UN5390: Scientific Computing I

Dr. Gowtham

Director of Research Computing, IT

Adj. Asst. Professor, Physics and ECE

EERC B39 · (906) 487-4096 · g@mtu.edu · @sgowtham

Week #02: 2016/09/06 and 2016/09/08

Cross-listed as BE5390, EE5390 and MA5390

Recap

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

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

Week #01 Recap

* Revision control system
* A brief history of computing
* Compliance and security

Week #01 Before we meet again

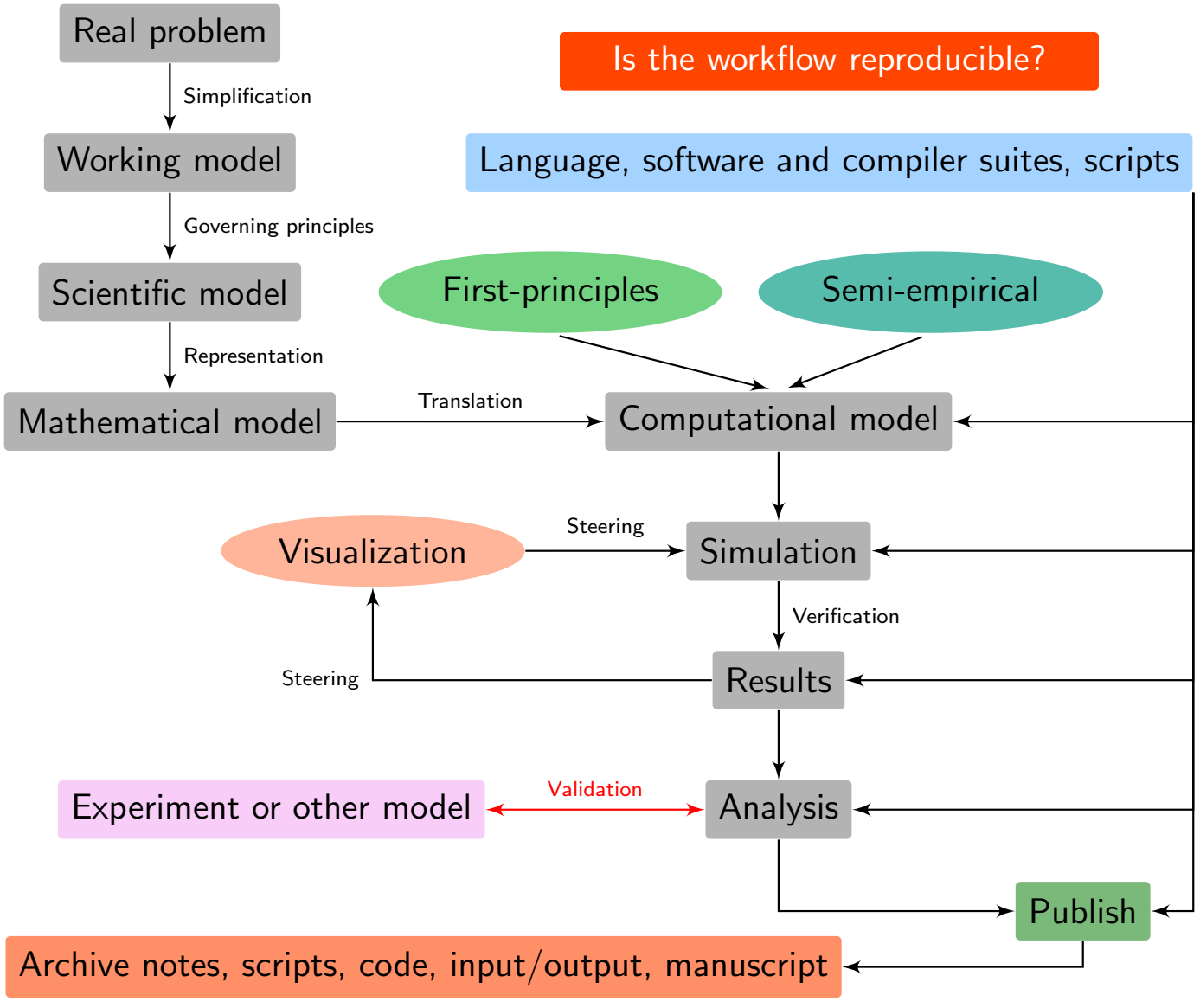
- Review the syllabus, course material, grade through week #02, notations, active participation, free time exercises, tips, opportunities, mathematical results, and videos
- Review/Complete the training camps (#01 – #10)
- Get to know your classmates
- Get a Twitter account. Follow your classmates and accounts given in **Additional references**
- Get started on and make progress in assignment #01
- Think of various components in a computational project
- Email the instructor your favorite poem (name and a link will suffice)

Computational Workflow

The art of being lazy and more efficient

http://dilbert.com/strip/2009-10-11/

Computational workflow



Understanding a tsunami

http://www.uoguelph.ca/geography/research/geog4480_w2015/Group02WebReport/
Illustration by Chris Wren and Kenn Brown; Olivia Carpino, Larissa Goshulak and Reilly McIlhone (University of Guelph)

Understanding a tsunami

* Real problem

  Understand the propagation of disturbances in incompressible fluids

* Working model

  Use a bathtub instead of a real ocean with shallow water approximation: depth of the fluid is small compared to the wave length of the disturbance

* Scientific model

  Derive equations using principles of conservation of mass and momentum

  Time (t) and space coordinates (x and y) are independent variables. Depth of the fluid (h) and two-dimensional fluid velocity field (u and v) are dependent variables. Mass is proportional to h, and momentum is proportional to uh and vh. Acceleration due to gravity (g) is the force acting on the fluid

Understanding a tsunami

* Mathematical model

  Shallow water equations (SWEs):

$$\frac{\partial}{\partial t}h + \frac{\partial}{\partial x}uh + \frac{\partial}{\partial y}vh = 0$$

$$\frac{\partial}{\partial t}uh + \frac{\partial}{\partial x}\left(u^2h + \frac{1}{2}gh^2\right) + \frac{\partial}{\partial y}uvh = 0$$

$$\frac{\partial}{\partial t}vh + \frac{\partial}{\partial x}uvh + \frac{\partial}{\partial y}\left(v^2h + \frac{1}{2}gh^2\right) = 0$$

Understanding a tsunami

* Mathematical model (continued)

  Introduce three vectors:

$$H = \begin{pmatrix} h \\ uh \\ vh \end{pmatrix} \quad U = \begin{pmatrix} uh \\ u^2 h + \frac{1}{2} g h^2 \\ uvh \end{pmatrix} \quad V = \begin{pmatrix} vh \\ uvh \\ v^2 h + \frac{1}{2} g h^2 \end{pmatrix}$$

  SWEs will then be an instance of the hyperbolic conservation law:

$$\frac{\partial H}{\partial t} + \frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = 0$$

Understanding a tsunami

* Mathematical model (continued)

  Confine the problem to a square region and specify reflective boundary conditions: $u = 0$ on vertical sides and $v = 0$ on horizontal sides. Waves that reach the boundary will be reflected back into the region

  Ignore the real world geometry – topography of the ocean floor, Coriolis force from earth's rotation and other potential external forces

  Lax-Wendroff method to compute numerical approximation to the solution. Introduce a regular square finite difference grid with a vector-valued solution centered in the grid cells

  $H^t_{i,j}$ represents a three component vector at each grid cell (i, j) that evolves with time step t

Understanding a tsunami

* Mathematical model (continued)

  Each time step involves two stages. The first stage is a half step: it defines values of H at time step $t + 1/2$ at the midpoints of the edges of the grid

$$H^{t+1/2}_{i+1/2,j} = \frac{1}{2}\left(H^t_{i+1,j} + H^t_{i,j}\right) - \frac{\Delta t}{2\Delta x}\left(U^t_{i+1,j} - U^t_{i,j}\right)$$

$$H^{t+1/2}_{i,j+1/2} = \frac{1}{2}\left(H^t_{i,j+1} + H^t_{i,j}\right) - \frac{\Delta t}{2\Delta y}\left(V^t_{i,j+1} - V^t_{i,j}\right)$$

Understanding a tsunami

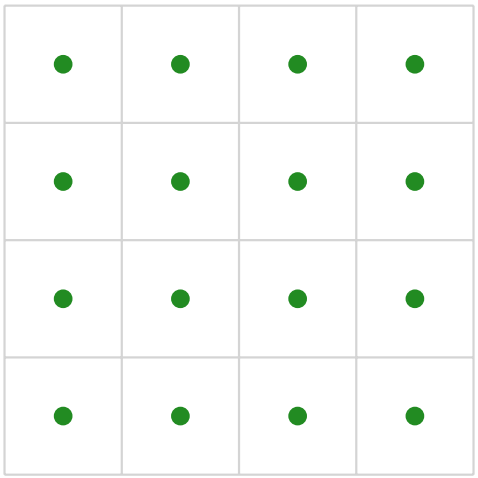
* Mathematical model (continued)

  The second stage completes the time step by using the values computed in the first stage to compute new values at the centers of the cells

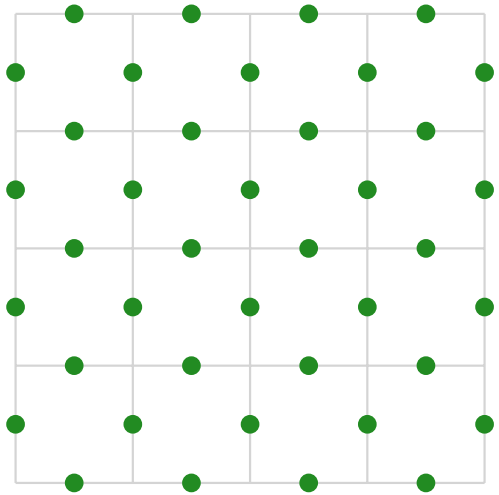
$$H_{i,j}^{t+1} = H_{i,j}^{t} - \frac{\Delta t}{\Delta x}\left(U_{i+1/2,j}^{t+1/2} - U_{i-1/2,j}^{t+1/2}\right) - \frac{\Delta t}{\Delta y}\left(V_{i,j+1/2}^{t+1/2} - V_{i,j-1/2}^{t+1/2}\right)$$

Understanding a tsunami

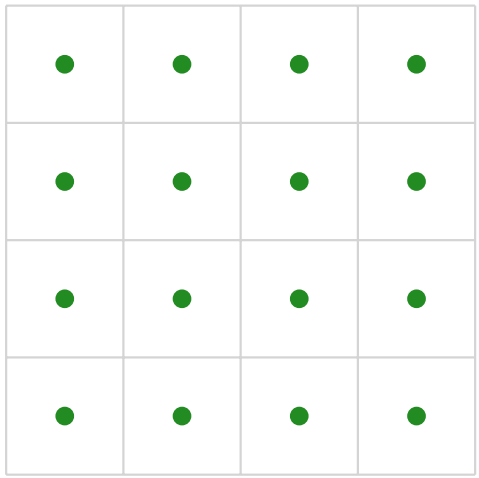
* Mathematical model (continued)



$t = t$



$t = t + 1/2$



$t = t + 1$

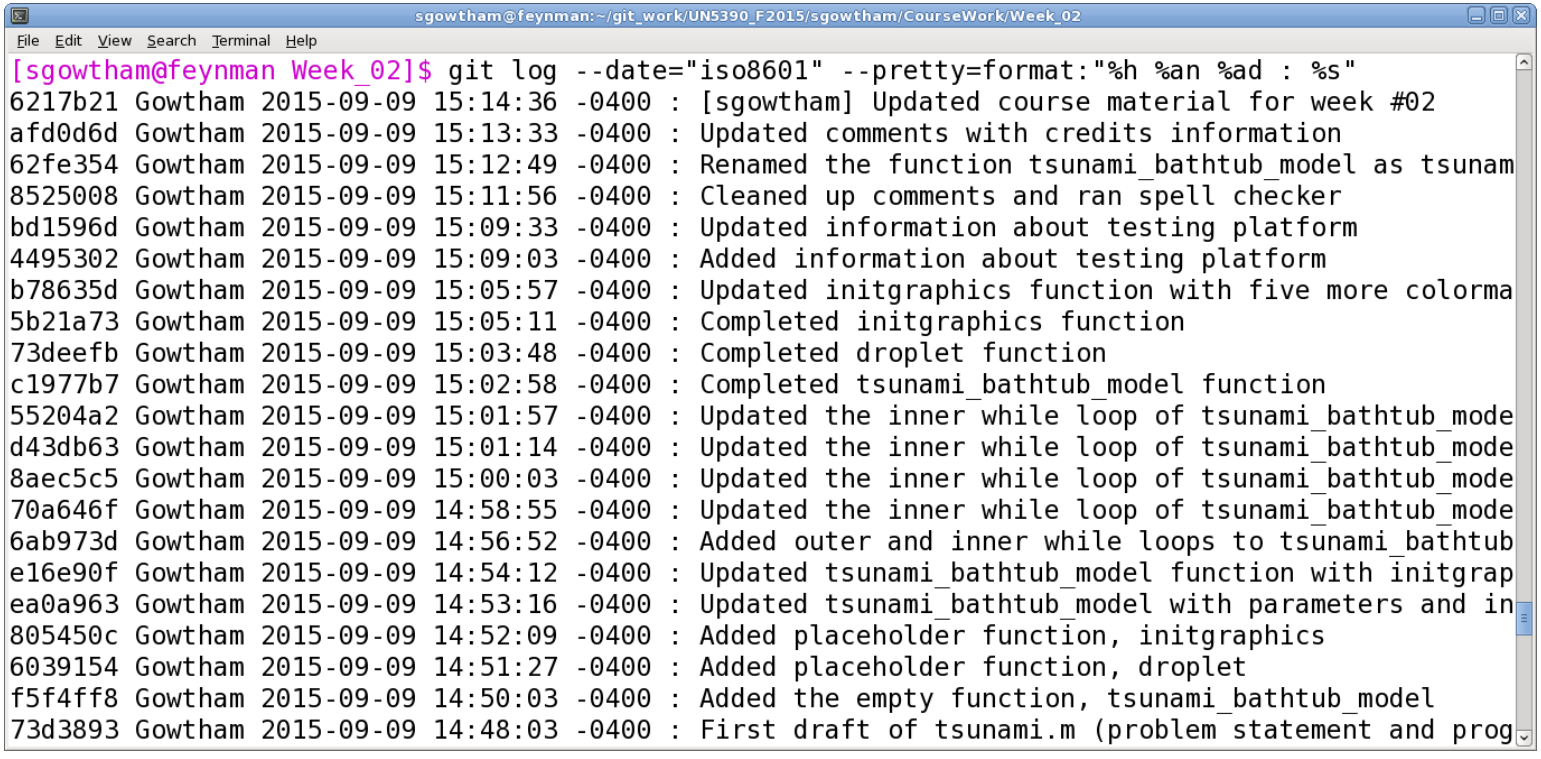
The cost of computing

$N \times N$ grid (with N^2 points) requires a total of $N\,(N + 10)$ values to move from one time step to the next one.

