a_{21} k_1, t_n + c_2 h)$$

$$k_3 = f(y_n + a_{31} k_1 + a_{32} k_2, t_n + c_3 h)$$

$$k_s = f(y_n + a_{s1} k_1 + a_{s2} k_2 + \dots + a_{s,s-1} k_{s-1}, t_n + c_s h)$$

John Charles Butcher (1933 – present): New Zealand mathematician

## Butcher tableau

The choice of  $s$  (an integer),  $a_{ij}$  (the coefficients of  $s \times s$  RK matrix),  $b_i$  ( $i = 1, 2, \dots, s$ ; the weights),  $c_i$  ( $i = 2, \dots, s$ ; the nodes), and relationship between  $a_{ij}$  and  $c_i$  uniquely identifies the  $s$ -stage RK method and ensures its consistency.

0	$a_{11}$	$a_{12}$	$\dots$	$a_{1,s-1}$	$a_{1,s}$
$c_2$	$a_{21}$	$a_{22}$	$\dots$	$a_{2,s-1}$	$a_{2,s}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$c_s$	$a_{s1}$	$a_{s2}$	$\cdots$	$a_{s,s-1}$	$a_{s,s}$
	$b_1$	$b_2$	$\cdots$	$b_{s-1}$	$b_s$

$$\sum_{j=1}^{i-1} a_{ij} = c_i \quad i = 2, 3, \dots, s$$

## Butcher tableau

Explicit RK2 method

0	0	0
1	1	0
	1/2	1/2

$[a_{ij}]$  needs to be a lower triangular matrix for explicit methods (i.e.,  $1 \leq j < i \leq s$ ). Drop the explicit mention of zeros

0		
1	1	
	1/2	1/2

Consistency check is satisfied

$$\sum_{j=1}^{i-1} a_{ij} = c_i \quad i = 2 \quad \Rightarrow \quad a_{21} = c_2$$

Recursive expression for RK2 (i.e., improved Euler) method

$$y_{n+1} = y_n + h \sum_{i=1}^2 b_i k_i$$

$$y_{n+1} = y_n + h b_1 k_1 + h b_2 k_2$$

Use  $b_i$  from the Butcher tableau and simplify

$$y_{n+1} = y_n + \frac{h}{2} (k_1 + k_2)$$

## Butcher tableau

Explicit RK4 method

0	0	0	0	0
1/2	1/2	0	0	0
1/2	0	1/2	0	0
1	0	0	1	0
	1/6	1/3	1/3	1/6

Drop the explicit mention of zeros along the diagonal and above

0				
1/2	1/2			
1/2	0	1/2		
1	0	0	1	
	1/6	1/3	1/3	1/6

Recursive expression for RK4 method

$$y_{n+1} = y_n + h \sum_{i=1}^4 b_i k_i$$

$$y_{n+1} = y_n + h b_1 k_1 + h b_2 k_2 + h b_3 k_3 + h b_4 k_4$$

Use  $b_i$  from the Butcher tableau

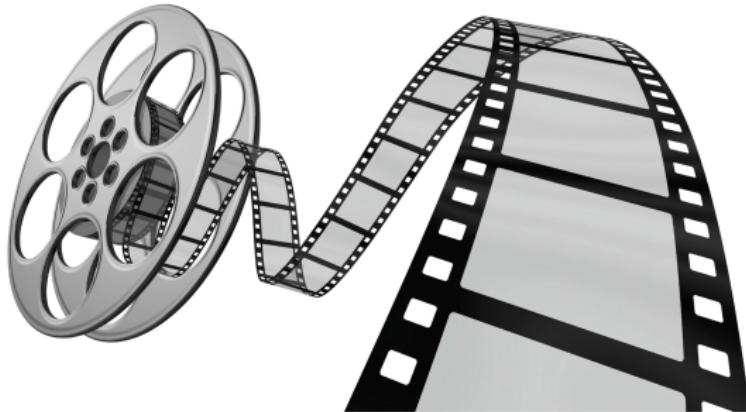
$$y_{n+1} = y_n + \frac{h}{6} k_1 + \frac{h}{3} k_2 + \frac{h}{3} k_3 + \frac{h}{6} k_4$$

Simplify

$$y_{n+1} = y_n + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4)$$

# Videos

If a picture is worth a thousand words ...



