CS100: Introduction to Computer Science

Lecture 1: Introduction
(Survey, Pictures)

Course Information

- Instructor: Xiaoyan Li
- Lecture: Mon. & Wed. 11:00am – 12:15pm
- Room: Kendade Hall 305
- Labs: Wed or Thu 1:00pm – 2:50pm
- Room: Kendade Hall G06 (Visilab)
- Office hour: Tu/Th 10:00am – 11:00am (or by appointment)
- Office: Clapp 227
- Email: xli@mtholyoke.edu

Lab Instructor & Teaching Assistant

- Lab instructor: Jasper Lin
  - Office: Clapp 201
  - Email: jlin@mtholyoke.edu
- Teaching assistant: Nina Yi
  - Office: Visilab
  - Email: yi20n@mtholyoke.edu
  - Office hours: Tue./Wed. 7:00-9:00pm

Course Objectives

- Fundamental understanding of the field,
- Experience with programming, and
- Research topics and applications
- This course fits for computer-science-major students as well as for non-major students

Course Information

- Textbook: Computer Science : an Overview
  by J. Glenn Brookshear, 9th Edition
- Topics:
  - Data encoding & storage
  - Machine architecture
  - Operating system
  - Networking & the Internet
  - Algorithm
  - Programming languages
  - Software engineering
  - Database systems
  - Theory of computation
  - Artificial intelligence ...

Structure of the Course

- The course is divided into three parts.
  1. Covers basic concepts in computer science.
  2. Discusses various programming languages.
     - Takes weekly labs to learn how to create a web page using HTML and how to program using PERL.
  3. Introduces some important applications and research topics in computer science
     - such as databases, web search, artificial intelligence and data mining.
Tentative schedule:

- **CS-100 Introduction to Computer Science**

Grading:

- Class participation: 5%
- Five homework assignments: 20%
- Six labs: 30%
- Two midterms: 30%
- One final exam: 15%

Advice:

- Try attend every class, learn actively
- Read textbook either before or after a lecture
- Start homework sooner, no late homework accepted
- Ask questions (in class, office hours, email)
- Discuss ideas with your classmates but not homework solutions
- **Give me prompt feedback!**

About the Computer Science:

- Computer science is a fast-growing field
  - Computing power (BOLDa mini pc, booksize, 1.9lbs)
  - Programming languages
  - Applications & on-going research
  - Impacts of computer science on society and our daily life
    - Communication: email, instant messenger, blogs, teleconferencing…
    - Banking, shopping, learning & teaching …
  - Career opportunities

Chapter 0:

- **Introduction**

Chapter 0:

- The Origins of Computing Machines
- The Role of Algorithms
- Relationship with Other Subjects
Figure 0.3 An Abacus

Origins of Computing Machines
- Early computing devices
  - Abacus: positions of beads represent numbers
  - Gear-based machines (1600s-1800s)
    - Positions of gears represent numbers
  - Blaise Pascal, Wilhelm Leibniz, Charles Babbage

Figure 0.4 The Mark I computer

Early Computers
- Based on mechanical relays
  - 1940: Stibitz at Bell Laboratories
  - 1944: Mark I: Howard Aiken and IBM at Harvard
- Based on vacuum tubes
  - 1937-1941: Atanasoff-Berry at Iowa State
  - 1940s: Colossus: secret German code-breaker
  - 1940s: ENIAC: Mauchly & Eckert at U. of Penn.

Personal Computers vs. Mainframes and Minicomputers
- Mainframes (1960s, room size)
  - Multi-users share the computers
    - Offline preparation of tasks (punched cards), no direct interaction
    - Time-shared terminal computers.
- Minicomputers (1970s, refrigerator)
  - Graphical user interface, high resolution screen, large memory storage, mouse, special software
- Personal computers (desktop, laptop)
  - IBM introduced the PC in 1981
  - Standard hardware design for most desktop computers
  - Most PnP use software from Microsoft

Personal Computers vs. Workstations
- Mainframes (1960s, room size)
- Minicomputers (1970s, refrigerator)
- Personal computers (1980s, desktop, laptop)
- Workstations (1980s, high-end desktop)
  - High performance CPU
  - Large memory
  - High speed networking
  - Extremely reliable components
  - Large displays
  - High 3D graphics hardware ...
What are the first few terms in your mind when you think of computer science?

Algorithm: A set of steps that defines how a task is performed
Program: A representation of an algorithm
Programming: The process of developing a program
Software: Programs and algorithms.
Hardware: Equipment

The study of algorithms was originally a subject in mathematics.
Early examples of algorithms
Euclidean Algorithm to find a greatest common divisor
Gödel’s Incompleteness Theorem: Some problems cannot be solved by algorithms.

The central role of algorithms in computer science

Which problems can be solved by algorithmic processes?
How can algorithm discovery be made easier?
How can techniques of representing and communicating algorithms be improved?
How can our knowledge of algorithms and technology be applied to provide better machines?
How can characteristics of different algorithms be analyzed and compared?

Description: This algorithm assumes that its input consists of two positive integers and proceeds to compute the greatest common divisor of these two values.

Procedure:
Step 1. Assign M and N the value of the larger and smaller of the two input values, respectively.
Step 2. Divide M by N, and call the remainder R.
Step 3. If R is not 0, then assign M the value of N, assign N the value of R, and return to step 2; otherwise, the greatest common divisor is the value currently assigned to N.
Computer science is the science of algorithms.

Relationship and Other Subjects

- Draws from other subjects, including
  - Mathematics, Engineering
  - Psychology, Biology
  - Business Administration, etc
- Brings new fields and issues, including
  - Management information systems, e-commerce
  - Digital library,
  - Bioinformatics, etc.

Next Lecture:

- Bits, storage and main memory
- Reading assignments: Chapter 1.1, 1.2