

Welcome!

Principles of Computer Science
CPS 1
LSRC B101
M, W, F 1:30-2:20

Professor Jeff Forbes

CPS 001

1.1

Today's topics

- What is this course about?
- How are we going to learn that?
- Who is this guy talking to us?
- Where do we go from here?
- An overview of computer science

- Upcoming
 - > The World Wide Web and HTML
 - > Problem Solving

CPS 001

1.2

Course Information

"A survey of the great ideas of computer science along with experience with programming, the theoretical foundations of computer science, how computer systems are organized and work, and the applications of computers including their effect on society."

● Grading Breakdown

Assessment	Weight (approx)
Labs	10%
Lab Final	5%
In-class	5%
Quiz/Assign	20%
Project	20%
Midterm	15%
Final	25%

- Scores on absolute scale
- *No make-ups, no late submissions*
- Important Dates
 - > Midterm 10/7
 - > Projects due 12/5
 - > Final 12/15 2pm-5
 - > Friday Quizzes
- Let me know ASAP if you have any concerns

CPS 001

1.3

Administrivia

- Check web pages *regularly*
 - > Course web page
<http://www.cs.duke.edu/courses/fall105/cps001/>
 - > Read Blackboard web page
<https://courses.duke.edu>
 - > Read discussion forums
<http://www.cs.duke.edu/phpBB2/index.php?c=54>
- Reading posted on Blackboard page due by Friday class time
- Lab starts on Monday, Sept. 5

- Seating Chart
 - > Pictures of you

- Collaboration policy
- Late policy

CPS 001

1.4

Frequently Asked Questions

- **What is the prerequisite?**
 - > High school algebra (?)
- **How does this course fit into the curriculum?**
 - > A survey, service course designed for non-majors
 - > CPS 4 & 6 are more programming oriented
 - > Satisfies QS and STS requirements
- **Why take this course?**
 - > Computers are interesting, useful, and ubiquitous
 - > Pure entertainment
- **I'm computer-phobic. Will I be able to handle this course?**
 - > Computers *rarely* bite and the ones you deal with in this course are relatively small and not very mobile

CPS 001

1.5

What does this course teach?

- **Syllabus**
- **Why doesn't this course teach anything *practical*?**
 - > Learn how to create a web page?
 - Read a book
 - > Learn to how to create a web browser?
 - Programming paradigms
 - Networking
 - Security
 - Operating systems and computer architecture
 - Concurrent processing
 - Social effects of computing
 - And many more issues involved

CPS 001

1.6

How does this course teach?

- **Active learning**
- **In class, we will use the Personal Response System (PRS)**
 - > Buy at bookstore
 - > ConcepTest: multiple choice questions that highlight an important concept gleaned from the lecture and/or reading
 - > *It's OK not to know!*
 - > Peer instruction: after seeing the results, you will have confer with your neighbor and revote
- **Just in Time Teaching**
 - > Discuss your assignment responses in class
 - > Class debates
- **Why respond?**
 - > Counts toward inclass work score
 - > Effort, participation, and altruism
 - > Makes class better !

CPS 001

1.7

On the subject of questions...

- **Did you ask any good questions today?**
 - > *Ideas and Information* by Nobel prize winning physicist Arno Penzias
 - > Questions which illuminate help nourish ideas
 - > Children are born curious
 - > Fear of public displays of ignorance prevents learning
- **Participate in class**
- **Go to office hours**
- **Make study groups with your classmates**

- **iPods!**

CPS 001

1.8

Getting help

- **Contact Information**
 - > Email: forbes@cs.duke.edu
 - > Phone: 660-6550
 - > Office Hours in D235 Levine Science Research Center
 - T 10:30-12, F 2:45-4:15
 - When my door is open
 - By appointment
- **TA**
 - > Grad TA: Siddhesh Sarvankar (siddhesh@cs)
 - Office hours TBA
 - LSRC D307, 660-4003
 - > Head UTA: TBA
- **And a whole gang of UTAs**

CPS 001

1.9

What is Computer Science?



- What does a computer scientist do?
- What does a programmer do?
- What does a systems administrator do?
- What do you want to do?