# Computer History Museum

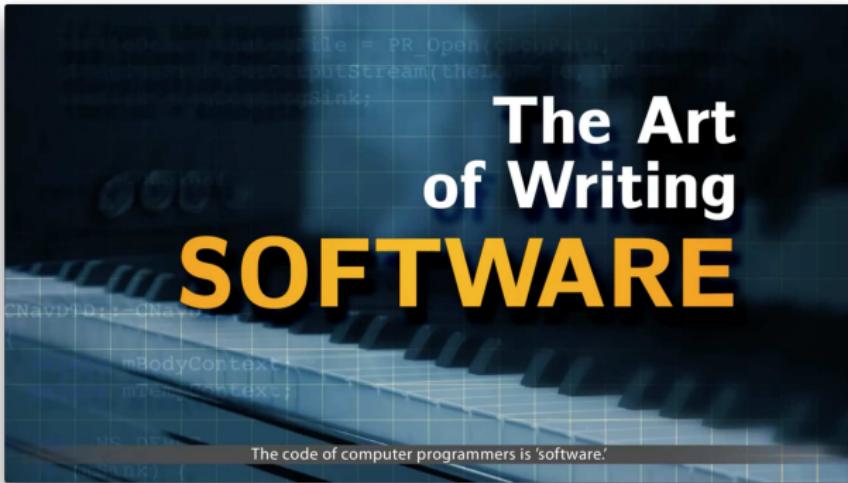


1401 N. Shoreline Blvd., Mountain View, CA 94043  
(650) 810-1010

# The Fairchild Notes



# The Art of Writing Software



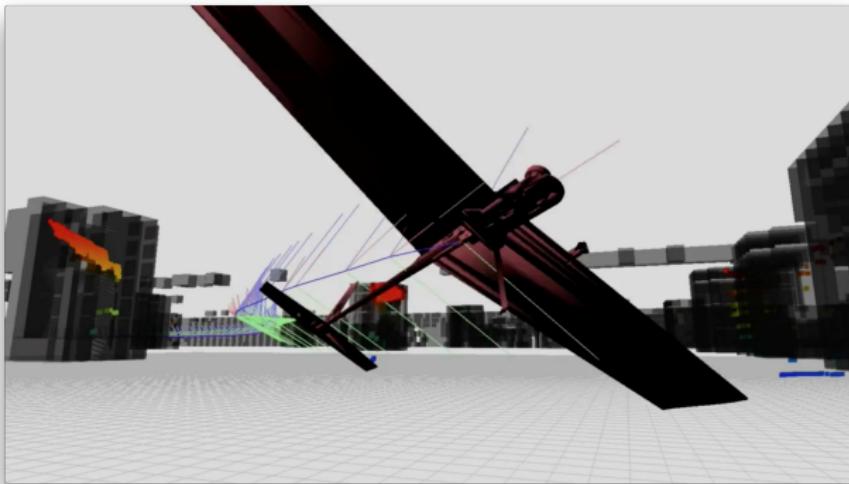
# Supercomputing



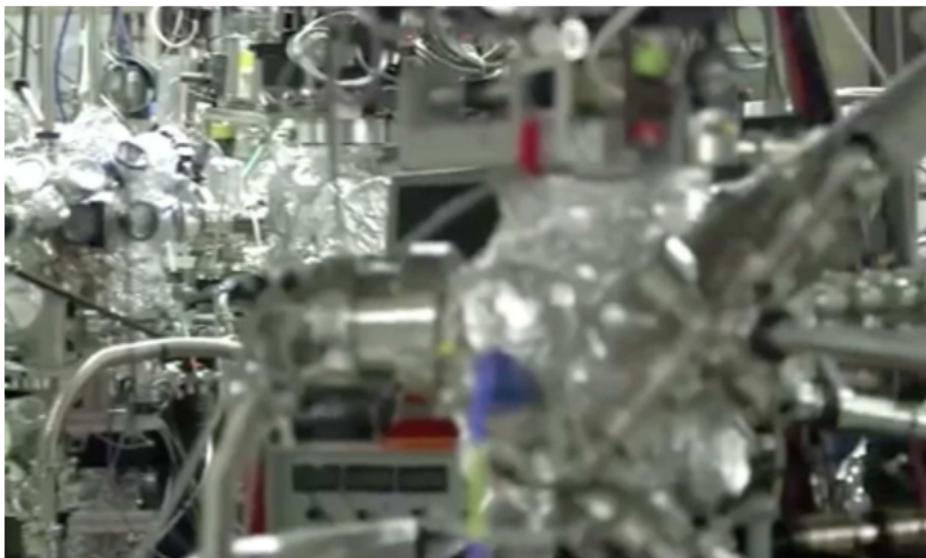
The International Conference for High Performance Computing,  
Networking, Storage and Analysis

What is HPC?

