COS 116: The Computational Universe

- Instructor: Adam Finkelstein
- TAs: Sema Berkiten & Sourindra Chaudhuri

- Labs
  - Mon, Wed 7:30-10:20pm, Friend 009
  - This week only: take-home lab
Course Summary

Computers have brought the world to our fingertips. We will try to understand at a basic level the science -- old and new -- underlying this new Computational Universe. Our quest takes us on a broad sweep of scientific knowledge and related technologies: propositional logic of the ancient Greeks (microprocessors); quantum mechanics (silicon chips); network and system phenomena (internet and search engines); computational intractability (secure encryption); and efficient algorithms (genomic sequencing). Ultimately, this study makes us look anew at ourselves -- our genome; language; music; "knowledge"; and, above all, the mystery of our intelligence. This course satisfies Princeton's Science and Technology (with Lab) distribution requirement.

Administrative Information
Ancient dream: “Breathe life into matter”

Golem (Jewish mythology)  Automaton (Europe)

Frankenstein (Shelley 1818)  Robot (Capek 1920)
“Breathe life into matter” – Another perspective
“Breathe life into matter” – A 20th century perspective

- “Matter”: Atoms, molecules, quantum mechanics, relativity …

- “Life”: Cells, nucleus, DNA, RNA, …

- “Breathe life into matter”: Computation
  One interpretation: Make matter do useful, interesting things on its own
Breathing life into matter…

Military was a major sponsor of computational research in 20th century
Electric Sheep

electricsheep.org
hifidreams.com
sample july 2006

Distributed under the Creative Commons Attribution 2.5 License

Scott Draves
Computational Universe
Some important distinctions

Computer Science vs. Computer Programming
(Java, C++, etc.)

Notion of computation vs. Specific implementation
(Silicon, robots, Xbox, etc.)
Course **not** about programming!

- Not necessary for understanding

- More time for to cover computer science (broader than COS126!)

- Little advantage to those who have prior programming experience
Brief history of computation

- Technological:
  - Clocks
  - Clockwork “Automata”
  - Mechanized looms, steam engines
  - Vacuum tubes, electronic calculators (1910-1930’s)
  - ENIAC (1945)
  - von Neumann Computer (1949, Princeton)
Brief history of computation

- **Intellectual**
  - Ancient Greeks, philosophers
    - (How to “formalize thought”)
  - Boolean logic (G. Boole, 1815-1864)
  - Crisis in math
    - Hilbert: Call to systematize math
    - Gödel: Incompleteness theorem
  - Lambda calculus (A. Church, 1936)
  - Turing machines (A. Turing, 1937)

Both at Princeton;
First clear notion of “What is computation?”

Wang tiles 1961
Computer Science:
A new way of looking at the world
Example 1:
Example 2: Public closed-ballot elections

- Hold an election in this room
  - Everyone speaks publicly (no computers, email, etc.)
  - End: everyone agrees on who won and margin
  - No one knows how anyone else voted

- Is this possible?
  - Yes! (A. Yao, Princeton)
Example 3: Computational Biology

Old Biology

New Biology

Microarrays

Pathways
COS 116

- First 10 lectures:
  - Cool things computers do and how

- Next 8 lectures:
  - What’s inside, internet, silicon chips

- Last 6 lectures:
  - Complexity, cryptography, viruses, search engines, artificial intelligence
This week’s lab: Web 2.0

Take-home lab – see course web page.

This week’s reading:

Brooks
pp 12-21, pp 32-51
See course web page.
Grading

- Midterm: 15%
- Final: 35%
- Lab reports: 35%
- Participation (class, blog): 15%

- Attendance expected at lectures and labs
Next couple labs: Scribbler. What determines its behavior?