Understanding a tsunami

* Computational model

  Select an appropriate language to develop the code and do so within a revision control system. MATLAB supports coding and visualization

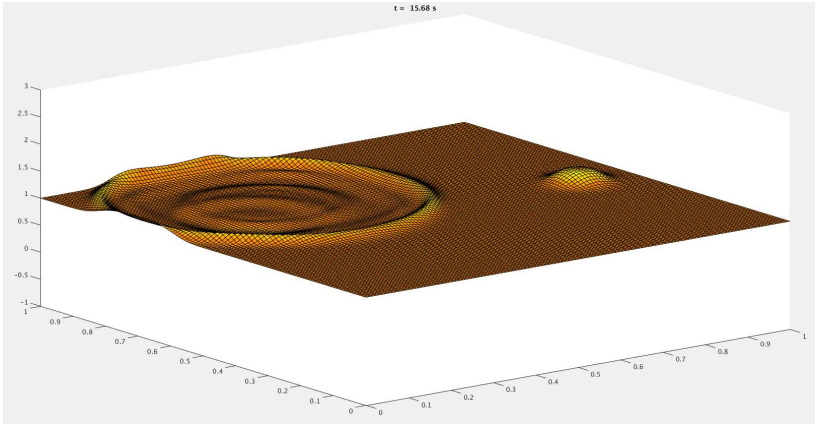


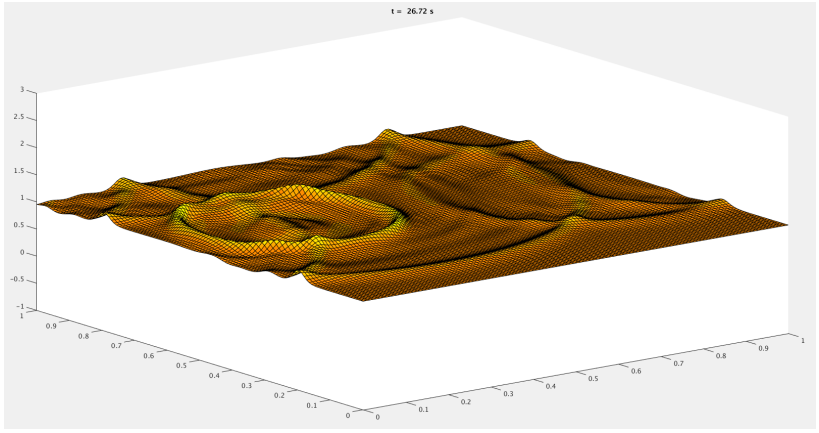
Result of the command `git log --date="iso8601" --pretty=format:"%h %an %ad :  %s"`

Understanding a tsunami

* Simulation, visualization, results and analysis

  Run the code and visualize the results to verify the model





* Experimental data or other model

  Analyze and compare the results to validate the model

  The Lax-Wendroff method amplifies non-physical oscillations – results in overflow of numerical values producing floating point `Inf` and `NaN`

`tsunami.m` in `AdditionalMaterial` folder. Screenshots from MATLAB R2014b.

Terms to know

Computational workflow

The process of combining data and processes into a configurable, structured, understandable and reproducible set of steps that implement (semi) automated computational solutions to a given problem (or a set of them).

Computational steering

The practice of manually intervening or interacting with a running computational process experimenting with *what if* scenarios that involve (sometimes, real time) visual feedback from the steered system.

Additional references

* Systems Of Conservation Laws
  P. Lax, B. Wendroff
  Communications on Pure and Applied Mathematics, vol. 13, p. 217 (1960)

* On The Black Art Of Designing Computational Workflows
  Y. Gil, P. A. González-Calero, E. Deelman
  Proceedings of WORKS'07, p. 53. Monterey, CA, USA (2007)

* The Toolbox Of A Successful Computational Scientist

* 3D Computational Steering With Parametrized Geometric Optics
  J. D. Mulder, J. K. van Wijk
  Proceedings of the 6th Conference on Visualization, p. 304. Atlanta, GA, USA (1995)

PDF in `AdditionalMaterial` folder.

Additional references

* SCIRun: A Scientific Programming Environment For Computational Steering
  S. G. Parker, C. R. Johnson
  Proceedings of 1995 ACM/IEEE Conference on Supercomputing (SC95). San Diego, CA, USA (1995)

* Lightweight Computational Steering Of Very Large Scale Molecular Dynamics
  D. M. Beazley, P. S. Lomdahl
  Proceedings of 1996 ACM/IEEE Conference on Supercomputing (SC96). Pittsburgh, PA, USA (1996)

* Computational Steering Software Systems And Strategies
  S. G. Parker, C. R. Johnson, D. Beazley
  Computing in Science and Engineering, p. 50 (1997)

PDF in `AdditionalMaterial` folder.

Additional references

* Modern Software Tools For Scientific Computing
  E. Arge, A. M. Bruaset, H. P. Langtangen (editors)
  Birkhäuser Boston (1997)

* Computational Steering
  R. van Liere, J. D. Mulder, J. J. van Wijk
  Future Generation Computer Systems, vol. 12, p. 441 (1997)

* High Performance Computational Steering Of Physics Simulations
  J. Vetter, K. Schwan
  Proceedings of 11th International Parallel Processing Symposium, p. 128. Genva, Switzerland (1997)

PDF in `AdditionalMaterial` folder.

Additional references

* A Survey Of Computational Steering Environments
  J. D. Mulder, J. J. van Wijk, R. van Liere
  Future Generation Computer Systems, vol. 15, p. 119 (1999)

* Computational Steering Library And Toolkit

* Workflow Management Systems

  Anduril | BioBIKE | Bioclipse | Galaxy | Kepler | Orange | zymake

* Twitter

  @Ames_Laboratory | @Argonne | @BerkeleyLab | @BrookhavenLab
  @DOEScience | @Energy | @FermiLab | @INL | @ISS_CASIS | @JBLab
  @Livermore_Lab | @LosAlamosNatLab | @NCSAatIllinois | @ORNL
  @PNNLab | @PPPLab | @SandiaLabs | @SLACLab

Programming Etiquette

Coding a reflection of one's logic ... for the most part

http://dilbert.com/strip/2007-11-26/

Problem definition

* Understand what is given
* Understand what is expected
* Understand what you have and what you don't have for resources

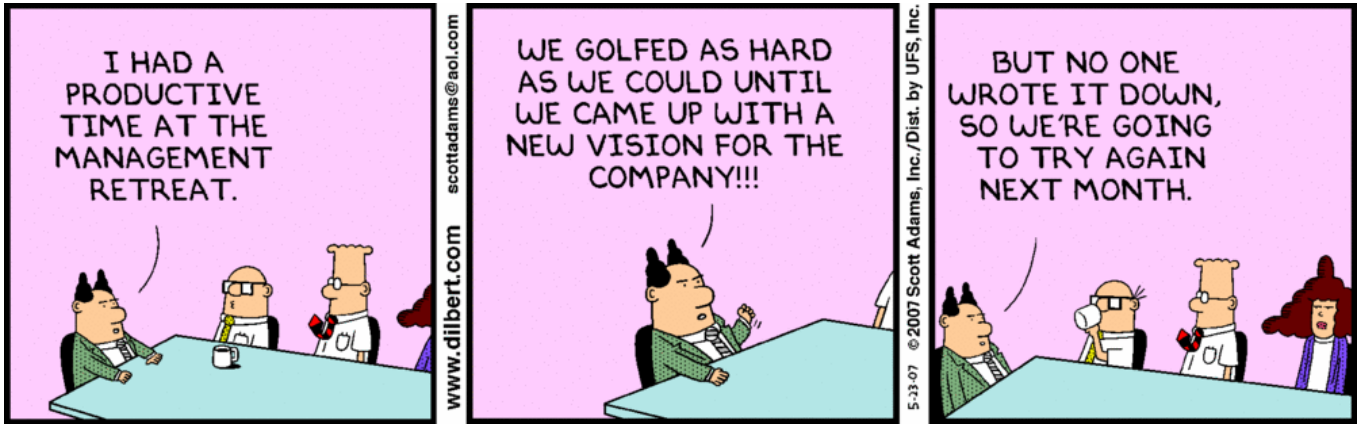
Literature search

* Your own personal and/or friends'/colleagues' collection
* Computing literature

  GitHub | Stack Overflow
* Object-oriented learning

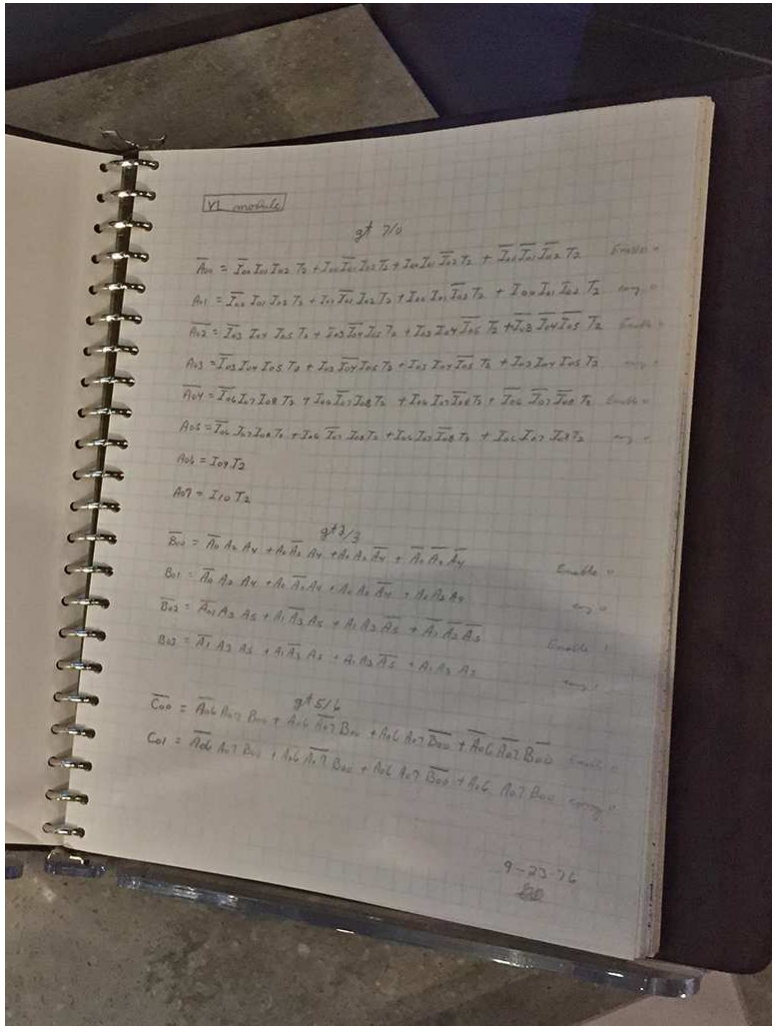
Notes

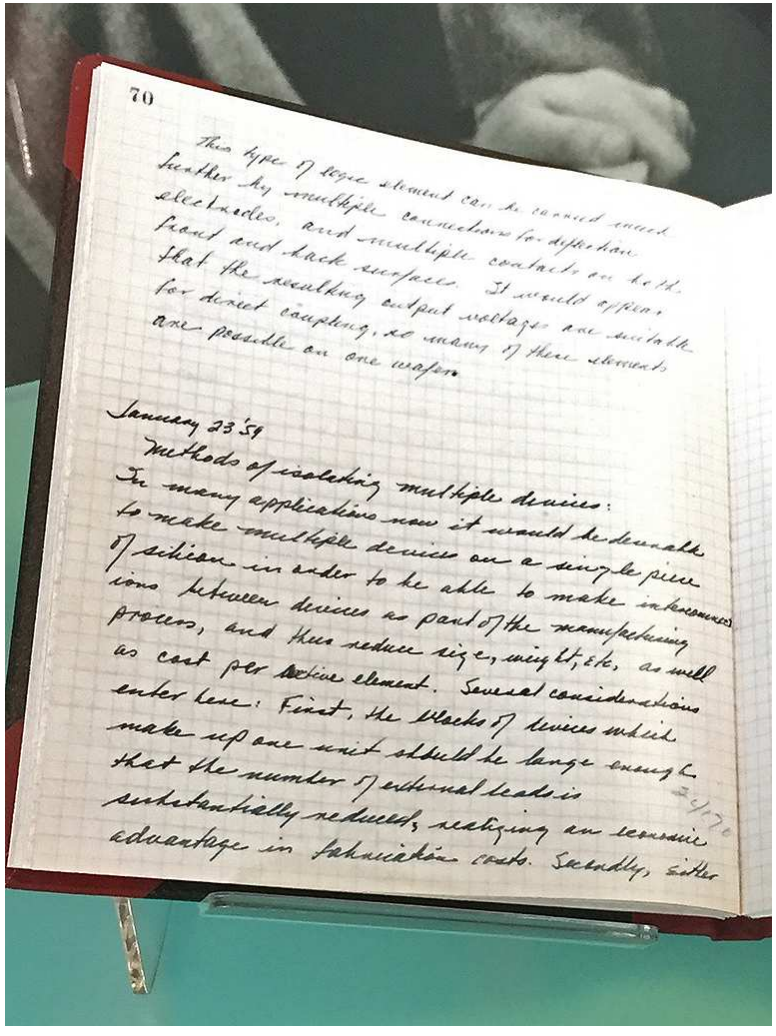
* As detailed as possible
* Include date, time, and location
* Include hostname, and version of OS, software, compilers
* Hand-written, not just an electronic version

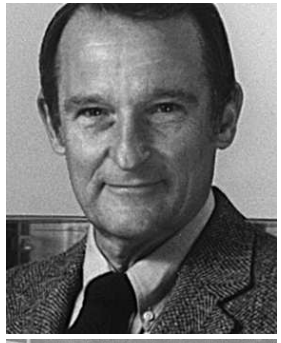


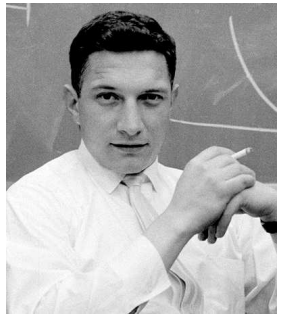
http://dilbert.com/strip/2007-05-23/

Notes





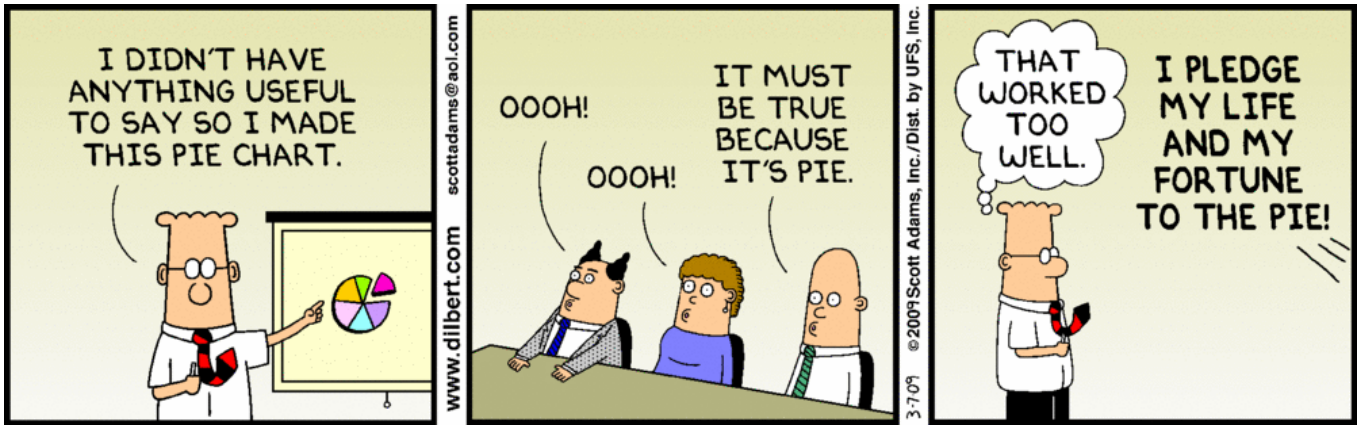




Seymour Roger Cray (1925 – 1996): American electrical engineer, entrepreneur; founder, Cray Research
Robert Norton Noyce (1927 – 1990): American physicist and entrepreneur; co-founder, Fairchild Semiconductor and Intel

Imagery

* Use schematics/plots - a picture is still worth a thousand words



http://dilbert.com/strip/2009-03-07/

Language of choice

* Learn more than one if you can
* Know what best fits your research needs
* Realize that not every language does everything well

Examples: Scripting (BASH, PERL, Python), programming (C/C++, FORTRAN, Java, Julia, Mathematica, MATLAB), documentation (LaTeX), database (SQL, Oracle), web design (CSS, HTML, PHP)

Appearance Code is poetry

When you get what you want in your struggle for self
And the world makes you king for a day
Just go to the mirror and look at yourself
And see what that man has to say.

