Undergraduate Curriculum Revision

Mehran Sahami
Faculty Lunch – March 11, 2008

Outline
• Brief review of current curriculum
• Structure of new curriculum
• CS Core courses
• Initial set of tracks
• Vote

Current Curriculum: Lower Division

AP credit or strong prior programming experience

Programming
- Programming methodology and abstraction
- Object-oriented systems design

Theory
- Discrete mathematics
- Discrete structures

CS106A → CS106B/X → CS107 → CS108

CS103A → CS103B → CS103X → CS103A

Current Curriculum: Upper Division

• Theory Depth
  – CS154: Automata and Complexity Theory
  – CS161: Design and Analysis of Algorithms
• Systems Depth
  – EE108B: Digital Systems II
  – Two Systems Electives (OS, Compilers, Networking, etc.)
• Applications
  – CS121/221: Artificial Intelligence
  – One Applications Elective (Databases, HCI, Graphics, etc.)
• 2-3 Restricted CS Electives
• Senior Project capstone course

Outline
• Brief review of current curriculum
• Structure of new curriculum
• CS Core courses
• Initial set of tracks
• Vote
Revised Curricular Structure: Core

Theory Core: 3 Courses
- Incorporates portions of current theory sequence
- Eliminates redundancies in cs103 and cs161
- CS-owned probability course with AI applications

Systems Core: 3 Courses
- Incorporates portions of current intro programming sequence
- Incorporates systems concepts in later programming projects
- CS106A considered "funnel" into core (not part of core)

Revised Curricular Structure: Tracks

-~4 Courses
- Students must complete requirements for any one track
- Developing depth in a specialization
- Provide course/theme options within each track
- Provide multi-disciplinary options
- Modularize curriculum

Why Tracks?

- Explicitly shows available options
  - Allows students to focus on areas in which they have the greatest interest, thus increasing appeal of program
- Helps eliminate image of CS as "just programming"
  - Shows diversity of themes in computer science
  - Provides more context for what is possible with CS degree
- Provides organizational infrastructure
  - Easier to evolve major as the field evolves
  - E.g., add/drop/modify tracks (or programs in them)

Some More Food For Thought

- Tracks already exist:
  - Applied Logic
  - Artificial Intelligence
  - Decision Making and Rationality
  - Human-Computer Interaction
  - Learning
  - Natural Language
  - Networking
  - Robotics and Mechatronics
  - Digital Systems
  - Biology
  - Computational Engineering
  - Computer Hardware
  - Computer Software
  - Communications and Signal Processing
  - ...in Symbolic Systems
  - ...and CSE
  - ...and MCS
  - ...and EE

Revised Curricular Structure: Electives

~2-4 Courses
- Restricted electives
- Allow pursuing breadth and/or additional depth
- Track-specific elective options allow for interdisciplinary work

Revised Curricular Structure: Capstone

1 Course
- "Senior project" capstone course
- Developing capstone courses to parallel tracks
- Both application development and research options
Structure Aligns With Broader Context

  - Supports tracks model
  - Revision committee adopted modular structure to support adaptability
- ICER: Integrative Computing Education & Research
  - Change the popular image of computing
  - Encourage curricular experimentation and innovation
  - Make sure introductory students recognize that the field offers many opportunities
  - Strengthen interdisciplinary connections
- Peer institutions
  - Many peer institutions moving in same direction for similar reasons

Increasing the “Footprint” of CS

- Core material everyone sees is streamlined to accommodate

“Footprint” of CS Students See Today

Tracks Allow More Depth...

...in a More Diverse Set of Areas

Total Potential “Footprint” is Larger
Curriculum Revision Process

- Curriculum committee met regularly throughout the Fall to discuss and refine core courses
  - Respective subcommittees formed to define Systems and Theory core
  - Proposals for both Systems and Theory core each revised at least 4 times
- Throughout the Winter, various faculty subcommittees met to define Tracks
  - Proposals for eight initial tracks produced (most with multiple revisions)
- Curriculum committee reviewed and approved proposals
  - Still need full faculty approval
  - That’s one of the reasons we’re all here...

