ACM/IEEE-CS Computer Science Curricula 2013 (CS2013)

Zachary Dodds

Harvey Mudd College

dodds@cs.hmc.edu

Mehran Sahami

Stanford University

sahami@cs.stanford.edu

Outline

- Computing Curriculum History
- The CS2013 Effort
  - Charter
  - Themes and Principles
- Survey of Curricular Document Usage
- Plans for CS2013
- Get Involved!

Computing Curriculum History

- Every decade, ACM and IEEE-Computer Society jointly sponsor a curricular volume on Computer Science
  - Aimed at providing modern curricular guidance for undergraduate Computer Science programs internationally
  - Previous volumes in 1968, 1978, 1991, and 2001
  - Starting in 2001, volumes splits by disciplines:
    - Computer Science (CS), Computer Engineering (CE), Information Systems (IS), Information Technology (IT), and Software Engineering (SE)
  - Modest “interim” revision of CS volume was released in 2008
- Next full CS volume is set for release in 2013
  - Hence the name “CS2013”
  - Work on this volume began in Fall 2010

Contents of Curricular Volumes

- Guiding principles and rationale
- Body of Knowledge
  - Topically organized set of “Knowledge Areas”
  - Knowledge Areas provide list of topics and learning outcomes
- Curricular structure and sample courses
  - Guidance on how to put together a complete curriculum
  - Outlines of stylized classes covering Body of Knowledge
  - Institutional challenges
- Professional considerations
  - Characteristics of CS graduates
  - Professional practice

Knowledge Areas in CS2013

- AL - Algorithms and Complexity
- AR - Architecture and Organization
- CN - Computational Science
- DS - Discrete Structures
- GV - Graphics and Visual Computing
- HC - Human-Computer Interaction
- IAS - Information Assurance and Security
- IM - Information Management
- **IS - Intelligent Systems**
- NC - Networking and Communications
- OS - Operating Systems
- PBD - Platform-based Development
- PD - Parallel and Distributed Computing
- PL - Programming Languages
- SDF - Software Development Fundamentals
- SE - Software Engineering
- SF - System Fundamentals
- SP - Social and Professional Issues

CS2013 Charter

To review the Joint ACM and IEEE/CS Computer Science volume of Computing Curricula 2001 and the accompanying interim review CS 2008, and develop a revised and enhanced version for the year 2013 that will match the latest developments in the discipline and have lasting impact.

The CS2013 task force will seek input from a diverse audience with the goal of broadening participation in computer science. The report will seek to be international in scope and offer curricular and pedagogical guidance applicable to a wide range of institutions. The process of producing the final report will include multiple opportunities for public consultation and scrutiny.

CS2013 Steering Committee

ACM

- Mehran Sahami, Chair (Stanford)
- Andrea Danyluk (Williams College)
- Sally Fincher (Univ. of Kent)
- Kathleen Fisher (Tufts University)
- Dan Grossman (Univ. of Washington)
- Beth Hawthorne (Union County Coll.)
- Randy Katz (UC Berkeley)
- Rich LeBlanc (Seattle Univ.)
- Dave Reed (Creighton)

IEEE-CS

- Steve Roach, Chair (UT, El Paso)
- Ernesto Cuadros-Vargas (Universidad Católica San Pablo, Peru)
- Ronald Dodge (US Military Academy)
- Robert France (Colorado State)
- Amruth Kumar (Ramapo College of NJ)
- Brian Robinson (ABB corporation)
- Remzi Seker (U. of Arkansas, Little Rock)
- Alfred Thompson (Microsoft)

High-Level Themes of CS2013 Effort

- "Big Tent" view of Computer Science
  - "Outward" looking view of the field
  - Able to bridge to multi-disciplinary work ("Computational X")
- Managing curriculum size
  - CS2001 reduced total required hours from CC'91
  - Aim to not increase required hours from CS2001
- Course exemplars as opposed to stylized courses
  - Pointers to existing courses that incorporate knowledge units
  - Not creating a set of reference classes
- Be aware of institutional needs
  - Variable goals, resources, and constraints
  - Variety of school sizes, school types, and available resources