For it isn't your father, or mother, or wife
Whose judgment upon you must pass
The fellow whose verdict counts most in your life
Is the one staring back from the glass.

He's the fellow to please – never mind all the rest
For he's with you, clear to the end
And you've passed your most difficult, dangerous test
If the man in the glass is your friend.

You may fool the whole world down the pathway of years
And get pats on the back as you pass
But your final reward will be heartache and tears
If you've cheated the man in the glass.

The Man In The Glass by Peter Dale Wimbrow Sr., 1934

Appearance

* A sense of *love at first sight*
* Must look pretty, have a good and logical flow, and be useful
* Statements and modules in top-down/alphabetical order

Write for people, and not for computers

As far as the course work assignments are concerned, readability of the code takes a much much higher precedence over its efficiency as long as it produces meaningful, reliable and reproducible result.

Extra emphasis on efficiency may be given to *do at home* exercices and/or the term (research) project.

Appearance

```
#include <stdio.h>
main(t,_,a)
char *a;
{return!0<t?t<3?main(-79,-13,a+main(-87,1-_,
main(-86, 0, a+1 )+a)):1,t<_?main(t+1, _, a ):3,main ( -94, -27+t, a
)&&t == 2 ?_<13 ?main ( 2, _+1, "%s %d %d\n" ):9:16:t<0?t<-72?main(_,
t,"@n'+,#'/*{}w+/w#cdnr/+,{}r/*de}+,/*{*+,/w{%+,/w#q#n+,/#{l,+,/n{n+\
,/+#n+,/#;#q#n+,/+k#;*+,/'r :'d*'3,}{w+K w'K:'+}e#';dq#'l q#'+d'K#!/\
+k#;q#'r}eKK#}w'r}eKK{nl]'/#;#q#n'){)#}w'){){nl]'/+#n';d}rw' i;# ){n\
l]!/n{n#'; r{#w'r nc{nl]'/#{l,+'K {rw' iK{;[{nl]'/w#q#\
n'wk nw' iwk{KK{nl]!/w{%'l##w#' i; :{nl]'/*{q#'ld;r'}{nlwb!/*de}'c \
;;{nl'-{}rw]'/+,}##'*}#nc,',#nw]'/+kd'+e}+;\
#'rdq#w! nr'/ ') }+}{rl#'{n' ')# }'+}##(!!/")
:t<-50?_==*a ?putchar(a[31]):main(-65,_,a+1):main((*a == '/')+t,_,a\
+1 ):0<t?main ( 2, 2 , "%s"):*a=='/'||main(0,main(-61,*a, "!ek;dc \
i@bK'(q)-[w]*%n+r3#l,{}:\nuwloca-O;m .vpbks,fxntdCeghiry"),a+1);}
```

Useful but ~~not pretty~~ ugly

Suppose that you received the above code, saved in a file called `TwelveDaysOfChristmas.c`, around Christmas. How likely are you to compile and run it on your own (or university's) computer?

Appearance

```c
// ForkBomb.c
// C program to demonstrate the fork bomb with memory leak. Compilation takes
// less than one second on most modern hardware running Linux OS with GCC.
//
// Compilation and execution:
// gcc ForkBomb.c -o ForkBomb.x
// ./ForkBomb.x

// Headers
#include <stdlib.h>

// main()
int main() {

  while (1) {
    // Replicate and allocate 8 GB memory
    fork(); // Not a bad call by itself; infinite while loop makes it dangerous
    double *ptr = (double *) malloc(1024 * 1024 * 1024 * sizeof(double));
  }

  // Indicate termination
  return 0;
}
```

Pretty but ~~not useful~~ dangerous

If you love your computer, don't compile/run this code.

Appearance Mostly pretty and somewhat useful

```c
// Factorial.c
//
// Computes factorial(n) where n is an
// integer (>=0) supplied by the user.
// Compilation/Execution takes about
// one second on most modern hardware
// running Linux OS with GCC.
//
// Compilation and execution:
// gcc Factorial.c -lm -o Factorial.x
// ./Factorial.x

// Headers
#include <stdio.h>

// Function declaration
int factorial(int n);

// main()
int main() {

  // Variable declaration/initialization
  int n = 0; // User-supplied number
  int N = 1; // factorial(n)

  // PRINT PROBLEM/PROGRAM STATEMENT

  // Accept user input
  printf("  A non-negative integer: ");
  scanf("%d", &n);

  // VALIDATE USER INPUT

  // Compute factorial and print result
  N = factorial(n);
  printf("  factorial(%d) = %d\n", n, N);

  // Indicate termination
  return 0;
}

// factorial()
int factorial(int n) {

  // Variable declaration/initialization
  int M = 1; // factorial(n)

  // Compute the factorial
  // factorial(0) or factorial(1) is 1
  if (n == 0 || n == 1) {
    M = 1;
  }

  // Recursive approach for n > 1
  if (n > 1) {
   M = n * factorial(n - 1);
  }

  // Return factorial to parent module
  return M;
}
```

Communication

* Meaningful nomenclature and comments

  Variables, arrays, structures and functions

* Documentation with metrics

  OS, architecture, hardware, compiler, versions, compilation and execution instructions, time required to compile/run, input and output requirements

* Revision control system

  Keep a detailed track of development

Just because something pops into our head doesn't mean ...

it should find its way out in writing. Any hint of profanity anywhere in a program will result in an irrevocable F for the final grade. A program with no or no understandable/usable comments will receive no credit.

Modularization Divide n' conquer

* Module/Sub-routine
  * Accomplishes recurring tasks efficiently
  * Reduces program size and makes debugging easier
  * Requires description and comments just like the main program

Augusta Ada King, Countess of Lovelace (1815 – 1852): English mathematician and writer

Modularization Divide n' conquer

```
// sum_loop()
int sum_loop(int N) {

  // A sub-routine to compute the sum of first N integers for a given value of N
  // using a for loop.
  //
  // Usage:
  // sum = sum_loop(N);

  // Variable declaration and initialization
  int i   = 0; // Loop index
  int sum = 0; // Sum of integers from 1 through N

  // Loop method
  for(i = 1; i <= N; i++) {
    sum = sum + i;
  }

  // Return the value of sum to the parent function/module
  return sum;
}
```

Augusta Ada King, Countess of Lovelace (1815 – 1852): English mathematician and writer

Modularization Divide n' conquer

```
// sum_gauss()
int sum_gauss(int N) {

  // A sub-routine to compute the sum of first N integers for a given value of N
  // using Gauss' method.
  //
  // Usage:
  // sum = sum_gauss(N);

  // Variable declaration and initialization
  int sum = 0;

  // Gauss method
  sum = N * (N + 1)/2;

  // Return the value of sum to the parent function/module
  return sum;
}
```

Johann Carl Friedrich Gauss (1777 – 1855): German mathematician
Augusta Ada King, Countess of Lovelace (1815 – 1852): English mathematician and writer

Testing

* Check every line/step, and input/output
* Be a devil's advocate and check for extreme cases
* Does the program do NOTHING when it is supposed to NOTHING?

Unit test

A method by which individual units of source code, sets of one or more program modules together with associated control data, usage procedures, and operating procedures are tested to determine whether they are fit for use. It helps find problems early, facilitates change, simplifies integration, improves documentation, and the code's design.

Testing

Regression test

A type of software testing that seeks to

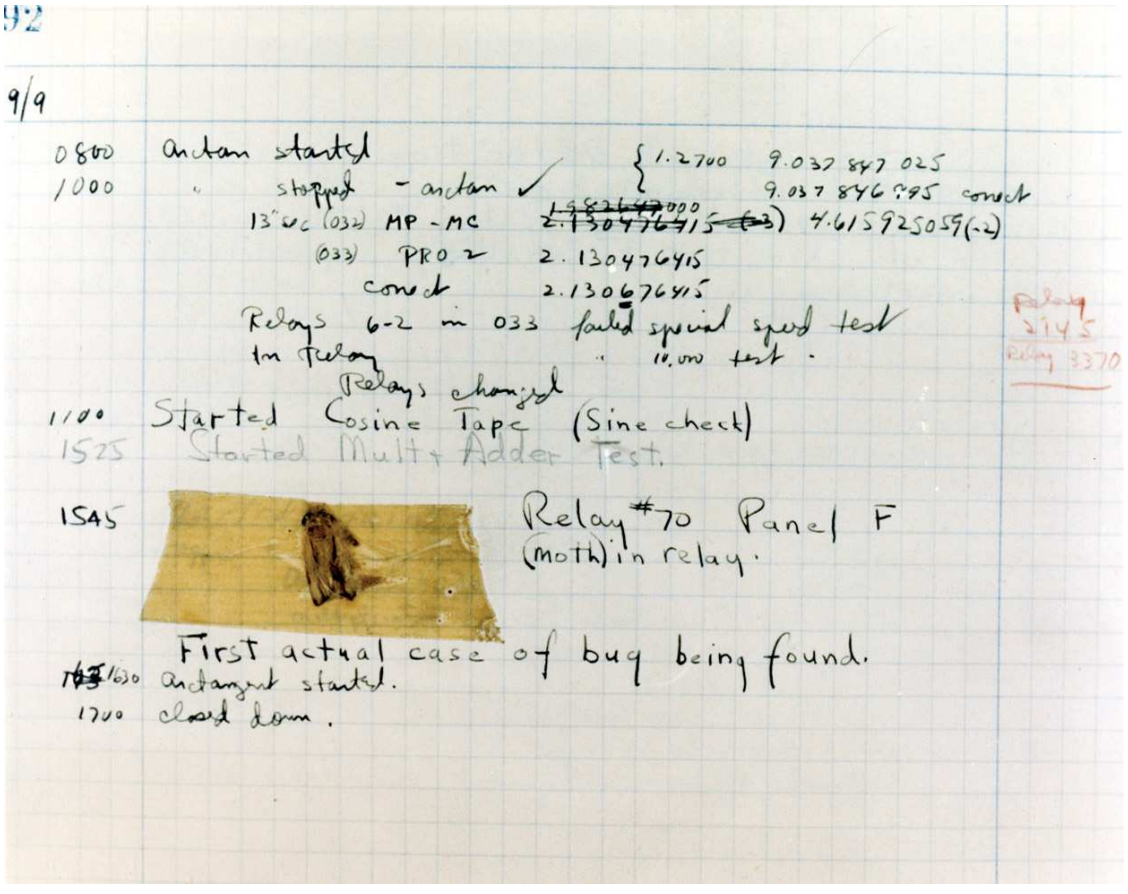
1. uncover new bugs (i.e., regressions) in existing functional and non-functional areas of a system after some changes have been made

2. ensure aforementioned changes have not inadvertently introduced new bugs (or re-introduced previously fixed old bugs), often in a different part of the code

The cause for re-appearance of bugs is often a poor revision control practice (or lack of a formal one, such as Git). The cause for new bugs is often a poor design and/or a fragile fix to a problem (i.e., solution tested for a particular case but not in general).

Debugging

* Identify the bug and understand its solution



Grace Brewster Murray Hopper (1906 – 1992): American computer scientist and US Navy Rear Admiral
She was one of the first programmers of Harvard Mark I (1944), invented the first compiler for programming languages, and popularized the idea of machine-independent programming languages. US Navy guided-missile destroyer, *USS Hopper*, and Cray XE supercomputer at NERSC, *Hopper*, are named in her honor of her achievements.