Please Acknowledge Everyone Involved

Curriculum Committee
- Jerry Cain
- Bill Dally
- Vladlen Koltun
- Phil Levis
- John Mitchell
- Andrew Ng
- Nick Parfait
- Eric Roberts
- Mendel Rosenblum
- Mehran Sahami
- Julie Zelenski
- Alex Aiken
- Serafim Batzoglou
- Gill Bejerano
- Ron Fedkiw
- Hector Garcia-Molina
- Leo Guibas
- Pat Hanrahan
- Scott Klemmer
- Daphne Koller
- David Koslow
- Jean-Claude Latombe
- Marc Levoy
- Chris Manning
- Mendel Rosenblum
- Mehran Sahami
- Julie Zelenski
- Nick Parfait
- Eric Roberts
- Alex Aiken
- Serafim Batzoglou
- Gill Bejerano
- Ron Fedkiw
- Hector Garcia-Molina
- Leo Guibas
- Pat Hanrahan
- Scott Klemmer
- Daphne Koller
- David Koslow
- Jean-Claude Latombe
- Marc Levoy
- Chris Manning
- Russ Altman
- Many additional faculty (email/informal meetings)

Shows real dedication to undergraduate education

Outline

- Brief review of current curriculum
- Structure of new curriculum
- CS Core courses
- Initial set of tracks
- Vote

Systems I: Programming Abstractions

- Philosophy: Classic CS2 course: problem solving, basic abstract data types, and recursion
- Basically, our current CS106X/B course
- General Topics
  - Programming methodology (engineering, modularity, documentation)
  - Algorithmic thinking and problem solving
  - Data abstractions
    - Stacks
    - Queues
    - Linked lists
    - Hash tables
    - Binary trees
    - Generics/templates
  - Recursion
    - Procedural recursion
    - Recursive backtracking
  - Searching and sorting
    - Basic algorithm analysis (big O) and comparison

Systems II: Computer Organization and Systems

- Philosophy: From hardware up to the source code
- Heavily modified CS107, with material from CS143 & EE108B
- General Topics
  - Machine architecture
    - Registers, ALU, CPU, RAM, I/O, basic assembly language
  - Caching, pipelining
  - Memory model
    - Pointers, Heap management, garbage collection
    - Low-level polymorphism and runtime type identification
  - Data representation
  - Facility with C programming as part of topical coverage
  - Compilation
    - Function call mechanics and stack frames
  - Semantic analysis
    - Simple (intermediate) code generation
  - Basic concurrency usage
    - Threading
    - Synchronization, locks and semaphores

Systems III: Principles of Computer Systems

- Philosophy: Building larger scale systems using OS and networking abstractions
- Entirely new course (not replacement for CS140 or CS244A)
- General Topics
  - Processes
    - Concurrency mechanics on a single processor
    - Context switching, interrupts and exceptions
    - Forking processes, process mechanics and management
    - Interprocess communication
    - Threading
  - Storage and file management
    - File systems
    - Virtual memory and paging
  - Networking
    - Sockets
    - Blocking vs. non-blocking strategies
    - Transport layer: TCP/IP
    - Network layer: names, routing
  - Understanding of distributed systems
Theory I: Mathematical Foundations of Computing

- Philosophy: Mathematical essentials for CS, with proofs
- New course leveraging CS103/154 (doesn’t replace CS154)
- General Topics
  - Logic and proof techniques (9 lectures)
    - Prop. and predicate logic (with quantification), formal proof methods
    - Applications: Satisfiability, SAT solving (Putnam-Davis)
  - Induction (4 lectures)
    - Formal proofs and applications: program proofs, structural induction
  - Sets, functions, and relations (4 lectures)
  - Theory and applications (error-correcting codes, social networks)
  - Intro to formal languages (1 lecture)
  - DFAs, NFAs, and Regular Expressions (4 lectures)
  - Context-free Grammars (2 lectures)
  - Turing machines (3 lectures)
  - TM, TM program, Undecidability and the Halting problem
  - NP-completeness (3 lectures)
    - P and NP, examples of NP-complete problems and reductions
    - SAT revisited and Cook’s theorem

Theory II: Intro. to Probability for Computer Scientists

- Philosophy: Probability relevant to CS, with applications
- Entirely new course
  - Replaces Stat116, adds CS applications and machine learning
- General Topics
  - Counting and Combinatorics
    - Combinations, Permutations, Pigeonhole principle
  - Relations, partial orders (concepts, definitions, and proofs)
  - Probability theory
    - Random variables and event spaces
    - Conditional probability, independence, conditional independence
    - Distributions: Uniform, Binomial, Multinomial, Normal, Poisson
    - Point estimation, expectation, variance
  - Bayes’ Theorem, Law of large numbers, Central Limit Theorem
  - Hypothesis testing
    - Applications: hashing, PageRank, data analysis, inference
  - Intro. to Machine Learning
    - Hypothesis spaces, learning as search
    - Data fitting, Naive Bayes, Logistic Regression
    - Applications: Email spam filtering, Recommender systems