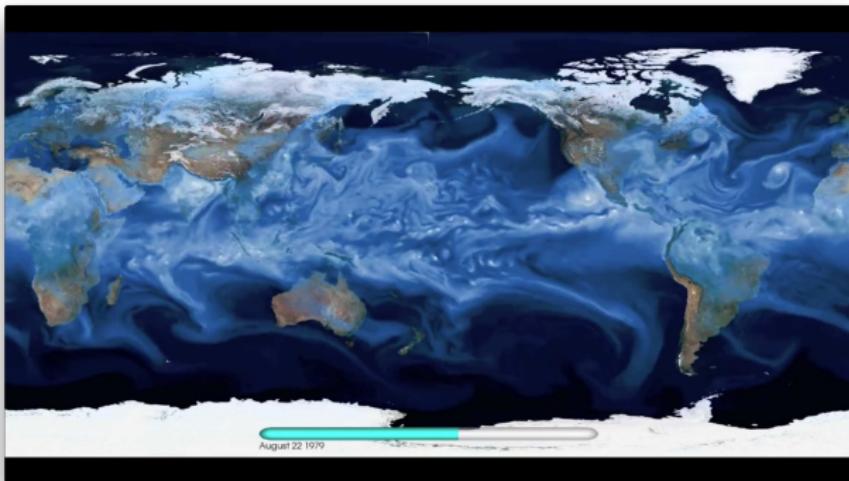
# Aerospace



# Batteries



# Climate modeling



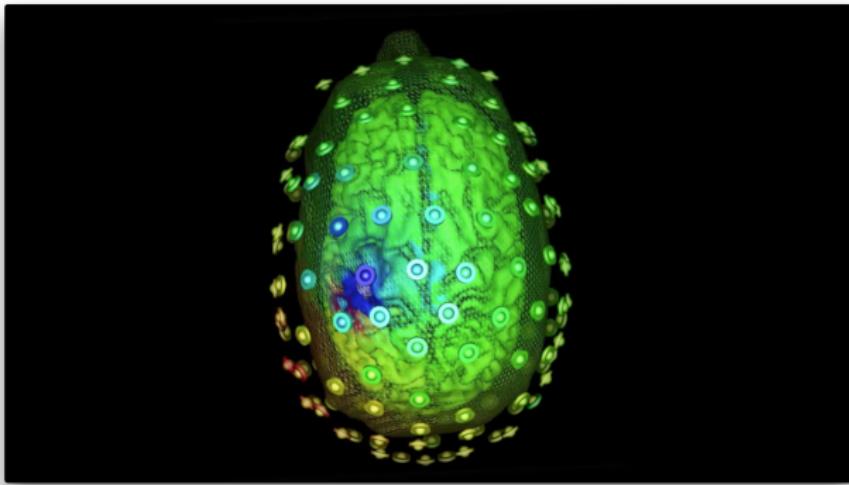
# Diapers, detergents, shampoo



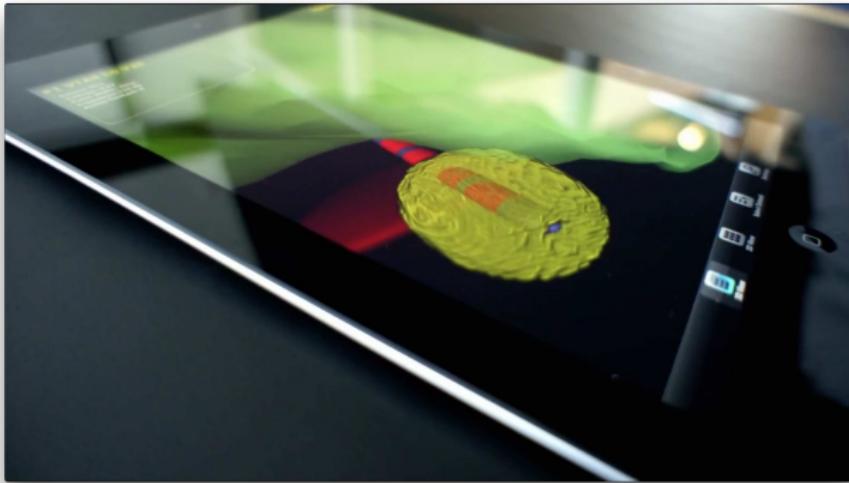
# Entertainment



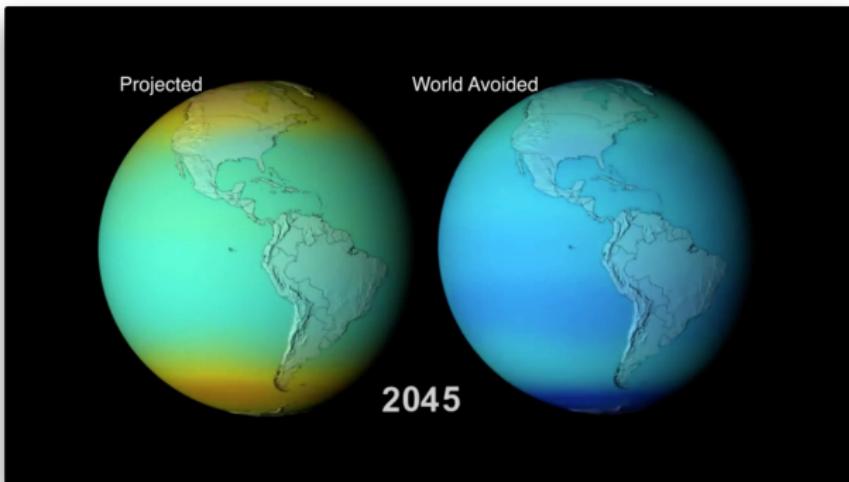
# Epilepsy treatments



# Parkinson's treatments



# Human-induced climate change



# Missing plane, MH370

## Malaysia Airliner MH370: Water of an Airliner

Gowtham Chen, Cong Gu, Philip J. Morris, Er  
Yi-Ching Wang, and Tomasz Wierzbicki

**O**n March 8, 2014, Malaysia Airlines Flight MH370 disappeared less than an hour after take-off on a flight from Kuala Lumpur to Beijing. The Boeing 777-200ER carried twelve crew members and 227 passengers. Captain Zaharie Ahmad Shah, who was flying, announced that "it is believed that the aircraft has crashed" due to "loss of control". It is therefore with deep sadness and regret that I must inform you that the aircraft has crashed in the Southern Indian Ocean. Though the exact fate of the aircraft is still unknown, the available evidence indicates a crash into the ocean. However, although as this is, not all emergency survival equipment was deployed, it appears they are compressed and in tact. In the "Wreck on the Southern Ocean" video, Captain Zaharie Shah, "Sully" Sullenberger and his crew successfully ditched US Airways Flight 1549, an Airbus A320, into the Hudson River in New York City after a bird strike on take-off from La Guardia Airport. That was on January 15, 2009.

