Welcome to Comp 101!

Computational Thinking

1) Course Mechanics
2) Course Objectives
3) Information

I thought this course was called “Computer FITness”
Meet the Crew...

Lectures: Leonard McMillan (SN-258)
          Office Hours: T 2-3

TA:      Nick Dragan (SN-045)
          Office Hours: TBA

Book:    No book... we’ll work from notes only

Laptops: Bring them... there will be in-class exercises
Course Mechanics

Grading:

Best 4 of 5 project/problem sets 40%
2 Quizzes 30%
Final Exam 30%

Project/problem sets will be distributed approximately every two weeks; three weeks around quizzes. Late problem sets are accepted up to 3 class periods after their due date (penalized at 70.71% compounded for every period they are late). The lowest score will be dropped.

I will attempt to make Lecture Notes, Problem Sets, and other course materials available on the web before class on the day they are given/assigned.
Comp 411: Course Website

Announcements, corrections, etc.

On-line copies of all handouts

http://www.unc.edu/courses/2008fall/comp/101/001
What is Comp 101?

Computers for non-CS-majors
CS for non-geeks
If you only take a single course on computers in your entire college career, what should you learn?
Computer FITness??

FIT is an acronym
   (the ness is to be cute I guess)

Fluency in Information Technology
   (Joke)
Foundations of Information Technology
   (Stale)
Fundamentals of Information Technology (aims too low)
Future of Information Technology (more laughs)

Fearless of Information Technology (my hope)
Goal 1: Demystify Computers

Strangely, most people (even some computer scientists) are afraid of computers.

We are only afraid of things we do not understand!

I do not fear computers. I fear the lack of them.

- Isaac Asimov (1920 - 1992)

Fear is the main source of superstition, and one of the main sources of cruelty. To conquer fear is the beginning of wisdom.

- Bertrand Russell (1872 – 1970)
Goal 2: Limits of Computation

Computers are powerful, fast, and getting faster everyday...

BUT, they do have provable limits

We know problems that:

No computer known computer can solve
No known program could solve within our lifetime
(or the lifetime of the universe...
Goal 3: Algorithms Matter

A good algorithm on a slow computer will beat a bad one on a fast computer... eventually if the size of the problem grows

Design matters!

Algorithms are beautiful things!

Like art, you don’t have to do it to appreciate it.
Goal 4: Power of Abstraction

Define a function, develop a robust implementation, and then put a box around it.

Abstraction enables us to create unfathomable systems (including computers).

Why do we need ABSTRACTION…

Imagine a billion --- 1,000,000,000
Understanding systems with >1G components

Personal Computer: Hardware & Software

Circuit Board: ≈8 / system
1-2G devices

Integrated Circuit: ≈8-16 / PCB
.25M-16M devices

Module: ≈8-16 / IC
100K devices

MOSFET

Scheme for representing information

Gate:
≈2-16 / Cell
8 devices

Cell:
≈1K-10K / Module
16-64 devices
Computational Structures

Technologies change...
What are computation’s fundamentals?
What details are essential for understanding?
Let’s Get Real!

What can we expect to learn in this course?

How to program?

1. Sadly, most CS majors are not proficient programmers after 1, 2, 3... courses, how can we expect more?

2. The details of coding can be tedious. Most of software development is dry engineering tasks, and debugging

3. We’ll settle for coding Zen
NO Programming?

Not quite....

You will write ...  
1. A control program for a virtual robot  
2. Programs to automate tasks on your computer  
3. Amazing spreadsheets  
4. Fancy animated webpages  
5. 3D video games
Is Programming Hard?

NO... You already know a lot about programming... we’ll demonstrate that as the class goes on.

The problem is that the semantic gap between people and computers was too large... but it’s shrinking.

Programming is about design and decomposition.

You also probably know some algorithms, and have taught them to others...

- given directions
- shared a recipe
- flowcharted some process...
Do I have to become a Programmer?

I didn’t sign up for this class to become a code-monkey!

I just want to use the programs on my laptop more efficiently.

I want to filter unwanted emails

Reorganize, and index my music, video, and data files

Transfer my data from an old computer to a new one

Get data out of one program and into another
IT-Envy

Do all of your friends have cooler facebook pages than you? Is yours rated the most lame among your friends?

Are you dying to BLOG, but don’t know where to start?

Maybe you’d rather Wiki?

Do you exist without an online presence?
Then Comp 101 is for You?

Computers are here to stay. How will you get along with them?
What is “Computation”?  

**Computation** is about “*processing information*”  

- Transforming *information* from one form to another  
- Deriving new *information* from old  
- Finding *information* associated with a given input  
- “**Computation**” describes the motion of *information* through time  
- “**Communication**” describes the motion of *information* through space
What is “Information”?

information, n. Knowledge communicated or received concerning a particular fact or circumstance. Information resolves uncertainty. Information is simply that which cannot be predicted. The less predictable a message is, the more information it conveys!

“A Computer Scientist’s Definition: Information resolves uncertainty.

It must not be football season...

Tarheels won!

“10 Problem sets, 2 quizzes, and a final!”
Real-World Information

Why do unexpected messages get allocated the biggest headlines?

... because they carry the most information.
What Does A Computer Process?

- Toasters processes bread and bagels
- Blenders processes smoothies and margaritas
- What does a computer process?
- 2 allowable answers:
  - Information
  - Bits
- How does information relate to bits?
Quantifying Information
(Claude Shannon, 1948)

Suppose you’re faced with $N$ equally probable choices, and I give you a fact that narrows it down to $M$ choices. Then you’ve been given:

$$\log_2(N/M)$$ bits of information

Examples:
- information in one coin flip: $\log_2(2/1) = 1$ bit
- roll of a single die: $\log_2(6/1) \approx 2.6$ bits
- outcome of a Football game: 1 bit
  
  (well, actually, “they won” may convey more information than “they lost”…)

Information is measured in bits (binary digits) = number of 0/1’s required to encode choice(s)
Log$_2$?

Calculators have 2 keys for computing logarithms: log$_{10}$ and ln

In this class, we’ll use log$_2$.

You’ll need to recall the following relationship:

$$\log_2 X = \log_k X / \log_k 2 = \log_{10} X / \log_{10} 2 = \ln X / \ln 2$$

e.g. $\log_2 10 = \log_{10} 10 / \log_{10} 2 = 1 / 0.3.. = 3.32..$
Next Time

An Alien Dissection...

More about information....

How it is encoded and processed in computers

How much information is in
- a phone number?
- your name?
- your license plate?
- a dice roll?