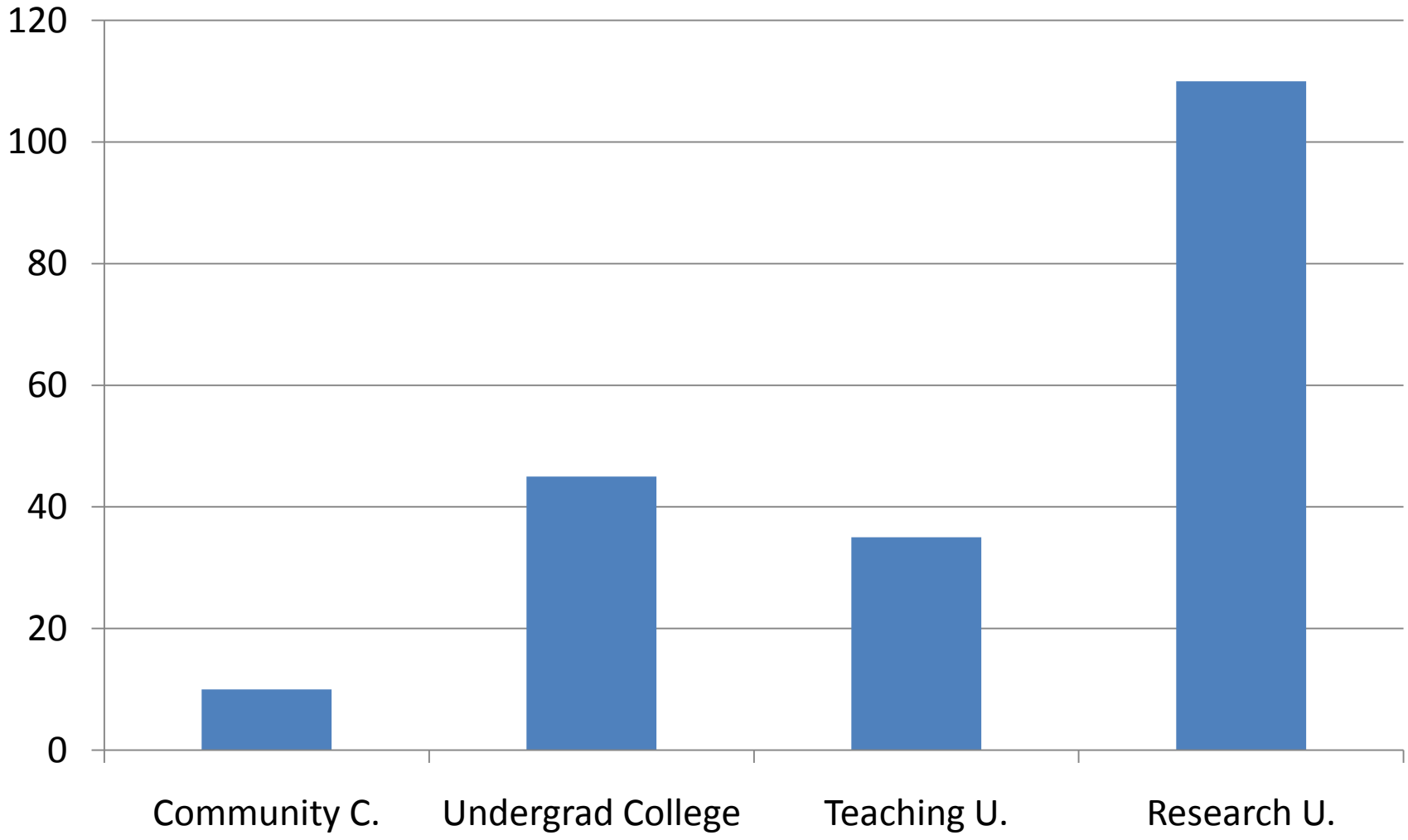
Principles for CS2013

1. Identify essential skills and body of knowledge for CS undergraduates.
2. CS is rapidly changing field, drawing from and contributing to variety of disciplines. Must prepare students for lifelong learning.
3. CS2013 is serving many constituents, including: faculty, students, administrators, curricula developers, and industry.
4. Curricular guidelines must be relevant to a variety of institution types (large/small, research/teaching, 4-yr/2-yr, US/int'l)
5. Provide guidance for level of mastery for topics, and show exemplars of fielded courses covering topics.
6. Provide realistic, adoptable recommendations that support novel curricular designs, and attract full range of talent to field.
7. Should include professional practice (e.g. communication skills, teamwork, ethics) as components of undergraduate experience.