Figure 12 and the video animation referenced on the second page of this article show our "trip reconstruction" of a crash into the Southern Indian Ocean.

Gowtham Chen is a member of mathematics at Texas A&M University and Texas A&M University of Qatar (TAU). He is also a member of the Institute for Numerical Simulation at TAU. His email address is gowtham@math.tamu.edu.

Cong Gu is currently a member of the mathematics department of TAU. His email address is conggu@math.tau.ac.il.

Philip J. Morris is a Boeing 737 Walkin Professor of Aerospace Engineering at the University of Southern California. His email address is pjm@usc.edu.

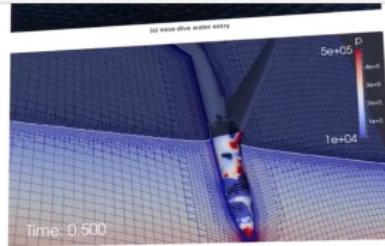


Figure 12. Pitch angle =  $-90^\circ$ , angle of approach =  $93^\circ$ . This corresponds to Case 4. A video animation can be viewed at <https://www.dropbox.com/s/vaf0qenjw0lk5yz/comb-90.mp4>.



Figure 13. Schematics for nose-diving. The ocean current pushes the aircraft to the right, causing it possibly to finish belly-up on the ocean floor.  
This corresponds to Case 4.

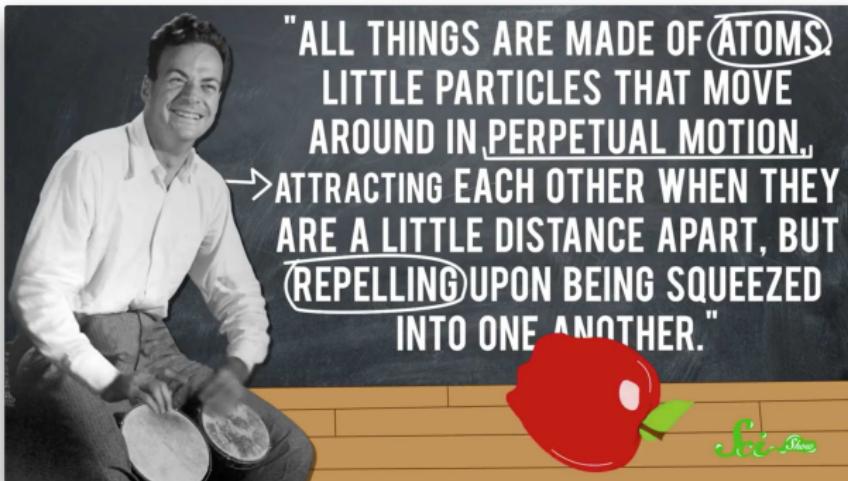
failure modes occur at low impact velocities, as has been demonstrated with a real model of a retired aircraft in DYCAST (Dynamic Crash Analysis of Structures) by NASA [FWR87]. These findings were published nearly three decades ago but remain valid today.