Theory III: Data Structures and Algorithms

- Philosophy: Analysis of data structures and algorithms
- Streamlined version of CS161 with topics from CS103 added
- General Topics
  - Algorithmic complexity and analysis (4 lectures)
    - Asymptotics: Big Oh, Omega, and Theta notation
    - Recurrence relations
    - Master theorem
  - Randomization, divide and conquer (2 lectures)
    - Introduction to randomized algorithms
    - Quicksort, divide and conquer
    - Heaps and counting sort (1 lecture)
    - Hashing (2 lectures)
    - Tree and graph definitions and properties, BSTs (2 lectures)
  - Greedy Algorithms (including min-cost spanning trees) (2 lectures)
  - Dynamic programming (3 lectures)
  - Graph algorithms, shortest paths, and applications (4 lectures)
  - Blind and heuristic search (A*) in graphs (1 lecture)

Prerequisite Structure

Courses at appropriate level of difficulty to mature students from introductory to depth material in a reasonable manner

Summary of Changes

- CS103 sequence and STAT116 are replaced as Math requirements with Theory I and II
- CS161 incorporated into Theory III (remains CS depth course)
- CS154, CS121/221 (and other application course) not in core
- No change in Math units, 9 net units opened in CS depth
- Systems I considered an Engineering Fundamental
- Systems II and III are CS depth courses
- CS108, EE108B and Systems electives not in core
- 10 net units opened in CS depth
- Existing electives provide 6 additional units of CS depth
- Net result: 25 units opened in CS depth

Outline

- Brief review of current curriculum
- Structure of new curriculum
- CS Core courses
  - Initial set of tracks
  - Vote
**Track Structure**

- Combination of track requirements and electives satisfies:
  - minimum of 7 courses, and
  - minimum of 25 units
- All tracks have at least 4 (possibly more) required courses
  - Students will generally have room for 2 to 4 elective courses
  - Required senior project is not considered part of track
- Elective courses
  - Set of general CS electives that all students may choose from
  - Additionally, each track specifies track-specific electives that may count as elective courses only by students in that track
  - Track-specific electives allow for additional depth or related interdisciplinary course options

**General CS Electives**


  *Students may not count both CS121 and CS221, or both CS148 and CS248 toward their major requirements.

  - Very similar to our current set of electives
    - Added courses that are no longer required in major (e.g., CS108, CS121/221)
    - Added some newly proposed undergraduate courses (e.g., CS124, CS142, CS164)

**Initial Set of Tracks Areas**

- Artificial Intelligence
- Theory
- Systems
- Human-Computer Interaction
- Graphics
- Information
- Bio computation
- Unspecialized
- Individually Designed

**AI Track**

- **Requirements**
  a) CS221
  b) Any two of: CS223A, CS223B, CS224N, CS226, CS227, CS228, CS229
  c) One additional class from category (b) or from the following:

- **Track electives**

- Courses in categories (b) and (c) above, as well as:


  *Students may not count both Phil51 and CS157 toward major requirements.

**Theory Track**

- **Requirements**
  a) CS154
  b) Any one of: CS164, CS225, CS229, CS261, CS268, CS361A, CS361B, CS365

- **Track electives**

- Courses in categories (b) and (c) above, as well as:

- CME302, CME305, Phil51*, Phil152

  *Students may not count both Phil51 and CS157 toward major requirements.

**Systems Track**

- **Requirements**
  a) CS140
  b) One of: CS143 or EE108B
  c) Two additional courses from category (b) or from the following:

- **Track electives**

- Courses in category (c) above, as well as:

- CS240E, CS240K, CS240N, CS240S, CS240U, CS242, CS243, CS244, CS245, EE271, EE282

  *Requires approval of undergraduate advisor.

**General CS Electives**


  *Students may not count both CS121 and CS221, or both CS148 and CS248 toward their major requirements.

