CompSci 6
Program Design and Methodology I

Robert Duvall and Dietolf Ramm
http://www.cs.duke.edu/courses/cps006/spring04
http://www.cs.duke.edu/~rcd
http://www.cs.duke.edu/~dr
Computer Science and Programming

- **Computer Science is more than programming**
  - The discipline is called *informatics* in many countries
  - Elements of both science and engineering
    - Scientists build to learn, engineers learn to build
    - Fred Brooks
  - 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**
What is Computer Science?

What is it that distinguishes it from the separate subjects with which it is related? What is the linking thread which gathers these disparate branches into a single discipline? My answer to these questions is simple --- it is the art of programming a computer. It is the art of designing efficient and elegant methods of getting a computer to solve problems, theoretical or practical, small or large, simple or complex.

(Tony) Hoare

C.A.R.
Computer Science

- **Artificial Intelligence** thinking machines
- **Scientific Computing** weather, cars, heart, modelling
- **Theoretical CS** analyze algorithms, models
- **Computational Geometry** theory of animation, 3-D models
- **Architecture** hardware-software interface
- **Software Engineering** engineering, science
- **Operating Systems** the soul of the machine
- **Graphics** from Windows to Hollywood
- **Many other subdisciplines**
Stories

- Who is Shawn Fanning and what did he do (19 years old)?
- Who is Marc Andreessen and what did he do (21 years old)?
- Who is Claude Shannon and what did he do (21 years old)?
- Who is Linus Torvalds and what did he do (21 years old)?
- Who is Dmitry Sklyarov and what did he do (26 years old)?
- Who is Tim Berners-Lee and what did he do (35 years old)?
Algorithms as Cornerstone of CS

- Step-by-step process that solves a problem
  - more precise than a recipe
  - eventually stops with an answer
  - general process rather than specific to a computer or to a programming language

- Searching: for phone number of G. Samsa, whose number is 929-9338, or for the person whose number is 489-6569
  - Are these searches different?

- If the phone book has 8 million numbers in it (why are there only 7.9 million phone numbers with area code 212?)
  - How many queries to find phone number of G. Samsa?
  - How many queries to find person with number 929-9338
  - What about IP addresses?
Search, Efficiency, Complexity

- Think of a number between 1 and 1,000
  - respond high, low, correct, how many guesses needed?

- Look up a word in a dictionary
  - Finding the page/word, how many words do you look at?

- Looking up a phone number in the Manhattan, NY directory
  - How many names are examined?

- How many times can 1,024 be cut in half?
  - $2^{10} = 1,024$, $2^{20} = 1,048,576$
Sorting Experiment: why do we sort?

- Groups of four people are given a bag containing strips of paper
  - on each piece of paper is an 8-15 letter English word
  - create a sorted list of all the words in the bag
  - there are 100 words in a bag

- What issues arise in developing an algorithm for this sort?
  -
  -

- Can you write a description of an algorithm for others to follow?
  - Do you need a 1-800 support line for your algorithm?
  - Are you confident your algorithm works?
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, ...
  - Assembly language, machine language,
  - Natural language such as English

● **Architecture**
  - Main memory, cache memory, disk, USB, SCSI, ...
  - pipeline, multi-processor
Theory, Language, Architecture

• We can prove that in the worst case quicksort is bad
  ➢ doesn’t matter what machine it’s executed on
  ➢ doesn’t matter what language it’s coded in
  ➢ unlikely in practice, but worst case always possible

• Solutions? Develop an algorithm as fast as quicksort in the average case, but has good worst case performance
  ➢ quicksort invented in 1960
  ➢ introsort (for introspective sort) invented in 1996

• Sometimes live with worst case being bad
  ➢ bad for sorting isn’t bad for other algorithms, needs to be quantified using notation studied as part of the theory of algorithms
What is an integer?