Debugging

- Angry Spouse Bug
- Bloombug
- Bugfoot
- Common Law Feature
- Defensive Coding
- Heisenbug
- Higgs Bugson
- Hindenbug
- Hydra Code
- Jenga Code
- Loch Ness Monster Bug
- Lorem Ipsum Bug
- Ninja Comments
- Reality 101
- Unicorn
- Yoda Conditions

http://blog.codinghorror.com/new-programming-jargon/

Optimization

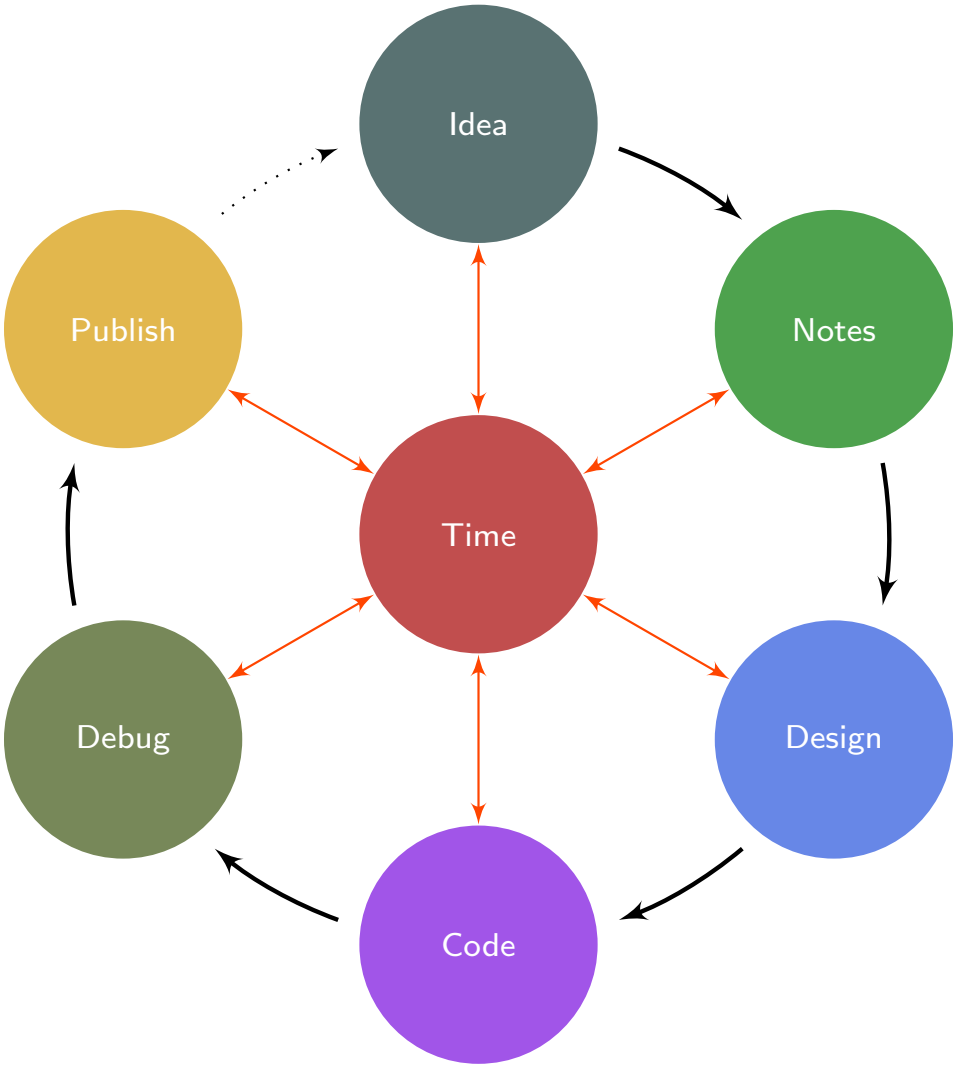
* Modifying the code to run more efficiently
* May be required for the term project
* Not mandatory for the assignments (25-50% slowness is acceptable)

Premature optimization

Act of letting performance considerations affect the code's design.

It is better to design, then code from the design, and then profile or benchmark the resulting code to identify which parts should/can be optimized. A simple and elegant design is often easier to optimize, and profiling may reveal unexpected performance problems that would be hidden behind the curtain of premature optimization.

Code life cycle



Time often isn't very friendly and its unfriendliness grows as the deadline approaches.

Additional references

* The Art Of Computer Programming, vol. 1-4A
  D. E. Knuth; Addison-Wesley (1968, 1969, 1973, 2011)

* The Science Of Debugging
  M. Telles, Y. Hsieh; Coriolis Technology Press (2001)

* The Idea Factory: Bell Labs And The Great Age Of American Innovation
  J. Gertner; Penguin Press (2012)

* The Design Of Everyday Things
  D. Norman; Basic Press (2013)

* Doxygen Official website | GitHub

  Automatic generation of documentation from source code

* Michigan Tech Multiliteracies Center (Walker Arts Building #107)

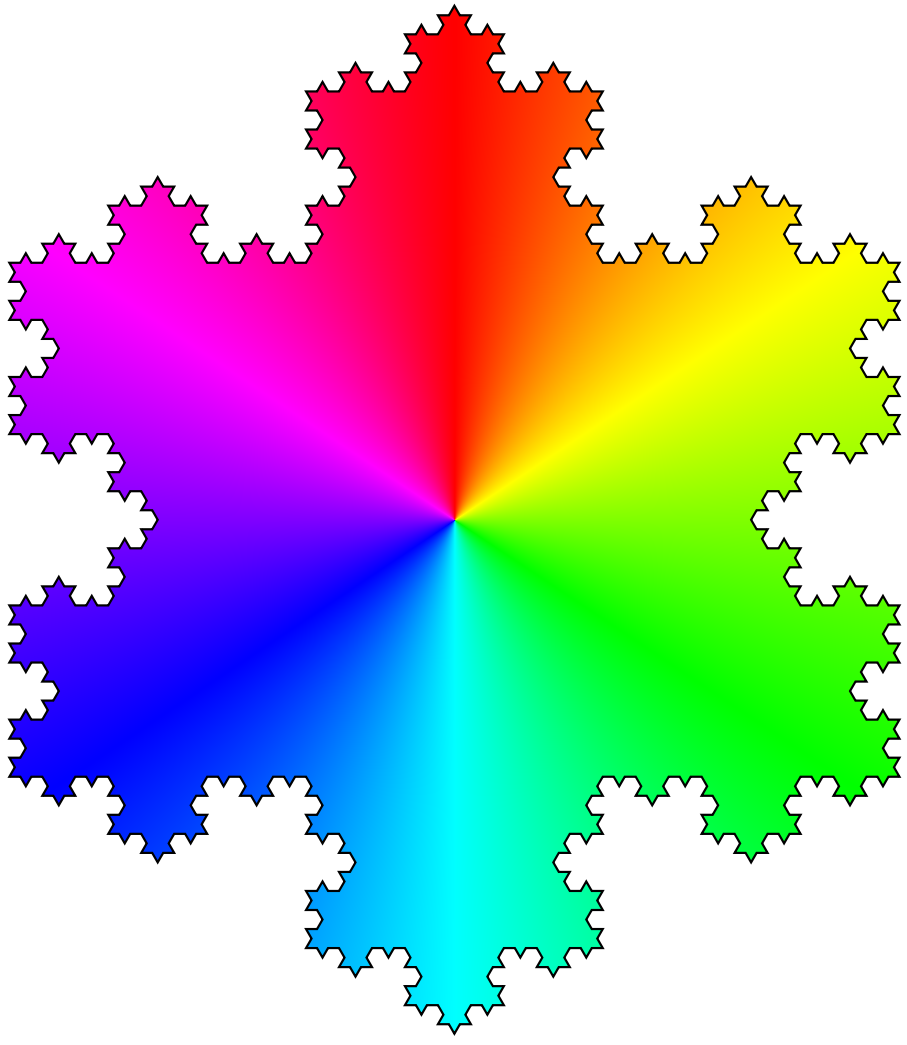
Additional references

* Twitter

  @AdviceToWriters | @Doxygen | @Grammarly | @PurdueWLab
  @WritersDigest | @WritersRelief | @WritingCom | @Writing_Tips

Before we meet again

- Review the syllabus, course material, grade through week #02, notations, active participation, free time exercises, tips, opportunities, mathematical results, and videos
- Make progress in assignment #01
- Think about different ways of seeking help



End of Tuesday lecture.

The art of seeking and citing help

Sarah Lucchesi

Associate Director, Education and Research
J. Robert Van Pelt, and John and Ruanne Opie Library
Library 210 · (906) 487-3379 · slucchesi@mtu.edu

The art of seeking and citing help

* Course guide
* Monday – Thursday 9 am – 7 pm, and Friday 9 am – 5 pm

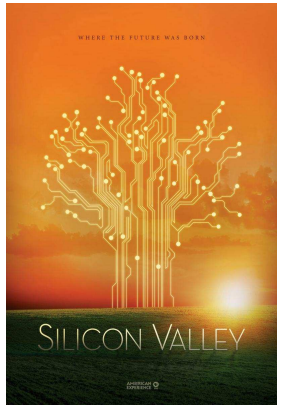
  Library and IT Service Desk
  IM Chat
  (906) 487-2507
* 24/7

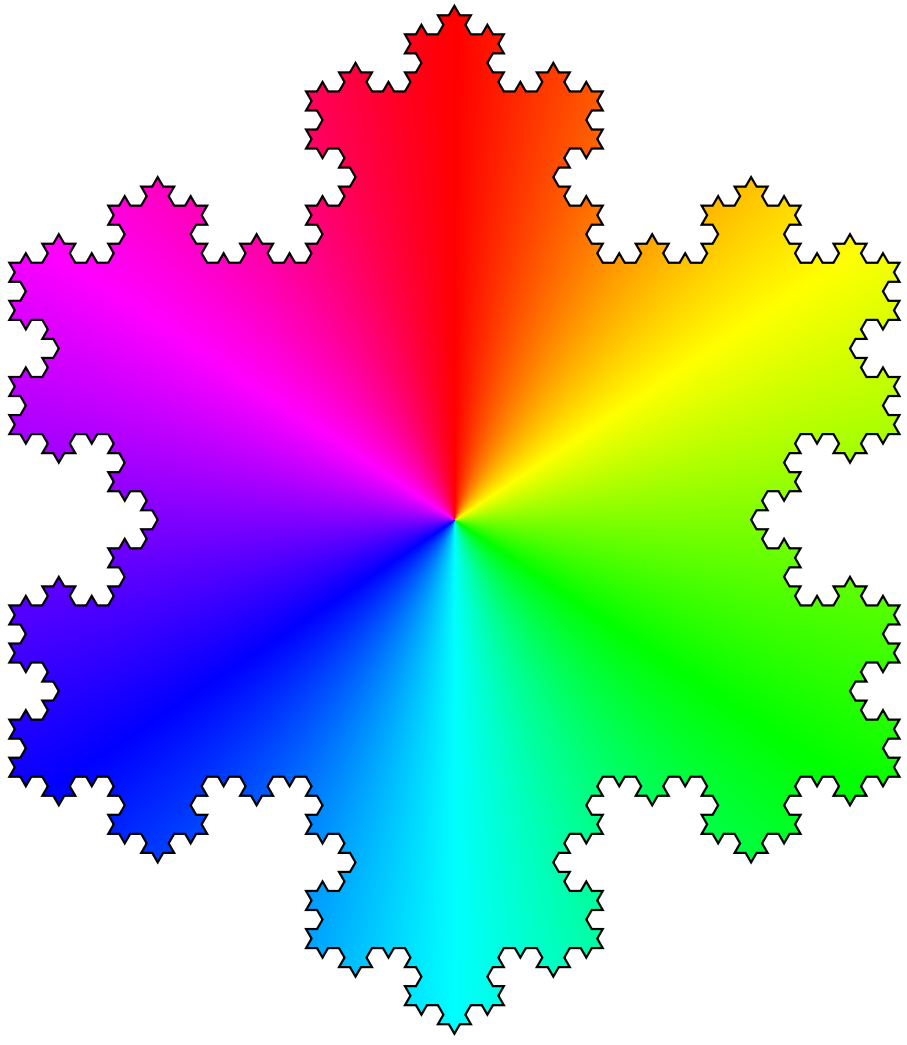
  http://mtu.edu/library/
  reflib@mtu.edu

Before we meet again

- Review the syllabus, course material, grade through week #02, notations, active participation, free time exercises, tips, opportunities, mathematical results, and videos
- Complete assignment #01
- Watch American Experience: Silicon Valley (PBS, video, 2013)
- No class during week #03

  Catch up on life, research, and other courses

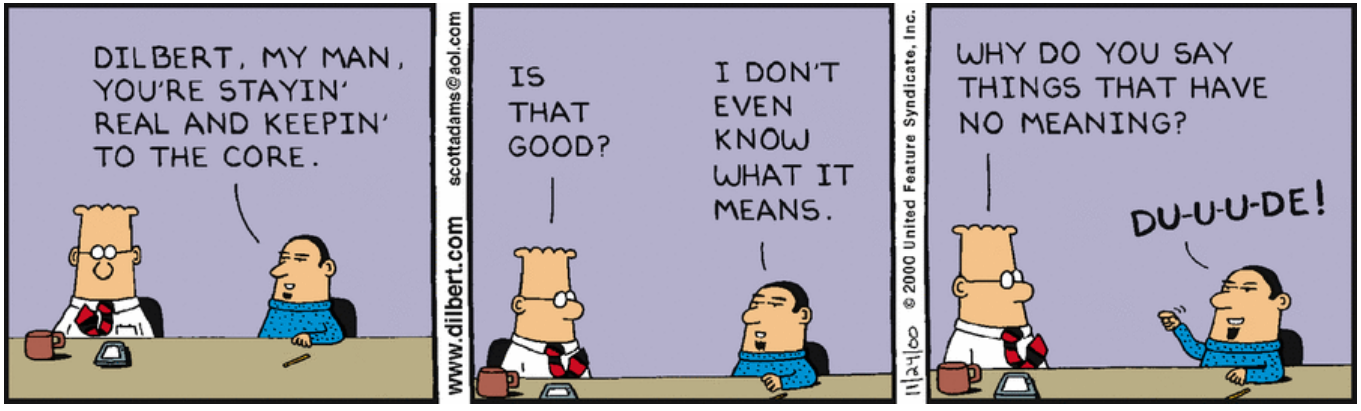




End of Thursday lecture.

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

Lorem here is a continuation of ipsum from there

Do at home and _Back of the envelope_ exercises


Derive/Prove/Guestimate lorem from ipsum

Active participation

Lorem is actively participating in 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 $\times$ Research marketing	02	12
03	PB&J sandwich recipe	02	14
04	Lead the soluton 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.

Research Marketing I Continued

* 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

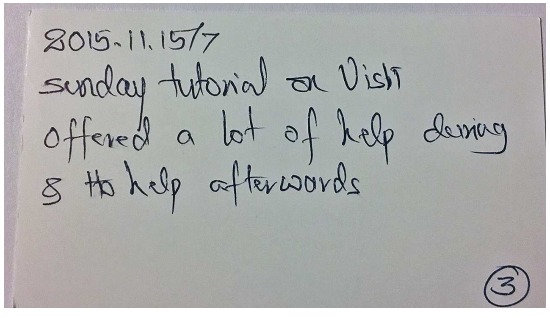
http://dilbert.com/strip/2011-10-07/

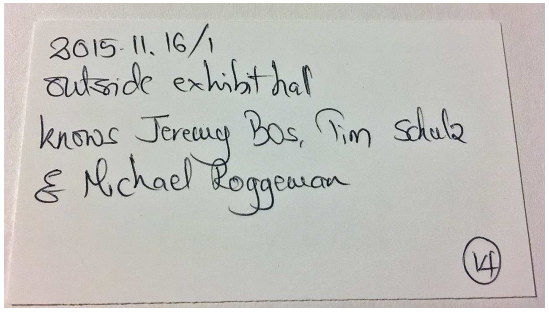
Research Marketing II

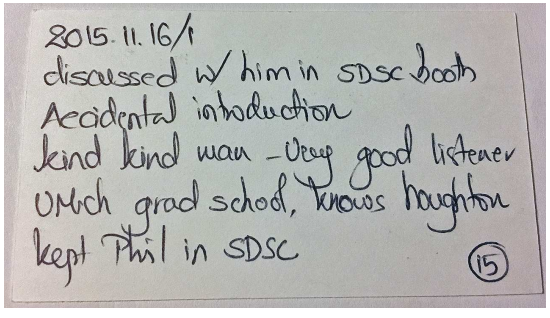
Professional business cards

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.