  - Very similar to our current set of electives
    - Added courses that are no longer required in major (e.g., CS108, CS121/221)
    - Added some newly proposed undergraduate courses (e.g., CS124, CS142, CS164)

**Initial Set of Tracks Areas**

- Artificial Intelligence
- Theory
- Systems
- Human-Computer Interaction
- Graphics
- Information
- Bio computation
- Unspecialized
- Individually Designed

**AI Track**

- **Requirements**
  a) CS221
  b) Any two of: CS223A, CS223B, CS224N, CS226, CS227, CS228, CS229
  c) One additional class from category (b) or from the following:

- **Track electives**

- Courses in categories (b) and (c) above, as well as:


  *Students may not count both Phil51 and CS157 toward major requirements.

**Theory Track**

- **Requirements**
  a) CS154
  b) Any one of: CS164, CS225, CS229, CS261, CS268, CS361A, CS361B, CS365

- **Track electives**

- Courses in categories (b) and (c) above, as well as:

- CME302, CME305, Phil51*, Phil152

  *Students may not count both Phil51 and CS157 toward major requirements.

**Systems Track**

- **Requirements**
  a) CS140
  b) One of: CS143 or EE108B
  c) Two additional courses from category (b) or from the following:

- **Track electives**

- Courses in category (c) above, as well as:

- CS240E, CS240K, CS240N, CS240S, CS240U, CS242, CS243, CS244, CS245, EE271, EE282

  *Requires approval of undergraduate advisor.
**HCI Track**

- **Requirements**
  a) CS147, CS247 (HCI Foundations)
  b) Any one of: CS148, CS248, CS376, CS377, CS378 (Advanced HCI)
  c) Any one of: CS108, CS140, CS221, CS223B, CS229, CS249A (Buttressing CS)
  d) Any one of: Psych55, Psych252, MS&E184, ME101, ME115 (Designing for People)

- **Track electives**
  - Courses in categories (b), (c), and (d) above, as well as:
    - ArtStudi60, Comm269, CME340, CS447*, CS448B*, Ling180, ME118, MS&E216A, Psych265, Psych211
  *requires approval of undergraduate advisor.

**Graphics Track**

- **Requirements**
  a) CS248
    (Starting in AY ’09-10, CS148 and CS248 will both be required as a two course sequence)
  b) Any one of: CS205A, CME104, CME108, Math62, Math113
    (Of the choices above, CS205A is strongly recommended as a preferred choice)
  c) Any two of: CS164, CS179, CS205B, CS268, CS348A, CS348B, CS448
    Note: CS164: Computing with Physical Objects (new course by Leo Guibas)
    CS179: Digital Photography (new course by Marc Levoy)

- **Track electives**
  - Courses in category (c) above, as well as:
    - ArtStudi 60, ArtStudi 70, ArtStudi 179, CS48N, CS277, CS326A, CME302, CME306, CME324, EE262, EE264, EE278, EE368, ME101, Psych30, Psych211, STS144

**Information Track**

- **Requirements**
  a) CS124, CS145
    Note: CS124: From Languages to Information
    (new course by Dan Jurafsky and Chris Manning)
  b) Two courses, which must be from different areas below:
    - Information-based AI Applications
      CS224N, CS224B, CS229
    - Database and Information Systems
      CS140, CS240D, CS245, CS345A, CS345C, CS346, CS347
    - Information Systems in Biology
      CS262, CS270, CS274
    - Information Systems on the Web
      CS276, CS348B, <<Future course on Internet Algorithmics by Rajeev Motwani>>

- **Track electives**
  - Courses in category (b) may also be counted as electives

**Biocomputation Track**

- **Requirements** (based on BMC Informatics Track)
  - Mathematics: (1 course less than CS)
    - Math 41, Math 42, Theory I, Theory II
  - Science: (1 course less than CS)
    - Phys 41, Chem 31A/B or 31X, Chem 33
    - Bio41, 42, 43 or HumBio2A, 3A, 4A or HumBio Core I, II, III
  - Engineering Fundamentals: (1 course less than CS)
    - CS108B/C, Systems I
  - Elective
  - Additional CS Core: (same as CS)
    - Theory III, Systems II, Systems IV
  - Bioinformatics Track: (6-7 units)
    - Any one of: CS121, CS221, CS228, CS229, CS232B
    - Any one of: CS276, CS277A, CS274, CS275, CS276, CS278, CS279, CS326
  - Biocomputation Electives: (12-13 units; different than general CS electives)
    - Two additional BMC Informatics electives
    - One BMC Cell/Mol elective
    - One BMC Organ elective
  - Total: 94-99 units versus 93 units for Standard CS Track