# People and personalities



and their stories

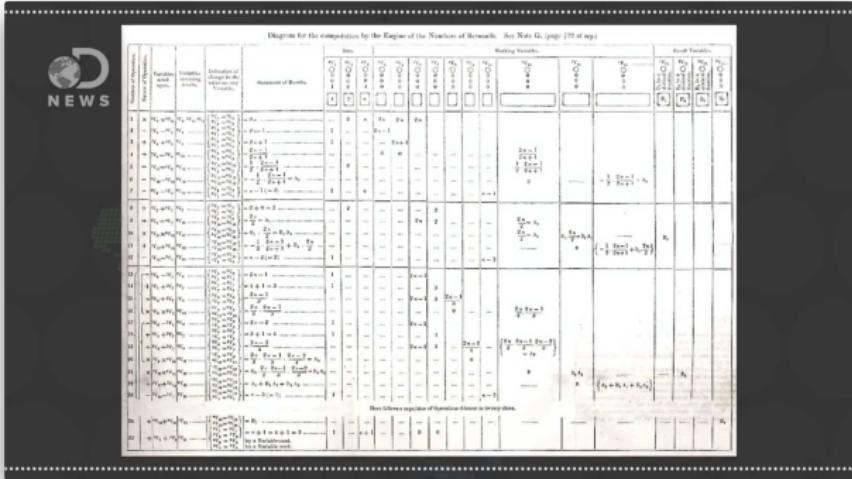
# Richard Phillips Feynman 1918 – 1988



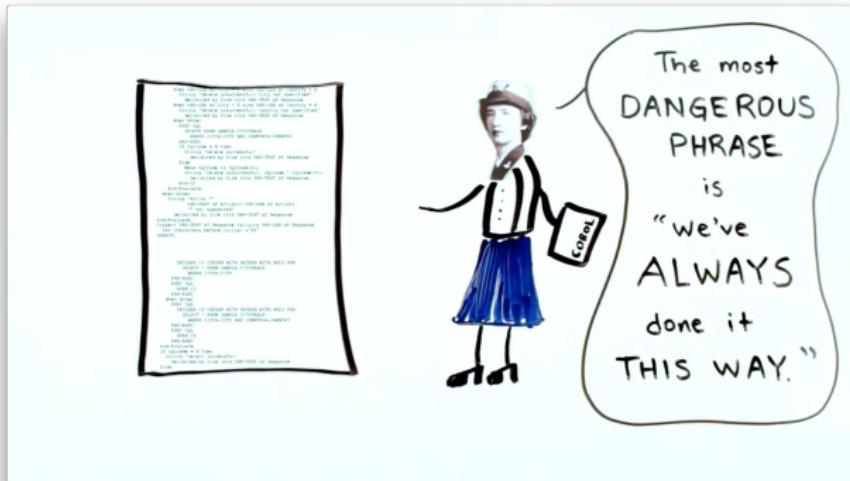
# Alan Mathison Turing 1912 – 1954



# Ada Lovelace 1815 – 1852



# Grace Brewster Murray Hopper 1906 – 1992

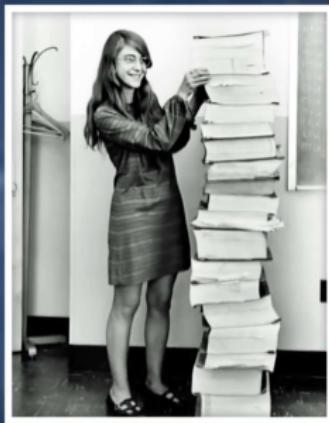


# Margaret Heafield Hamilton

1936 – present

SHE BECAME  
THE HEAD OF THE  
APOLLO FLIGHT  
SOFTWARE  