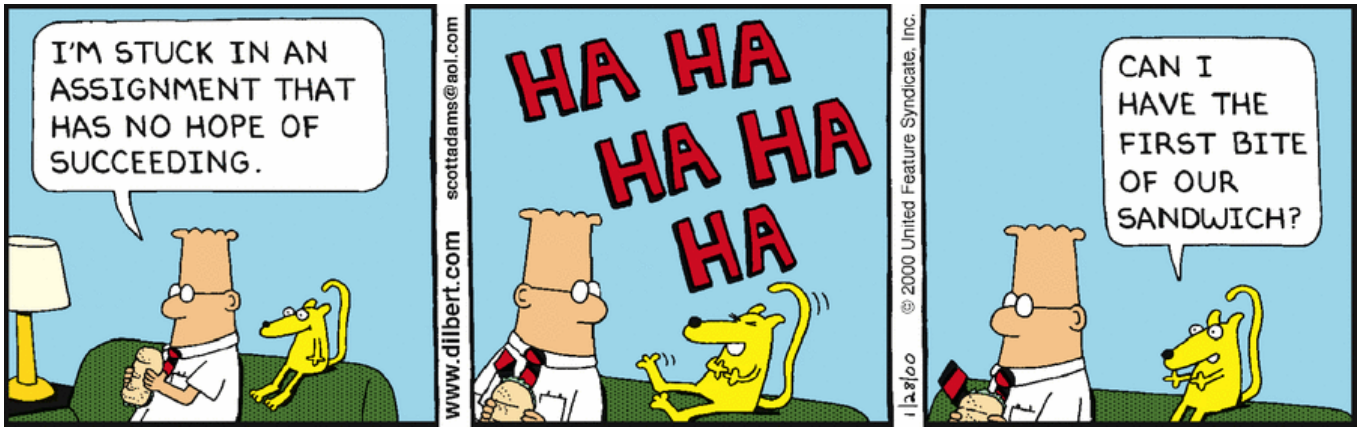






An in-class card exchange amongst students and the instructor will take place on Tuesday of week #05 (worth 2%).

PB&J Sandwich Recipe



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

PB&J sandwich recipe

Submission workflow

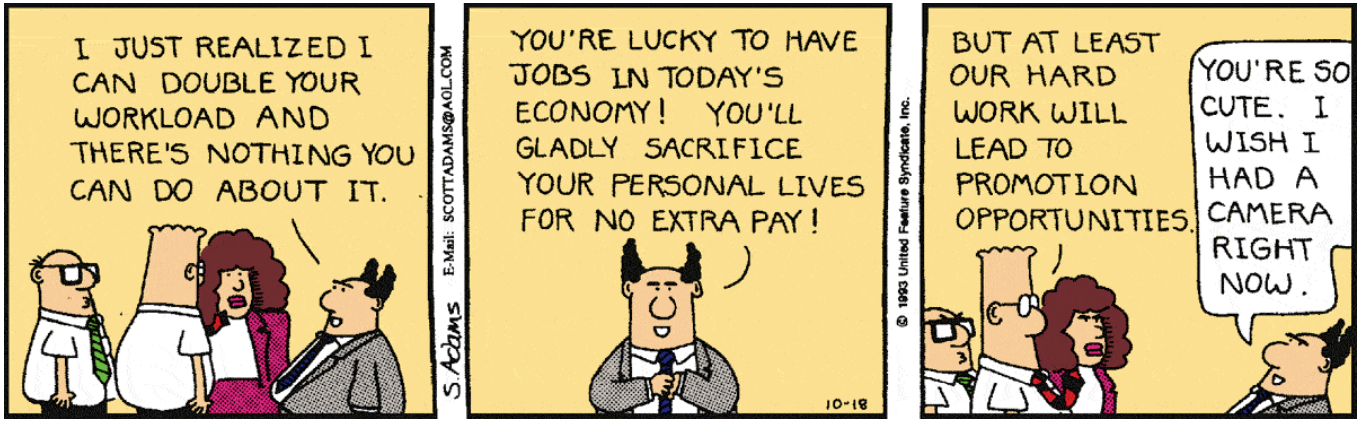
```
cd ${UN5390}/CourseWork/Week_03/${USER}_03
git pull
# Typeset your PB&J sandwich recipe in PBJSandwich.txt
git add PBJSandwich.txt
git commit -m "AP #03:  PBJSandwich.txt"
git push origin master
```

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%).

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 Δt between lightning and thunder

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.

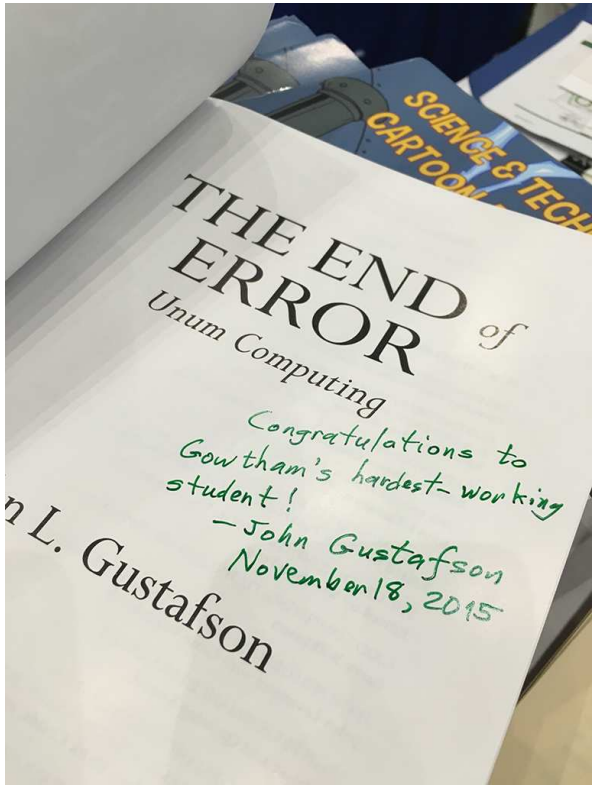
Doing them all

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.

Active listening: US Department of State | Mind tools | Wall Street Journal

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,

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

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,

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

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

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.

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.

LaTeX 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.