**Unspecialized Track**

- **Requirements**
  a) CS154
  b) Any one of: CS140, CS143
  c) One additional class from category (b) or from the following: EE108B, CS144, CS155, CS240D, CS242, CS244
  d) Any one of: CS121 or CS221, CS223A, CS228, CS228, CS229
  e) Any one of: CS145, CS147, CS148 or CS248, CS262

  - This is basically our current curriculum
    - Adapted to fit into new track structure
    - Some additional options for all courses

**Individually Designed Track**

- **Students may propose Individually Designed Track**
- Must be an intellectually coherent program of study
- Proposal should justify program and why it cannot be satisfied via an existing track

- **Must specify equivalent of track and electives**
- Minimum of 7 courses: at least 4 must be CS courses numbered 100 or above
  - Each course must be taken for a minimum of 3 units
  - Minimum of 25 total units for track + electives

- **Proposal must be approved by undergraduate advisor and Associate Chair**
- Approvals must be obtained at least 2 quarters prior to completion of program

- **Proposal cannot modify any non-track/elective requirements**
  - E.g., SoE requirements (Math, Science, Eng Fundamentals) cannot be modified
  - Must take all CS Core courses
Preliminary Unit Calculations

- Core (30 units)
  - Theory: 3 courses @ 5 units = 15 units
  - Systems: 3 courses @ 5 units = 15 units
- Upper division (28 units)
  - Track: 4-5 courses
  - Electives: 2-4 courses
  - Capstone: 3 units
- Total related units = 58 units (same as now)
  - 10 units are classified as Math
  - 5 units are classified as Engineering Fundamentals
- Total CS depth units = 43 units (same as now)

Curriculum Comparison

<table>
<thead>
<tr>
<th>Current</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming: 3 courses</td>
<td>Systems core: 3 courses</td>
</tr>
<tr>
<td>Theory: (2 + 2 depth): 4 courses</td>
<td>Theory core: 3 courses</td>
</tr>
<tr>
<td>Systems depth: 3 courses</td>
<td>Track (depth): 4-5 courses</td>
</tr>
<tr>
<td>Applications: 2 courses</td>
<td>Electives: 2-4 courses</td>
</tr>
<tr>
<td>Electives: 2 courses</td>
<td>Capstone: 1 course</td>
</tr>
<tr>
<td>Capstone: 1 course</td>
<td>TOTAL: 14-15 courses</td>
</tr>
<tr>
<td>TOTAL: 15 courses</td>
<td>Same number of units in both cases</td>
</tr>
</tbody>
</table>

Outline

- Brief review of current curriculum
- Structure of new curriculum
- CS Core courses
- Initial set of tracks
- Vote

Timeline

- Spring 2006/07: Preliminary agreement on structure.
- Fall 2007/08: Complete definition of core.
- Winter 2007/08: Define initial set of tracks. 
  [Vote to adopt new curriculum.]
  Start transition plan for new courses.
- Spring 2007/08: Complete transition plan.
  Publicize new program to students.
- Fall 2008/09: New curriculum requirements in place.
  Begin shift in core course content.
- Winter 2008/09+: Continued development of course contents and new courses.

Thank you for your attention

Questions before voting

Vote! (early and often)
Additional Material

Issues We’ve Considered

- Undergrad program isn’t broken, why fix it?
  - Field has evolved more significantly than curriculum in last 15 years, and will continue to do so
  - Modularity allows for easier evolution of curriculum with the field
  - “Footprint” of CS has potential to be much larger by explicitly providing more options and inter-disciplinary ties

- Students won’t know what to take
  - They don’t need to know right away (common core)
  - Can provide effective advising from faculty and staff

- What if very few students take a particular track?
  - Some MS concentrations are small, but still available
  - Provides useful feedback for future revisions

More Issues We’ve Considered

- Students will take fewer systems courses and potentially have weaker programming skills
  - There is still a substantial amount of programming in curriculum
  - Students will still be well prepared for a broad array of work
  - Not all students want to take high-intensity programming jobs
  - For students who do, the Systems track is likely to be popular
  - Note: Currently, 47% of MS students take Systems specialization

Current Curriculum: Additional SoE Reqs.

- Mathematics
  - Two quarters Calculus (many students receive AP credit)
  - One quarter Probability
  - Two restricted Math electives

- Science
  - Two quarters Physics (Mechanics, Elec. & Magnetism)
  - One restricted Science elective

- Engineering Fundamentals
  - E40: Introductory Electronics
  - One Engineering Fundamental Elective

- Technology in Society (One restricted elective course)