DEVELOPMENT  
TEAM

Credit: NASA

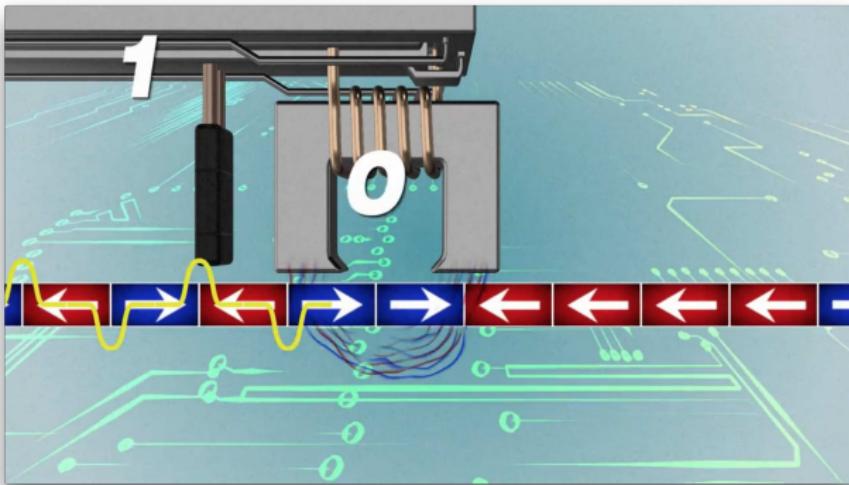


Sci

**TED** Ed

**LESSONS WORTH SHARING**

# Hard Drives



# Algorithm

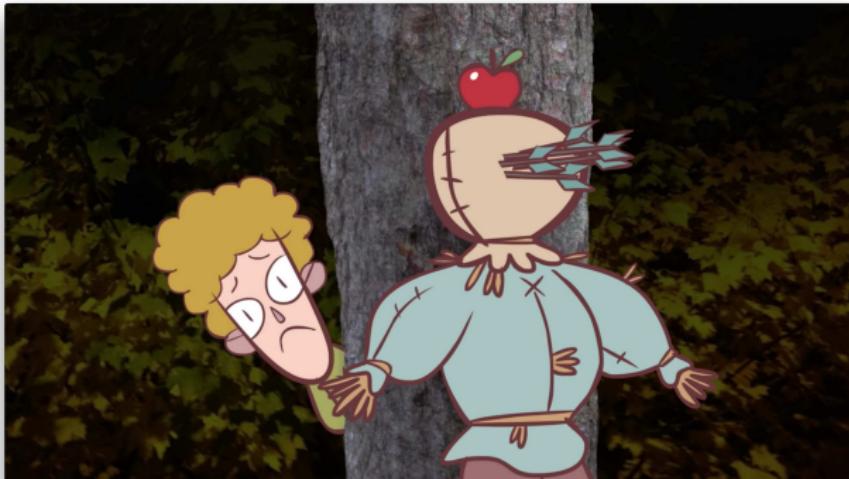
## Pseudocode

let **N** = 0

For each person in room

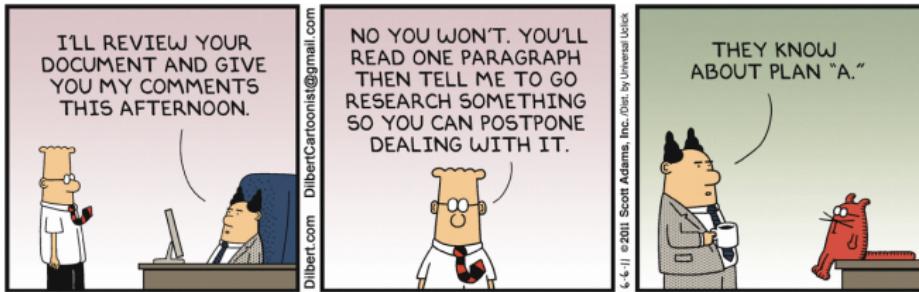
Set **N** = **N** + 1

# Accuracy vs Precision



# Review of Performance

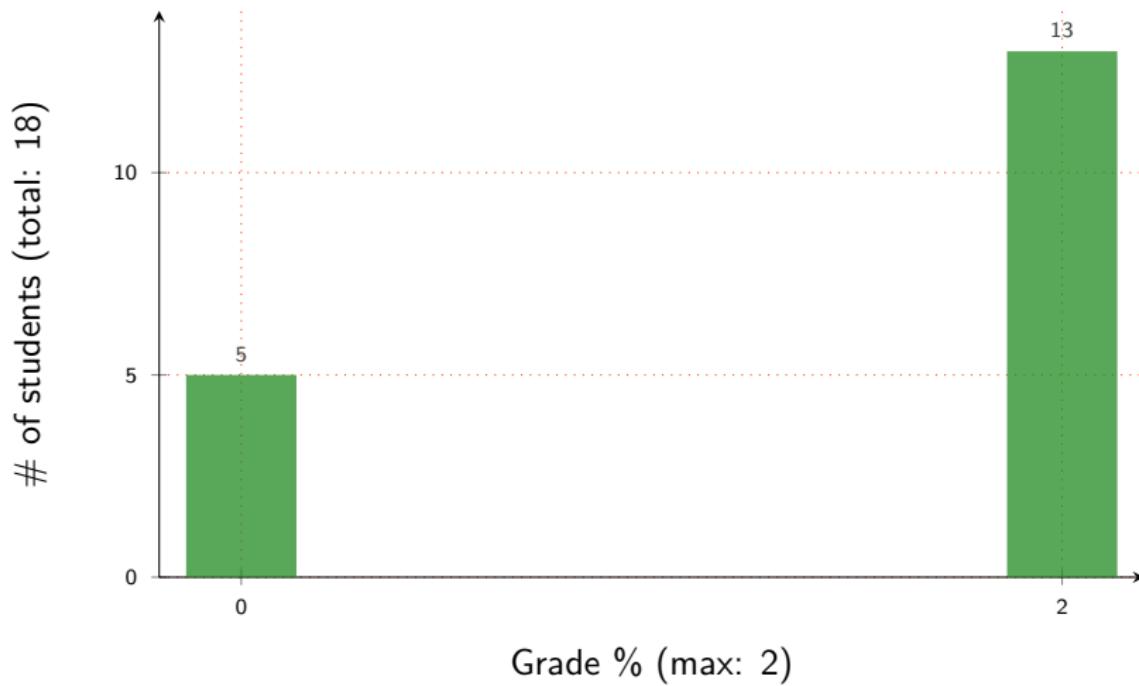
How well have we been performing?