- In mathematics we can define integers easily, infinite set of numbers and operations on the numbers (e.g., +, -, *, /) \{…-3, -2, -1, 0, 1, 2, 3, …\}

- In programming, finite memory of computer imposes a limit on the magnitude of integers.
  - Possible to program with effectively infinite integers (as large as computation and memory permit) at the expense of efficiency
  - At some point addition is implemented with hardware, but that’s not a concern to those writing software (or is it?)
  - C++ doesn’t require specific size for integers, Java does

Floating-point numbers have an IEEE standard, it’s more expensive to do arithmetic with 3.14159 than with 2
Alan Turing (1912--1954)

- Instrumental in breaking codes during WW II
- Developed mathematical model of a computer called a Turing Machine (before computers)
  - solves same problems as a Pentium III (more slowly)
- Church-Turing thesis
  - All “computers” can solve the same problems
- Showed there are problems that cannot be solved by a computer
- Both a hero and a scientist/mathematician, but lived in an era hard for gay people
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?

- **Why do we care?**

A Computer Science Tapestry
C.A.R. (Tony) Hoare (b. 1934)

- Won Turing award in 1980
- Invented quicksort, but didn’t see how simple it was to program recursively
- Developed mechanism and theory for concurrent processing
- In Turing Award speech used “Emporer’s New Clothes” as metaphor for current fads in programming

“Beginning students don’t know how to do top-down design because they don’t know which end is up”
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
- From ideas to electrons
From High- to Low-level languages

- C++ is a multi-purpose language, we’ll use it largely as an object-oriented language, but not exclusively
  - Contrast, for example, with Java in which everything is a class
  - Contrast with Fortran in which nothing is a class
- Compilers translate C++ to a machine-specific executable program
  - The compiler is a program, input is C++, output is an executable
  - What language is the compiler written in?
  - In theory C++ source code works on any machine given a compiler for the machine
- C++ and other programming language are more syntactically rigid than English and other natural languages
Ripped from (real) Headlines

- Police Squad Helps Dog Bite Victim
- Milk Drinkers Turn to Powder
- Kids Make Nutritious Snacks
- Red Tape Holds Up New Bridge
- Teacher Strikes Idle Kids
- Tuna Biting Off Coast of Washington
- Local High School Dropouts Cut in Half
- Old School Pillars are Replaced by Alumni

(borrowed from Rich Pattis, pattis@cs.cmu.edu)
Levels of Programming Language

- Machine specific assembly language, Sparc on left, Pentium on right, both generated from the same C++

```c
main:
    save %sp,-128,%sp
    mov 7,%o0
    st %o0,[%fp-20]
    mov 12,%o0
    st %o0,[%fp-24]
    ld [%fp-20],%o0
    ld [%fp-24],%o1
    call .umul,0
    nop
    st %o0,[%fp-28]
    mov 0,%i0
    b .LL1
    nop

main:
    pushl %ebp
    movl %esp,%ebp
    subl $12,%esp
    movl $7,-4(%ebp)
    movl $7,-4(%ebp)
    imull -8(%ebp),%eax
    movl %eax,-12(%ebp)
    xorl %eax,%eax
    jmp .L1

.LL1:
    xorl %eax,%eax
    jmp .L1
```

A Computer Science Tapestry
1.18
Alternatives to compilation

- Some languages are *interpreted*, Scheme and Java are examples
  - like simultaneous translation instead of translation of written document. The same word may be translated many times
  - The interpreter is a program that translates one part of a source code at a time
    - The interpreter is machine specific, written in some programming language
- JVM, the Java Virtual Machine
  - Like a PC or Mac but machine is virtual, written in software
  - Executes Java byte codes which are created from Java source
    - Like assembly language: between source code and executable
      - JVM must be written for each architecture, e.g., Linux, Windows, Mac, BeOS, ...
What is digital?

- What’s the difference between
  - Vinyl LP and CD/DVD?
  - Rolex and Timex?

- Sampling analog music for CD’s
  - 44,100 samples/channel/second * 2 channels * 2 bytes/sample * 74 minutes * 60 seconds/minute = 783 million bytes

- How does MP3 help?
What is a computer?

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

A computer is a computer, is a computer, Church-Turing Thesis, all have same "power"
Chips, Central Processing Unit (CPU)

- CPU chips
  - 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.
Why is programming fun?

What delights may its practitioner expect as a reward?

First is the sheer joy of making things

Second is the pleasure of making things that are useful

Third is the fascination of fashioning complex puzzle-like objects of interlocking moving parts

Fourth is the joy of always learning

Finally, there is the delight of working in such a tractable medium. The programmer, like the poet, works only slightly removed from pure thought-stuff.