LaTeX 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.???
rm -f UN5390.sty UN5390_Settings.tex
```

Replace 01 with the appropriate week number.

LaTeX 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

LaTeX workflow for assignments

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

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.

Doxygen Generate documentation from comments in source code

C/C#/C++/FORTRAN/IDL/Java/PHP/Python, LaTeX, and Doxygen

It supports multiple output formats including 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.

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 Jaszczak
04	Hashim Mahmud	hnalmahm	ME-EM	Gregory Odegard
05	Jeffrey Brookins *	jmbrooki	MSE	Jaroslaw Drelich
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

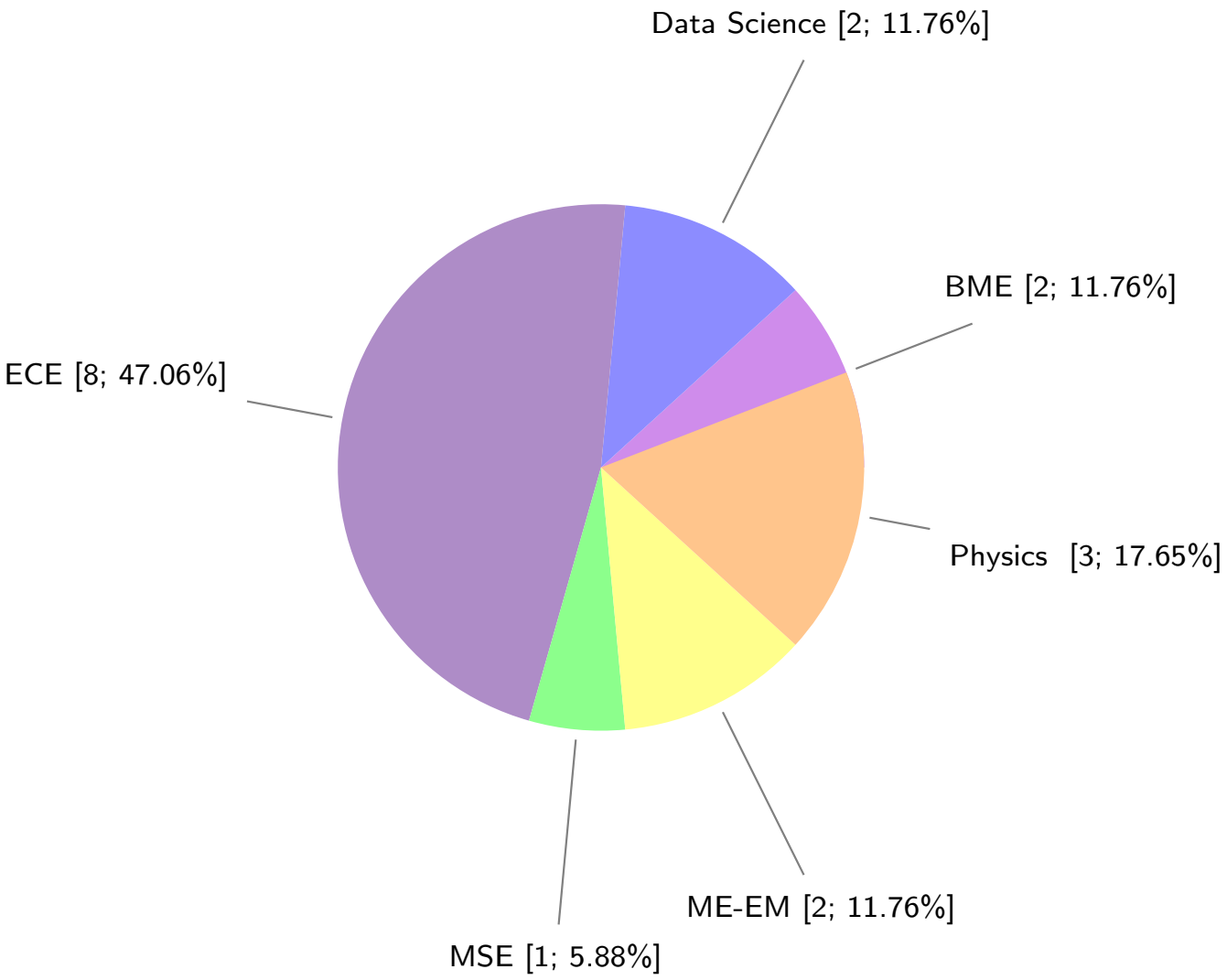
#	Name	Email	Advisor
09	Cal Riutta *	cdriutta	Jinfeng Jiang

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

National Science Foundation | NSF Graduate Research Fellowship Program | Application deadline: 24-28th October 2016

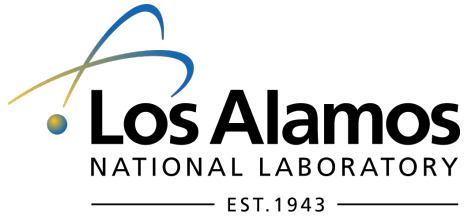
CareerFEST and Career Fair

* 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 LaTeX template in `${UN5390}/LaTeXTemplates/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.

CareerFEST and Career Fair Los Alamos National Laboratory



- 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 (LaTeX)

CareerFEST and Career Fair MIT Lincoln Laboratory

* 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 (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

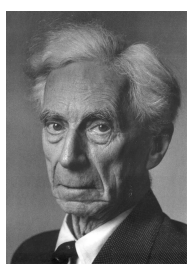
Organized by Keweenaw Climate Community, and sponsored by the local chapter of the American Chemical Society and the Department of Social Sciences at Michigan Tech.

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

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