CPS 001

1.10

Computer Science and Programming

- **Computer Science is more than programming**
 - > The discipline is called *informatics* in many countries
 - > Elements of both science and engineering
 - > Elements of mathematics, physics, cognitive science, music, art, and many other fields
- **Computer Science is a young discipline**
 - > Fiftieth anniversary in 1997, but closer to forty years of research and development
 - > First graduate program at CMU (then Carnegie Tech) in 1965
- **To some programming is an art, to others a science, to others an engineering discipline**

CPS 001

1.11

Quotations about Computer Science

"Computer science has such intimate relations with so many other subjects that it is hard to see it as a thing unto itself"

- Marvin Minsky, 1979

"It has often been said that a person does not really understand something until he teaches it to someone else"

- Donald Knuth

"Actually, a person does not really understand something until he can teach it to a computer"

- Judith Gal-Ezer and David Harel

CPS 001

1.12

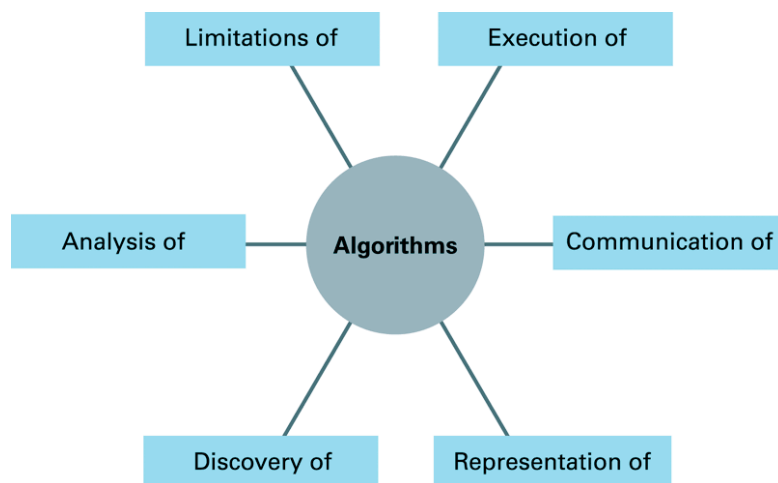
Precise, logical thinking

- Breaking down a task into *unambiguous* steps
- Computers are *deterministic*
- **Algorithm:** *a set of steps that defines how a task is performed*
- **Debugging**
 - > Programs will rarely work the first time one writes them
 - > Systematic approach to detecting, diagnosing, and fixing errors
 - > Debugging skills are useful in many parts of life
- **How many studentss are there in this class?**
 - > Design an algorithm

The Ice Cream Story

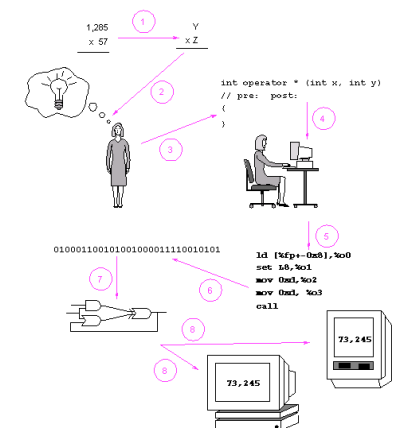
"This is the second time I have written you, and I don't blame you for not answering me, because I kind of sounded crazy, but it is a fact that we have a tradition in our family of ice cream for dessert after dinner each night. But the kind of ice cream varies so, every night, after we've eaten, the whole family votes on which kind of ice cream we should have and Drive down to the store to get it. It's also a fact that I recently purchased a new car and since then my trips to the store have created a problem. You see, every time I buy vanilla ice cream, when I start back from the store my car won't start. If I get any other kind of ice cream, the car starts just fine. I want you to know I'm serious about this question, no matter how silly it sounds: 'What is there about your car that makes it not start when I get vanilla ice cream, and easy to start whenever get any other kind?'"

Algorithms as Cornerstone of CS



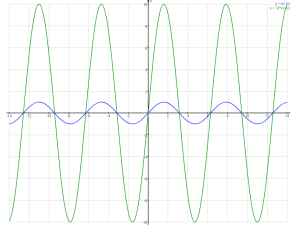
Creating a Program

- **Specify the problem**
 - > remove ambiguities
 - > identify constraints
- **Develop algorithms, design classes, design software architecture**
- **Implement program**
 - > revisit design
 - > test, code, debug
 - > revisit design
- **Documentation, testing, maintenance of program**
- **A programming language is a way to describe an algorithm.**



Representation of information

- What's the difference between
 - > Rolex and Timex?
 - > VCR tape and DVD?
- What is *digital*?
- Sampling analog music for CD's
 - > $44,100 \text{ samples/channel/second} * 2 \text{ channels} * 2 \text{ bytes/sample} * 74 \text{ minutes} * 60 \text{ seconds/minute} = 783 \text{ million bytes}$
- How does MP3 help?