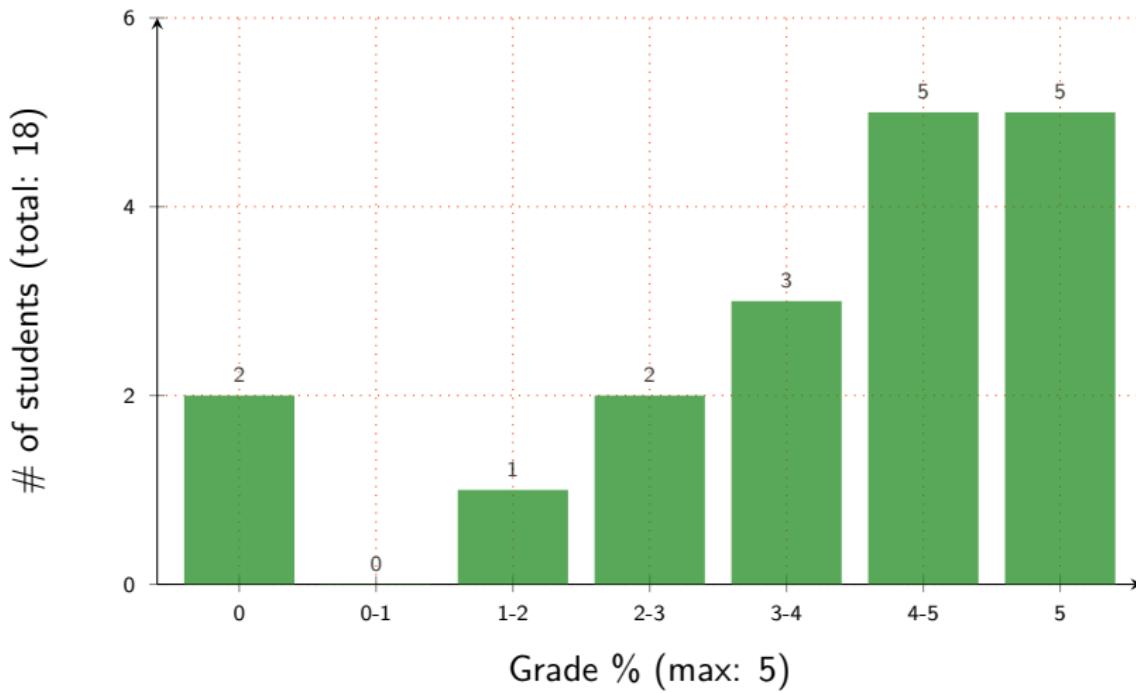
<http://dilbert.com/strip/2011-06-06/>

# Active Participation #01

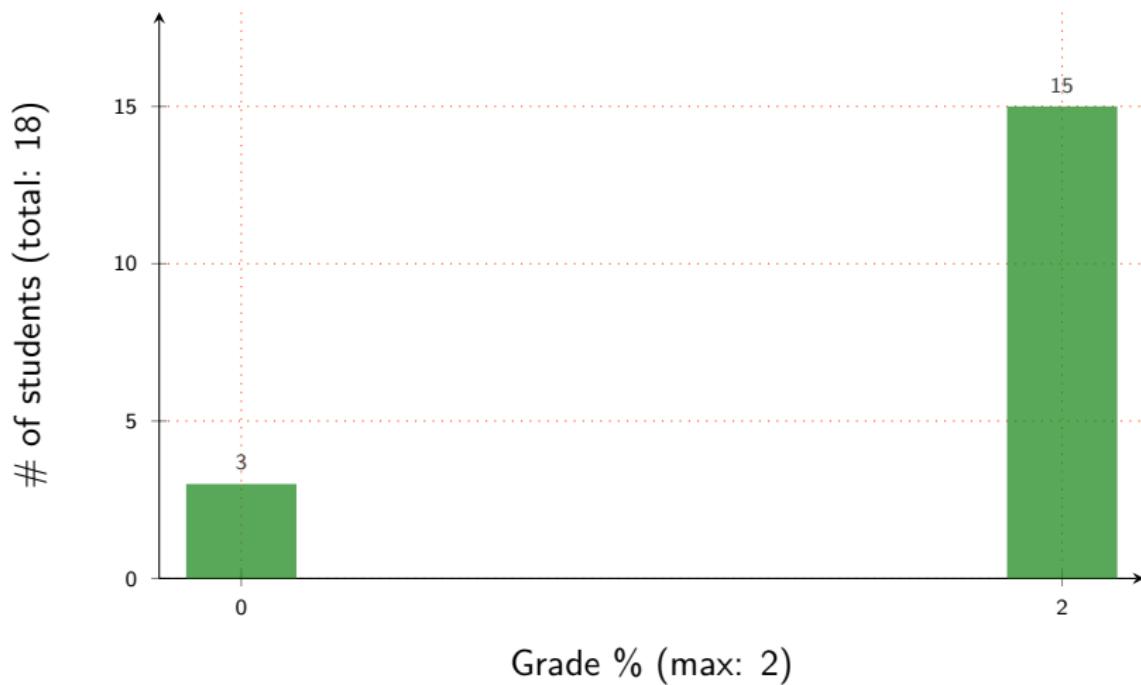
Research Marketing I: Twitter



# Assignment #01

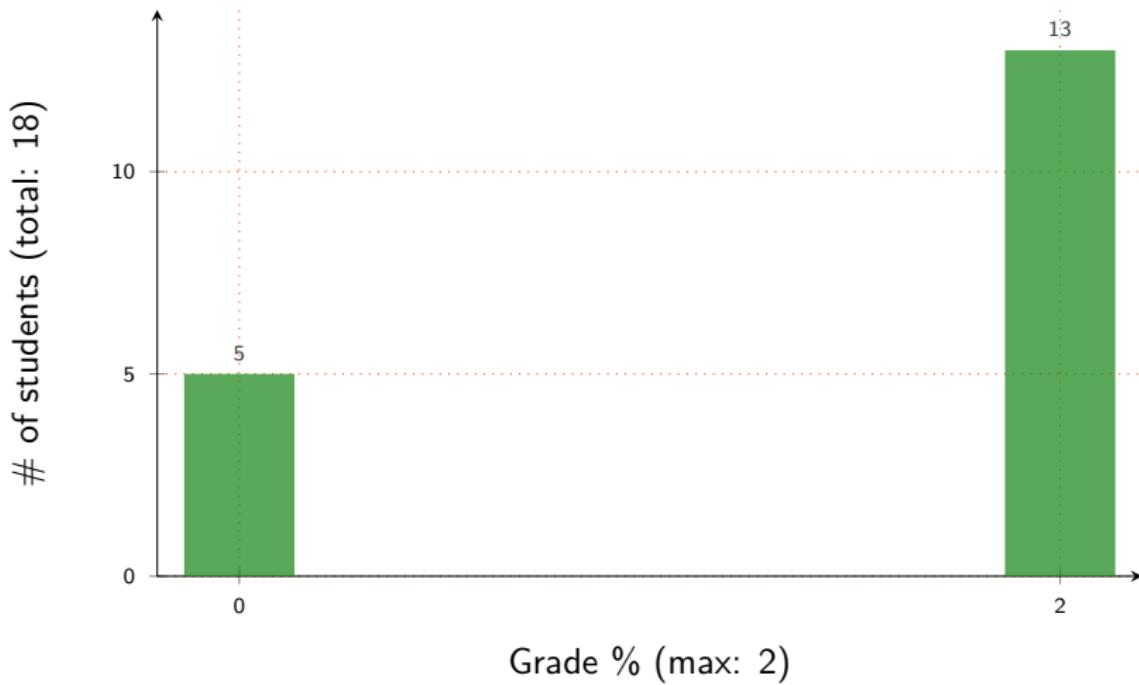


# Active Participation #02 PB&J Sandwich Recipe

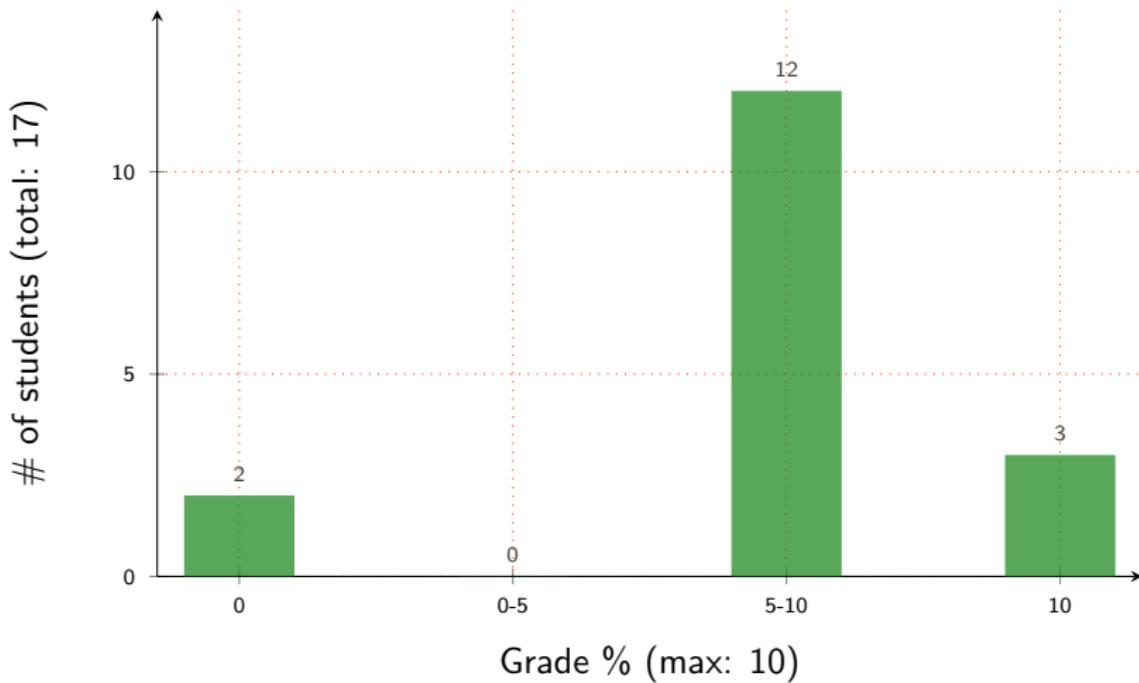


# Active Participation #03

Research Marketing II: Professional/University Business Cards



# Assignment #04



## Superior and Top 500



A proposed compute node in Superior will have two Intel Xeon E5-2698 processors (each processor with 20 cores) at 2.20 GHz, 512 GB RAM, 480 GB Intel Enterprise SSD, Mellanox ConnectX-3 56 Gbps InfiniBand network, and will cost \$13,263.13.

Ignoring the cost of physical space, racks, network, storage, electricity and labor, estimate the cost to build a #500 supercomputer (~405 TFLOPS) with homogeneous compute nodes as the ones described above.

For a computer with  $N$  identical/homogeneous processors,

$$\text{FLOPS} = N \times \text{CPU speed} \times \frac{\text{FLOPs}}{\text{CPU cycle}}$$

Celsius  $\longleftrightarrow$  Fahrenheit



Convert temperature between Celsius and Fahrenheit scales.

Is there a well-known technique to verify the conversion scheme?

## Matrix elements



How many elements in a square matrix of order  $N$ ? How will this number change if the matrix is upper (or lower) triangular?

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \quad \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ 0 & b_{22} & \dots & b_{2n} \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & b_{nn} \end{pmatrix}$$

## The impact and limitations of Moore's Law



Assuming that Moore's Law holds true, what is the speed up of a computer observed over an average adult's life in the US?

## Drawing queens



Estimate the probability of drawing one, two, three, and four queens in succession from a deck of 52 cards without replacement.



## Computing power of your laptop

Estimate the computing power of your laptop in GFLOPS. You may need to check the manufacturer's notes for hardware parameters.

For a computer with  $N$  identical/homogeneous processors,

$$\text{FLOPS} = N \times \text{CPU speed} \times \frac{\text{FLOPs}}{\text{CPU cycle}}$$

# Got questions?

If you do, find a way to contact me; and do so sooner than later

EERC B39 · (906) 487-4096 · [g@mtu.edu](mailto:g@mtu.edu) · @sgowtham

Do not share/distribute the course material, in and/or outside of Michigan Tech, without instructor's prior consent