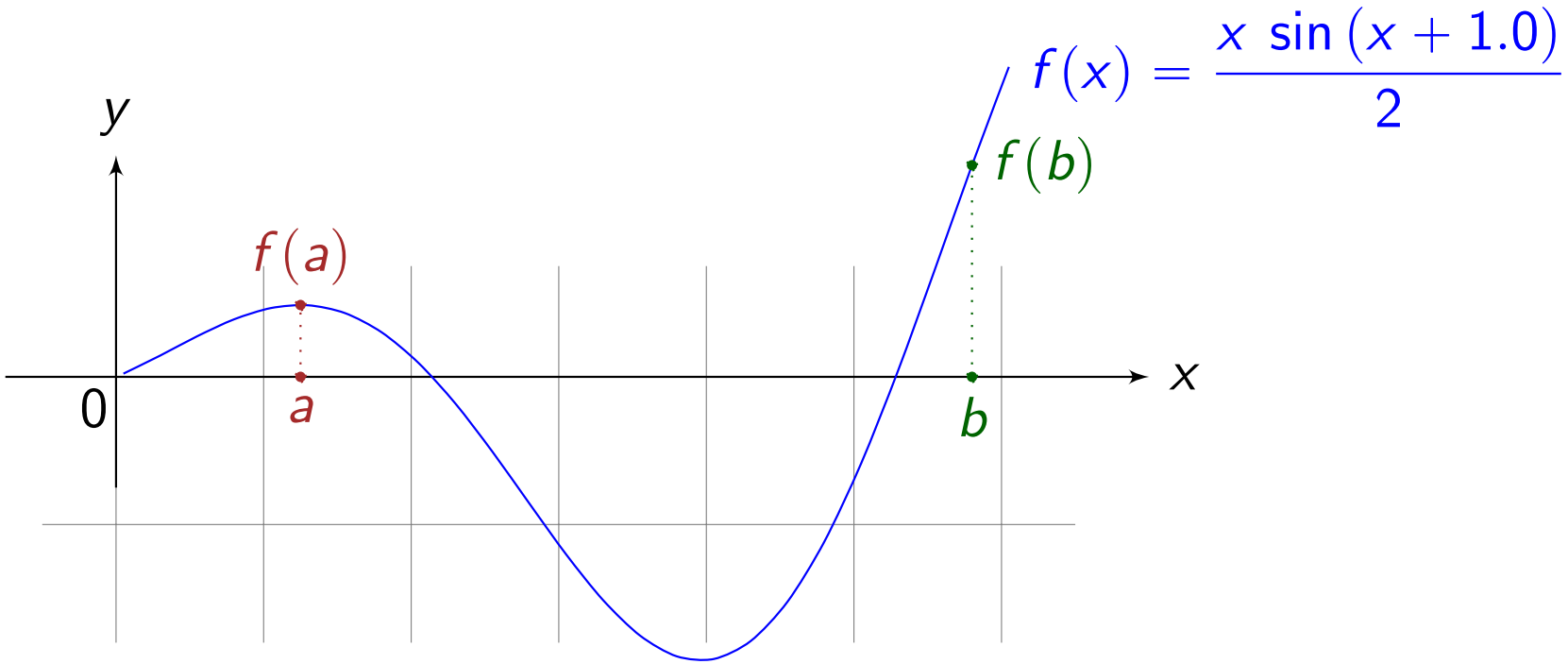
Changing the notations while retaining the underlying essence,

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

Re-arranging the terms,

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

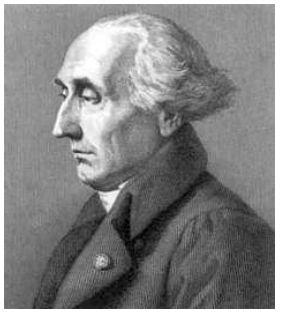
Intermediate value theorem (IVT)

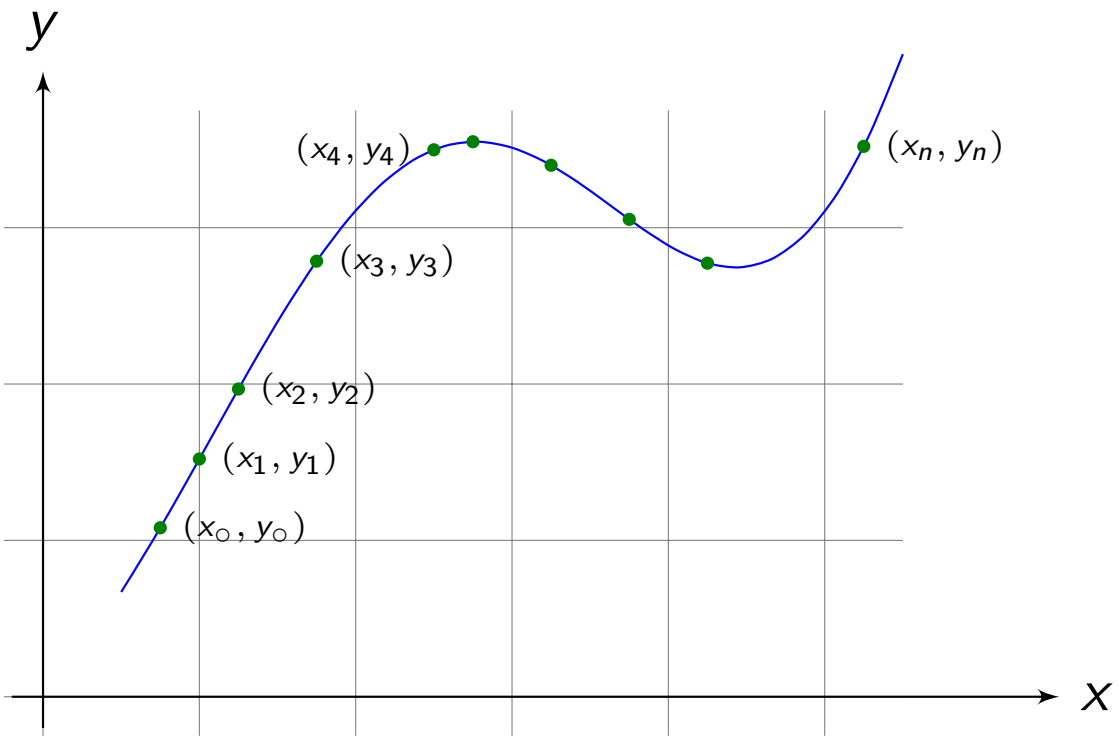


For any function $f(x)$ that is continuous on $[a, b]$, and has values $f(a)$ and $f(b)$ at a and b respectively, then $f(x)$ also takes any value between $f(a)$ and $f(b)$ at some point within the interval.

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

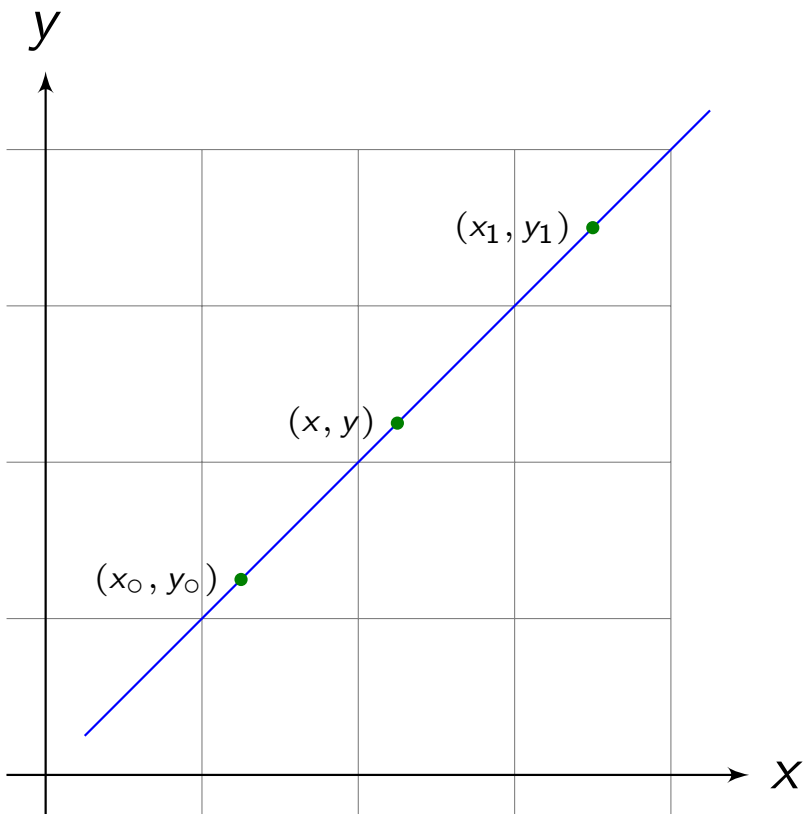
$$l_i(x) = \prod_{\substack{m=0 \\ m \neq i}}^{n} \frac{x - x_m}{x_i - x_m}$$

and are built to have the *Kronecker delta* property

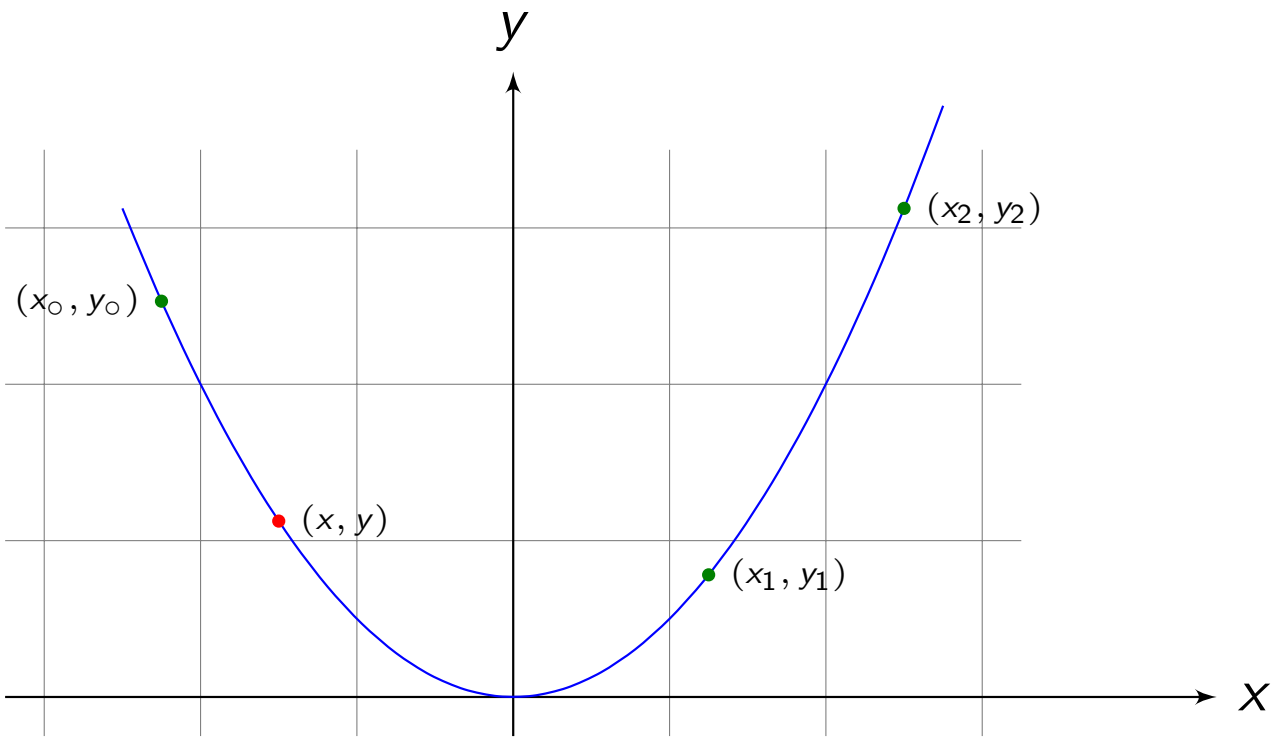
$$l_i(x_j) = \delta_{ij}$$

Lagrange polynomial interpolation Linear

Suppose that $(x_\circ, y_\circ)$ 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



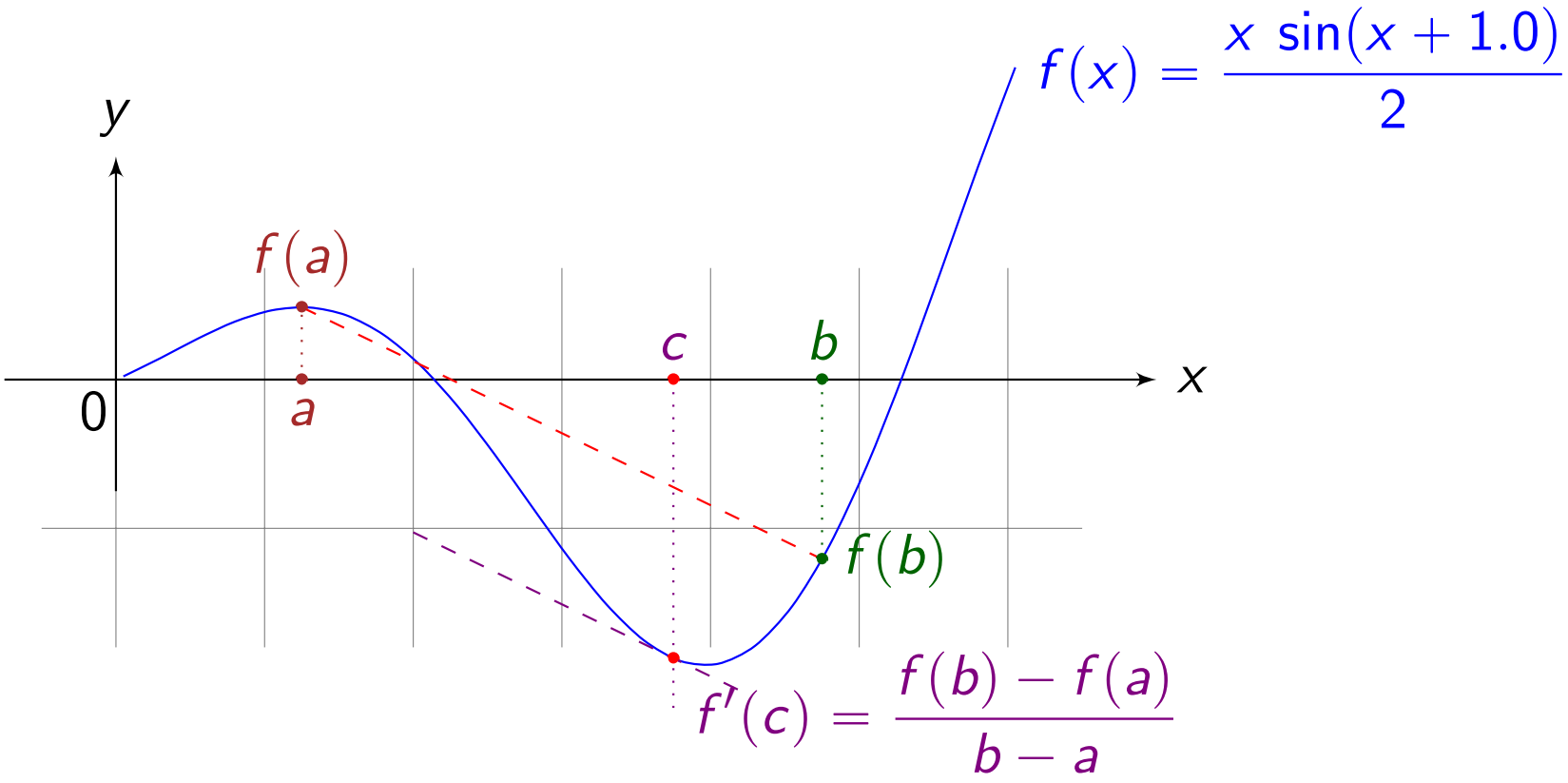
$$\mathcal{L}_2(x) = \frac{(x - x_1)(x - x_2)}{(x_\circ - x_1)(x_\circ - x_2)} y_\circ + \frac{(x - x_\circ)(x - x_2)}{(x_1 - x_\circ)(x_1 - x_2)} y_1 + \frac{(x - x_\circ)(x - x_1)}{(x_2 - x_\circ)(x_2 - x_1)} y_2$$

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, $(i = 0, 1, 2, ..., n)$ 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



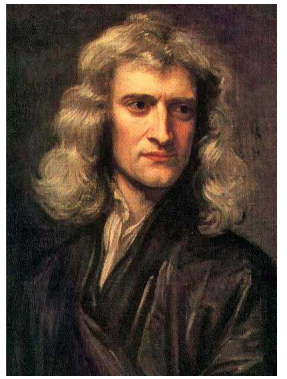
For any function that is continuous on $[a, b]$ and differentiable on (a, b), there exists a point c in (a, b) such that the line joining $f(a)$ and $f(b)$ (i.e., the secant) is parallel to the tangent at c.

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, $g(x) \geqslant 0$, 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

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

then the integral, I, takes the form

$$I = \int_a^b \left[\alpha_\circ + \alpha_1 x + \alpha_2 x^2 + \ldots + \alpha_{n-1} x^{n-1} + \alpha_n x^n\right] dx$$