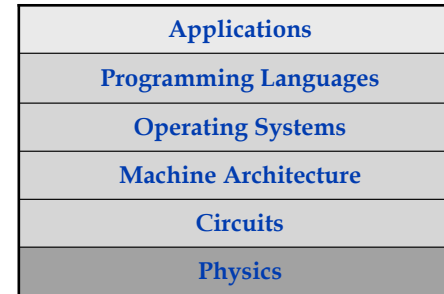


CPS 001

1.17

Layers of abstraction

The User:



The Result:



CPS 001

1.18

Survey the field

- Artificial intelligence
- Graphics/Multimedia
- Parallel Computation
- Programming Languages
- Systems
- Scientific Computing
- Theory
- User Interfaces

CPS 001

1.19

Themes and Concepts of CS

- Theory
 - > properties of algorithms, how fast, how much memory
 - > average case, worst case: sorting cards, words, exams
 - > *provable* properties, in a mathematical sense
- Language
 - > programming languages: C++, Java, C, Perl, Fortran, Lisp, Scheme, Visual BASIC, ML, ...
 - > Assembly language, machine language,
 - > Natural language such as English
- Architecture
 - > Main memory, cache memory, disk, USB, SCSI, ...
 - > pipeline, multi-processor

CPS 001

1.20

Complexity: What's hard, what's easy?

- What is a prime number?
 - > 2, 3, 5, 7, 11, 13, ...
 - > Largest prime?
- 48112959837082048697
- 671998030559713968361666935769
- How do we determine if these numbers are prime?
 - > Test 3, 5, 7, ...
 - > If we can test one million numbers a second, how long to check a 100 digit #?
- Why do we care?
- 671998030559713968361666935767 is not prime, I can prove it but I can't give you the factors.
- Finding factors is "hard", determining primality is "easy"
 - > What does this mean?
 - > Why do we care?
- Encryption depends on this relationship, without encryption and secure web transactions where would we be?

CPS 001

1.21

Questions you will be able to answer

- Vendor tries to sell you a system that will check all of your systems and procedures to see if they are correct.
 - > A good deal?
- Programmer tells you that to optimize the routing of your sales personnel is beyond the power of today's computers.
 - > Do you believe her?
- Computer consultant demonstrates complicated management system with test data including a handful of employees.
 - > Is the performance with this small set of data a good indicator of how the system will perform with all of your company data entered?

CPS 001

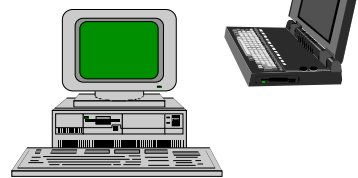
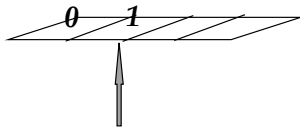
1.22

What is a computer?

- Turing machine: invented by Alan Turing in 1936 as a theoretical model

Mainframe, PC, laptop, supercomputer

infinite tape, moving tape-reader



A computer is a computer, is a computer, Church-Turing Thesis, all have same "power"

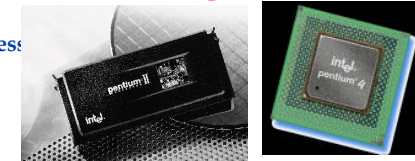
CPS 001

1.23

Chips, Central Processing Unit (CPU)

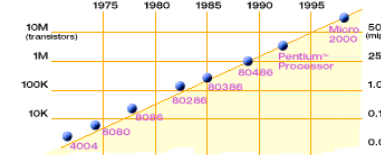
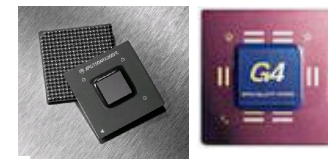
- CPU chips/Microproces

- > Pentium (top)
- > G3/4 (bottom)
- > Sound, video, ...



- Moore's Law

- > chip "size" (# transistors) doubles every 12--18 months (formulated in 1965)
- > 2,300 transistors Intel 4004, 42 million Pentium 4



CPS 001

1.24