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 $x_\circ$, then

$$f(x) = \sum_{n=0}^{\infty} \frac{(x - x_\circ)^n}{n!} \frac{d^n}{dx^n} f(x)\Big|_{x=x_\circ}$$

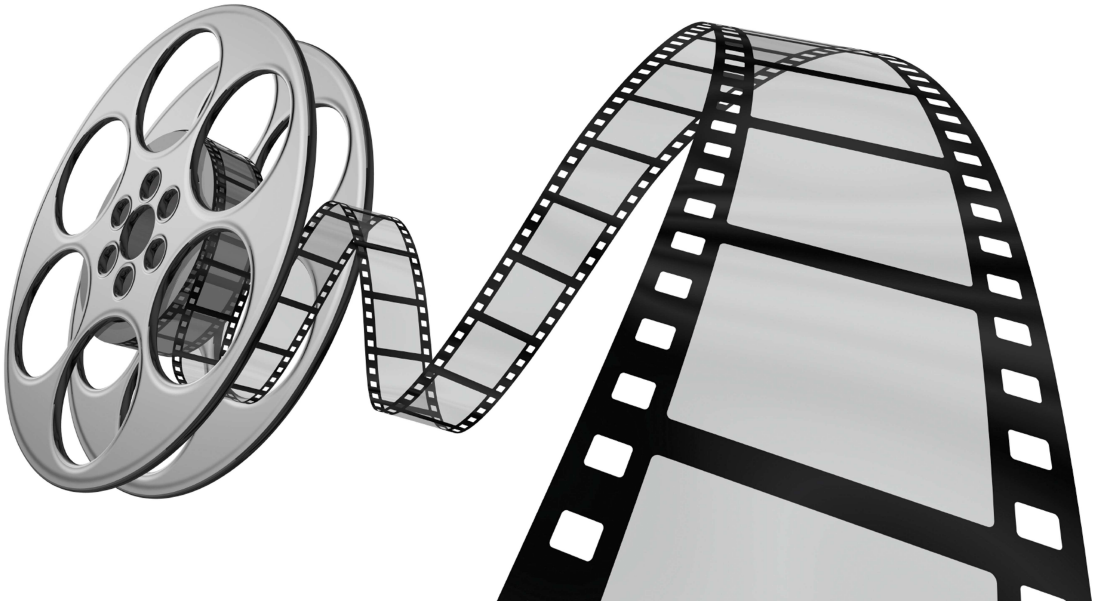
A more general form that clearly identifies the error term is given by the p^{th} order Taylor series expansion of $f(x)$ with $\tilde{x} \in [x, x + \Delta x]$

$$f(x + \Delta x) = \sum_{n=0}^{p} \frac{(\Delta x)^n}{n!} \frac{d^n}{dx^n} f(x) + \frac{(\Delta x)^{p+1}}{(p+1)!} \frac{d^{p+1}}{dx^{p+1}} f(\tilde{x})$$

Brook Taylor (1685 – 1731): English mathematician

Videos

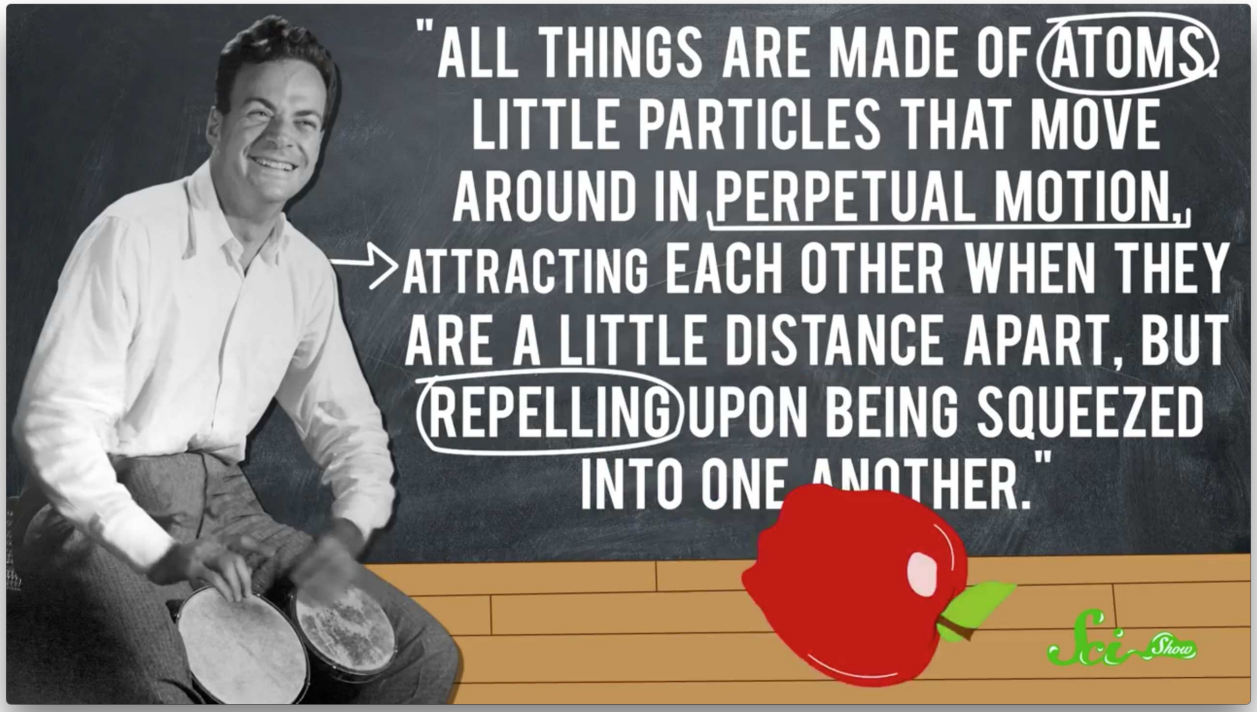
If a picture is worth a thousand words ...



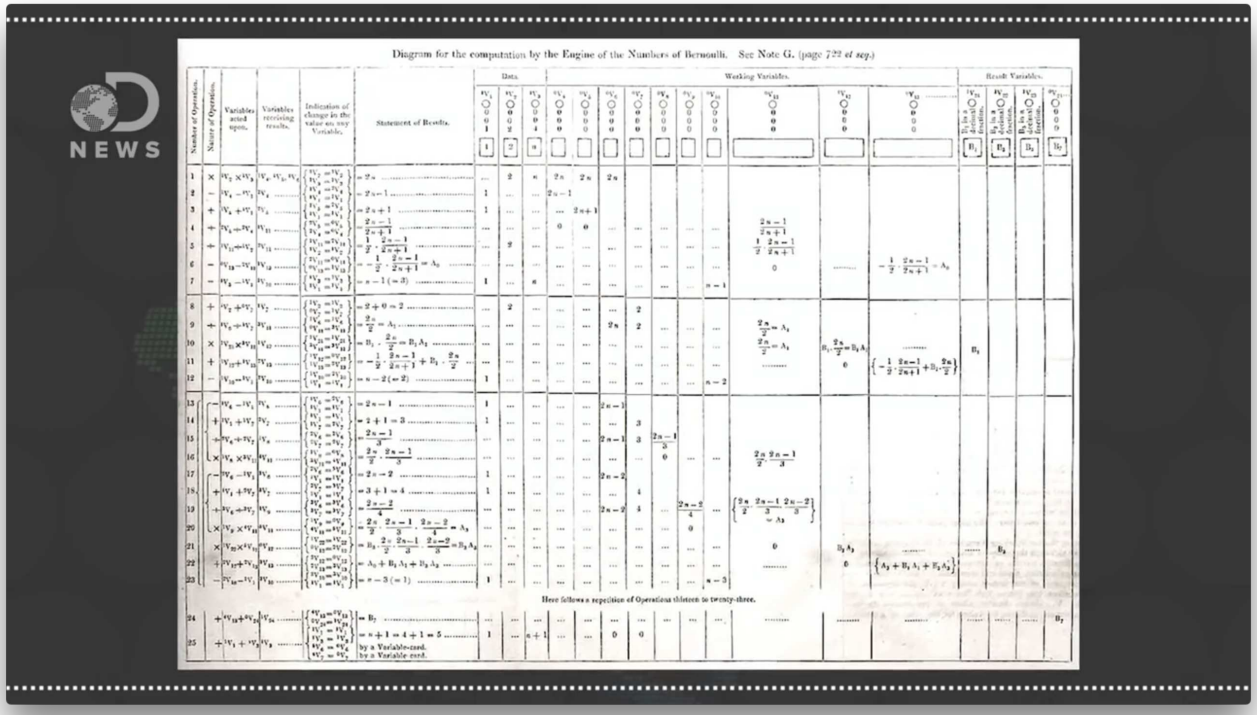
People and Personalities

and their stories

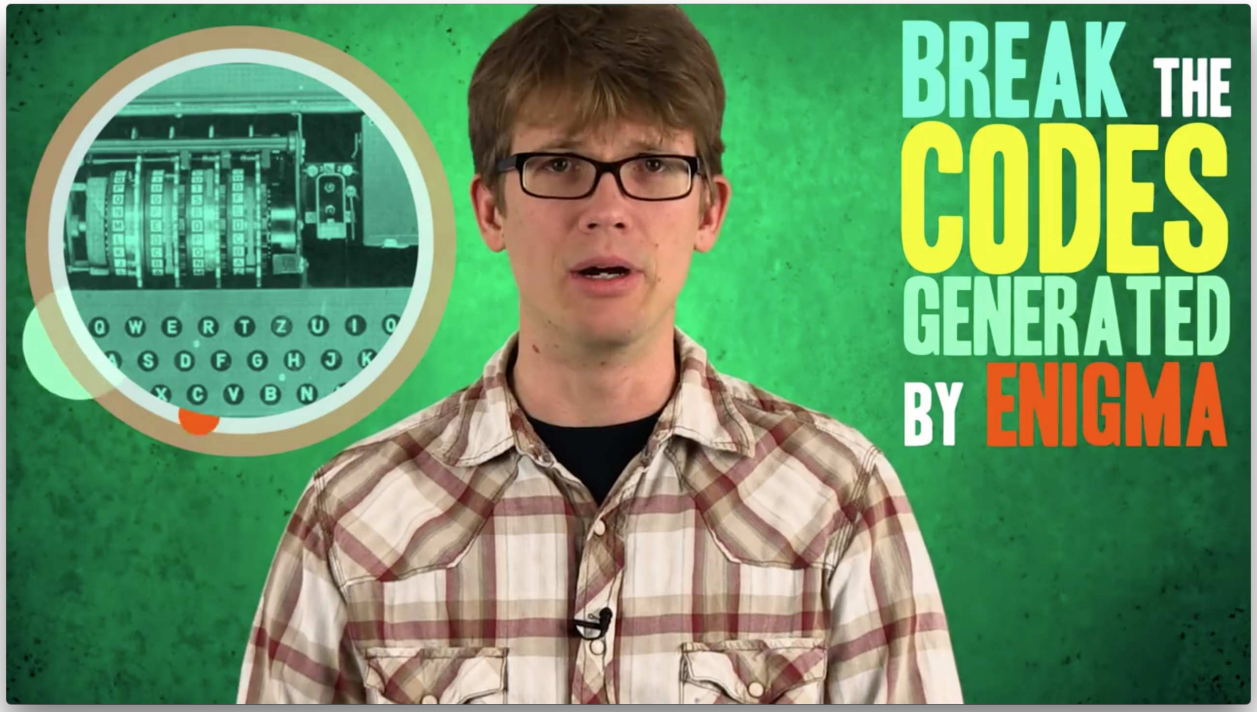
Richard Phillips Feynman 1918 – 1988



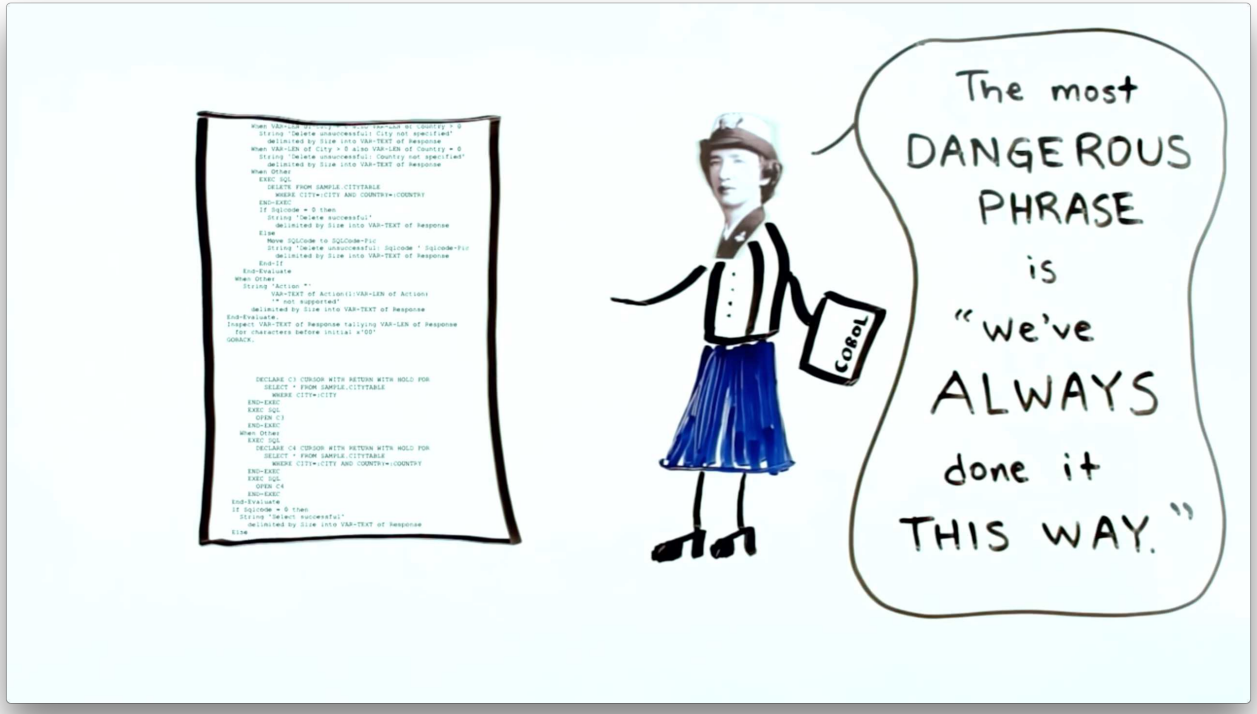
Ada August King, Countess of Lovelace 1815 – 1852



Alan Mathison Turing 1912 – 1954



Grace Brewster Murray Hopper 1906 – 1992



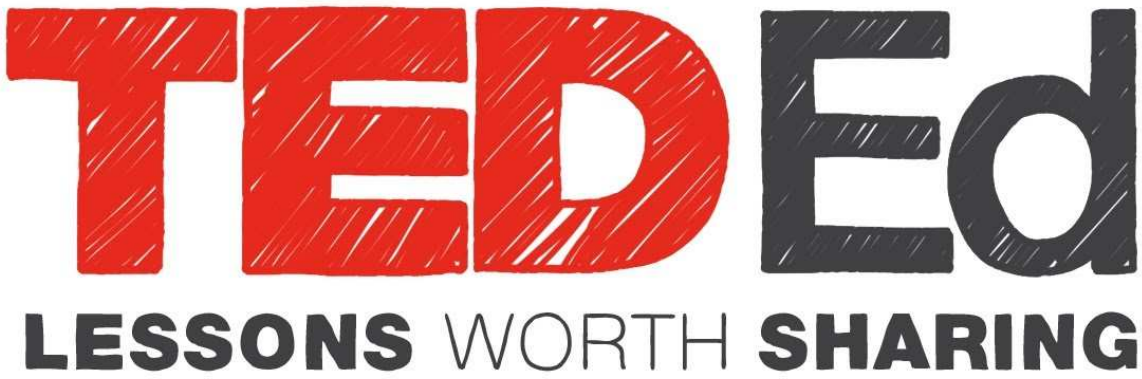
Computer History Museum

1401 N. Shoreline Blvd., Mountain View, CA 94043

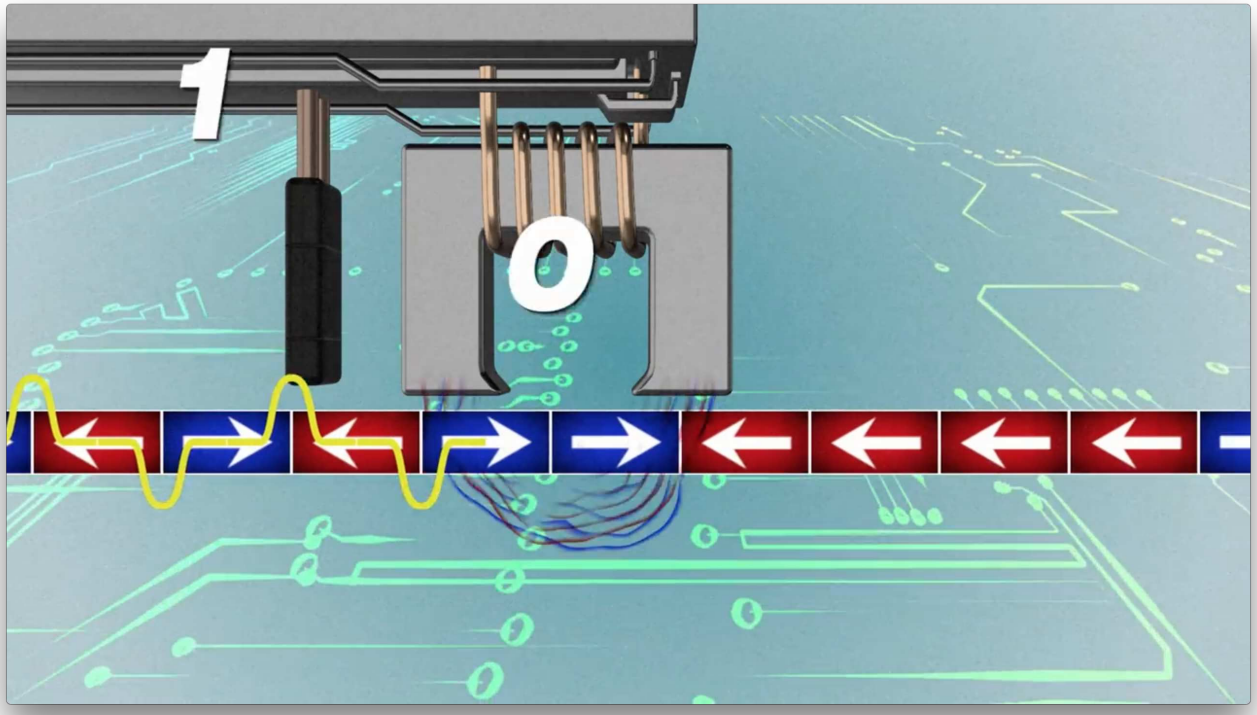
(650) 810-1010

The Fairchild Notes

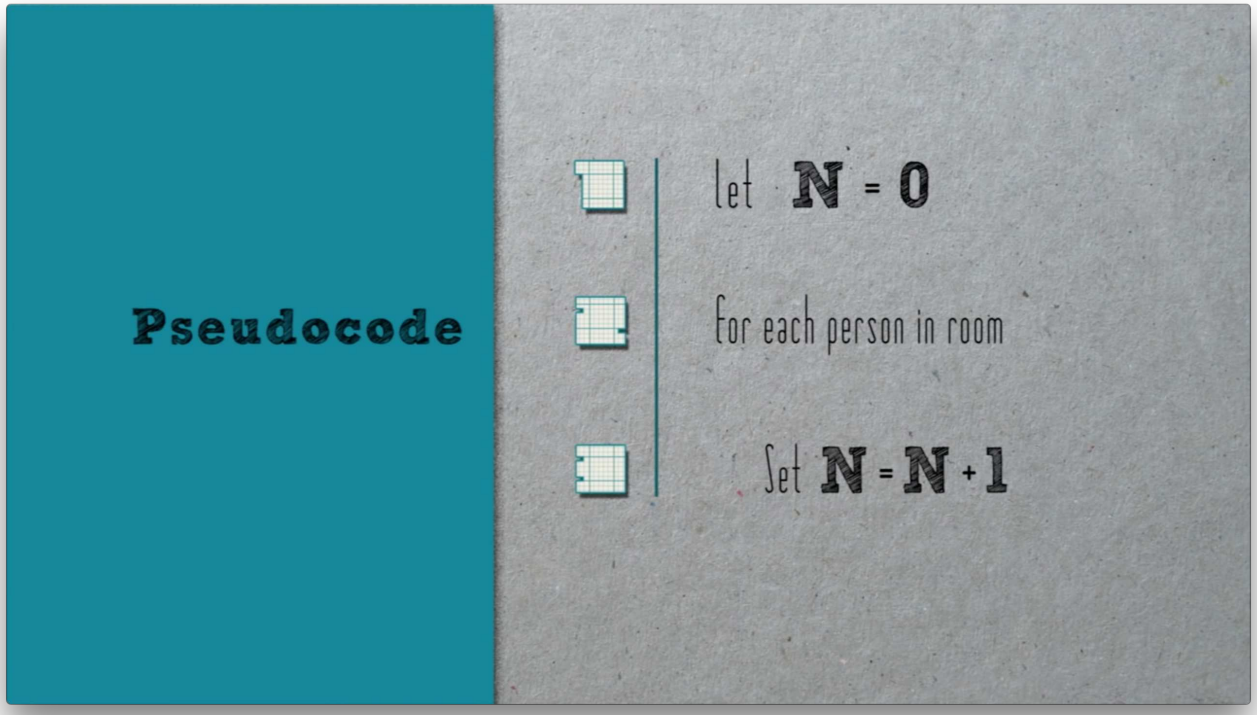
TED Ed



Hard Drives



Algorithm

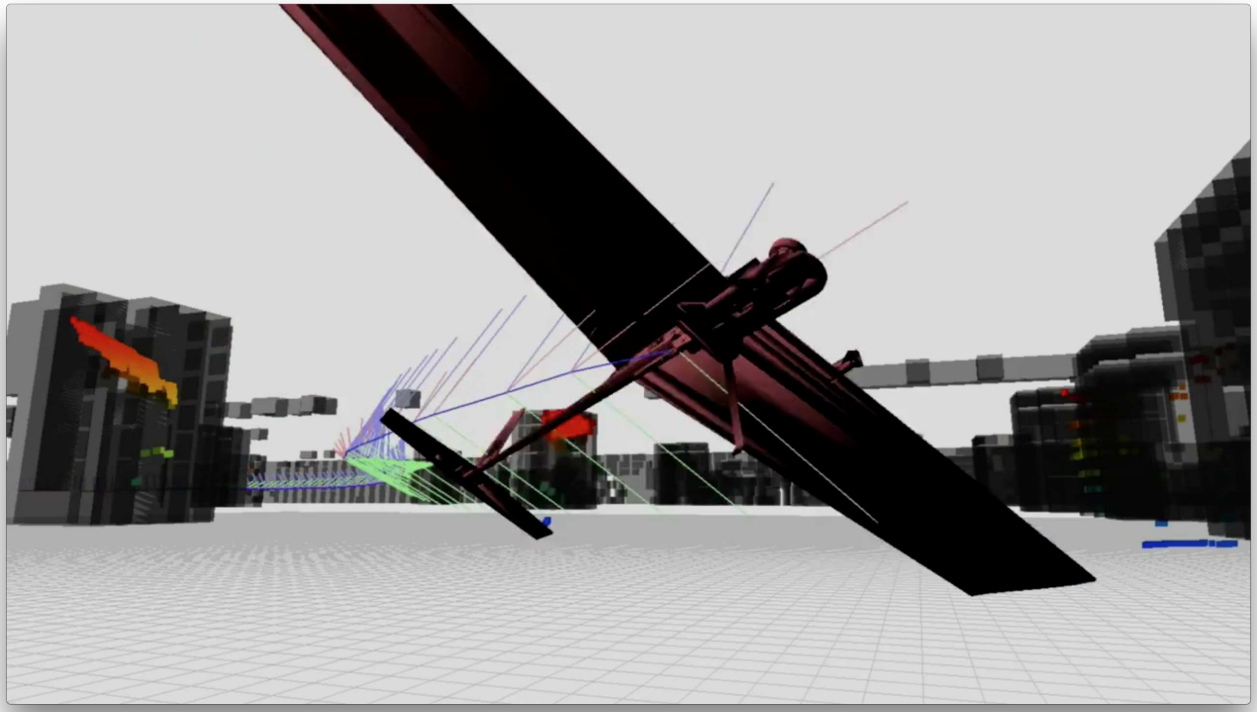


Supercomputing

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

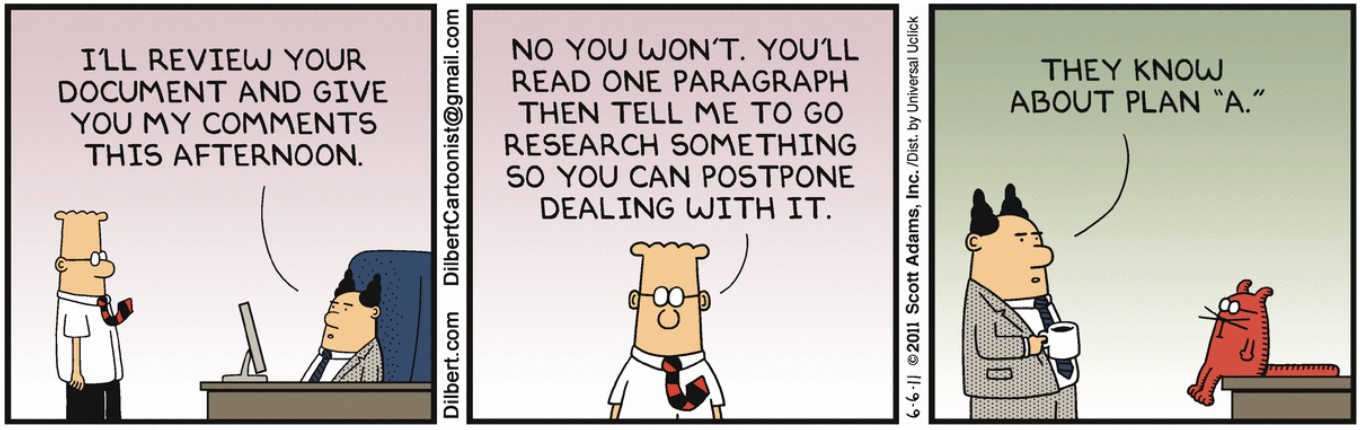
An overview

Aerospace



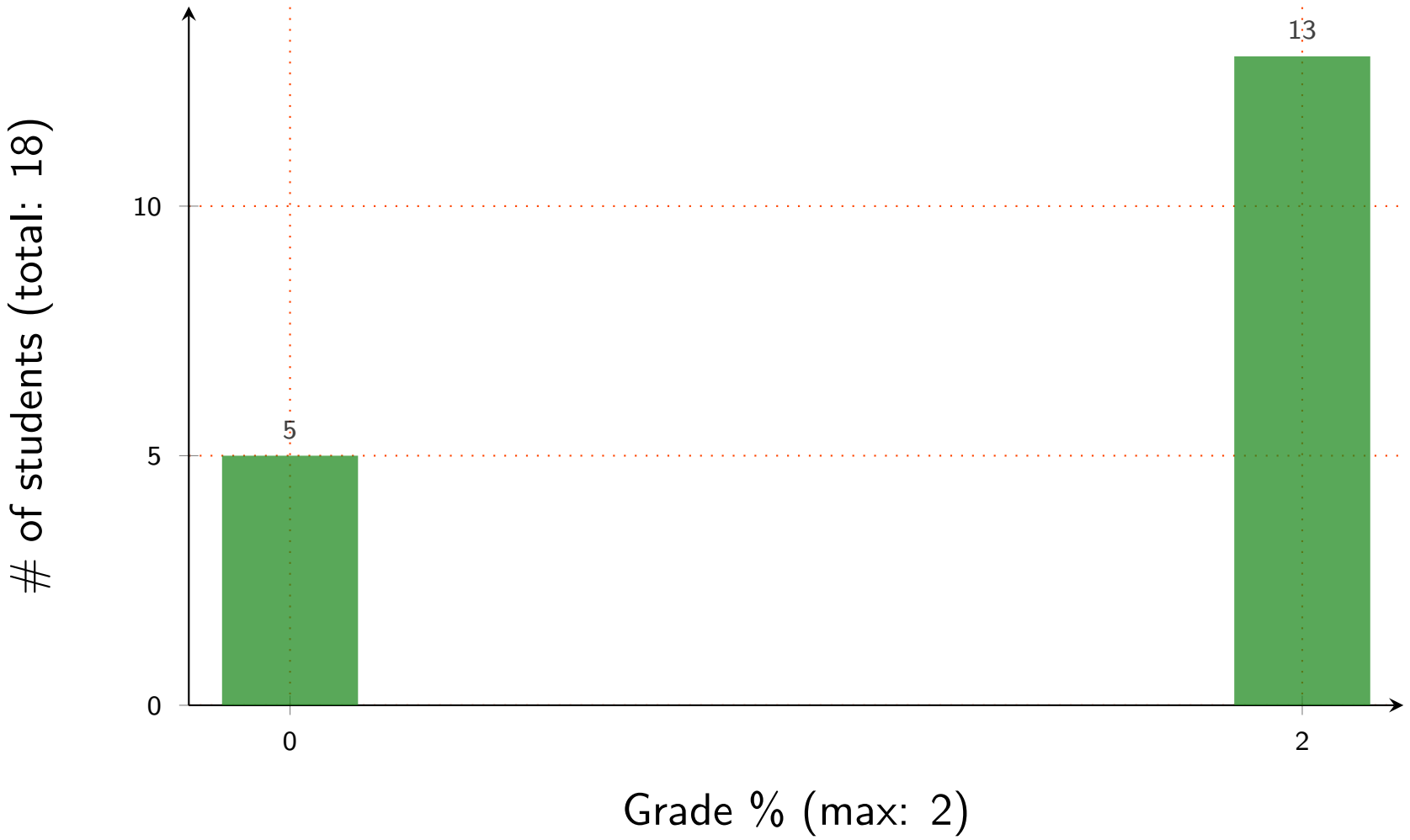
Review of Performance

How well have we been performing?



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

Active Participation #01 Research Marketing I: Twitter



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,

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

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 · @sgowtham