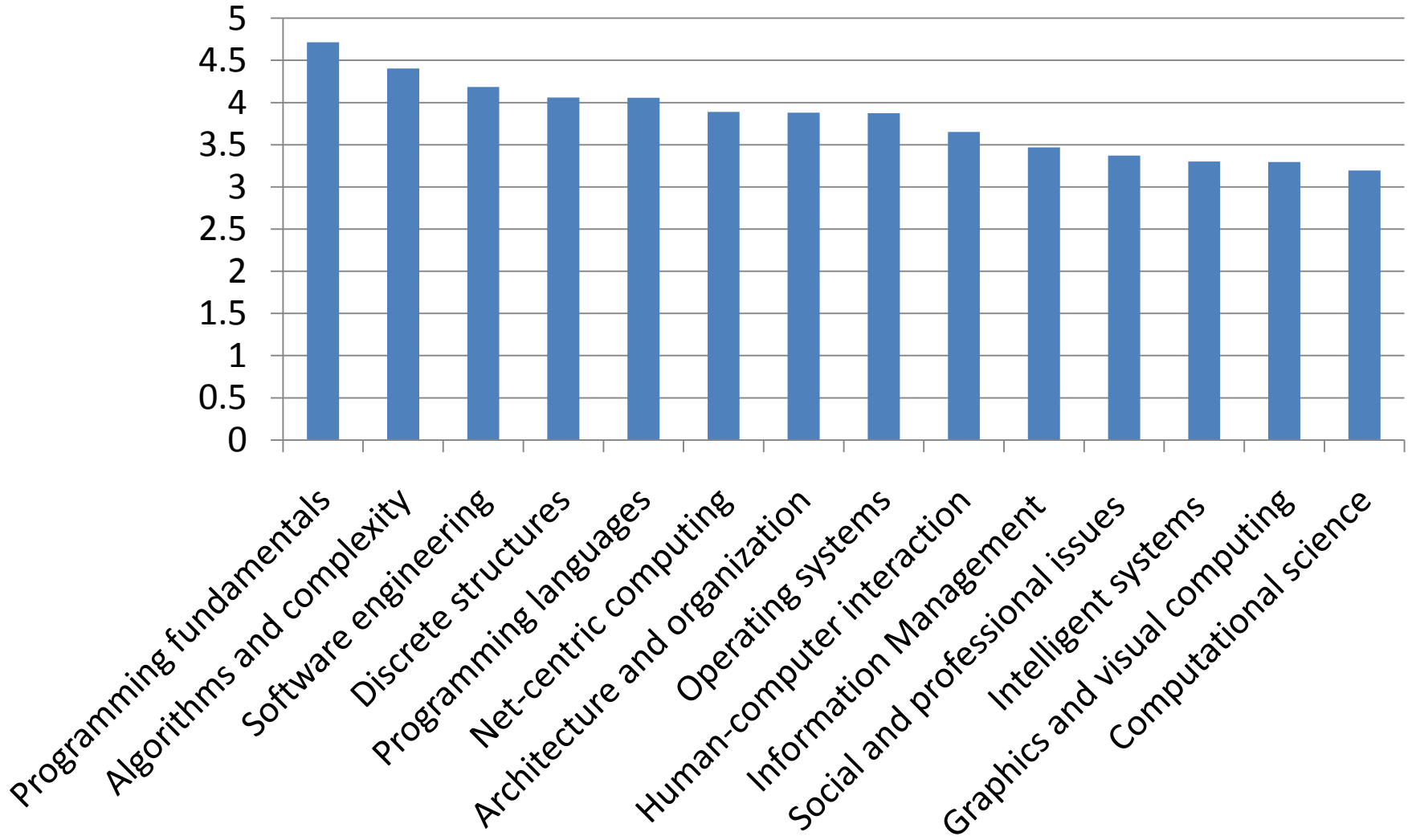
Survey of Curricular Document Usage

- Developed survey to gather data for CS2013
  - Reviews usage of CC2001 and CS2008
  - Rating of importance of existing knowledge areas
  - Rating of principles (e.g., importance of stylized classes)
  - Suggestions for new topics of import/knowledge areas
- Survey released in December, 2010
  - ~1500 US department chairs/directors of UG education
  - ~2000 International department chairs
  - Received 201 responses

Type of School



Importance of Knowledge Areas



“Coming Attractions” in CS2013 (Part 1)

- Reorganization of topics in many Knowledge Areas
  - Notably, includes a reworking of topics in Programming Fundamentals, Programming Languages, and Algorithms
    - Move paradigm-specific concepts (e.g., OOP, Functional) to Programming Languages
- Addition of new Knowledge Areas
  - Parallel and Distributed Computing
  - Information Assurance and Security
  - Systems Fundamentals
    - Analogous to Programming Fundamentals but for systems
- Provide references to exemplar courses/curricula to show pathways for implementing Body of Knowledge
  - Replaces definitions of stylized courses in previous reports

“Coming Attractions” in CS2013 (Part 2)

- Three-tiered classification of Body of Knowledge Units
  - Core (Tier 1): absolutely essential topics, all of which are required for any undergraduate CS program
  - Core (Tier 2): important foundational topics, the vast majority (80-90%) of which should be in an undergrad CS program
    - Still considered “Core” topics
    - Tiering allow for prioritization and more flexibility for local customization of CS curricula
  - Elective: additional topics that can be included to complete an undergraduate CS program
- Provide guidance on depth of coverage for topics
  - 3 levels of depth: Knowledge, Application, and Evaluation

Timeline

- Fall 2010: Steering Committee formed
  - Subcommittees formed to review Knowledge Areas
- Winter/Spring/Summer 2011: Work on Body of Knowledge
- August 2011: EAAI-11 panel
  - Panels are various venues (SIGCSE, FCRC, FIE, etc.)
- December 2011: (Preliminary) Strawman draft of CS2013
  - First draft of Body of Knowledge
  - Circulate for comments to community
- Winter/Summer 2012: Committee meetings
  - Continue incorporation of feedback
  - Further hone report
- Fall 2012: Stoneman draft of CS2013
- Summer 2013: Final CS2013 Report

Getting Involved!

- Multiple opportunities to get involved in this effort
  - Review and comment on draft Knowledge Areas
  - Suggest a role that you see can contributing to this effort
- Website: **`cs2013.org`**
  - Plan to use these as means for (draft) report dissemination and community engagement
- Email Mehran Sahami and Steve Roach to get involved

  `sahami@cs.stanford.edu` and **`sroach@utep.edu`**
- Specifically looking for feedback on Intelligent Systems (IS) area today
  - Break-out activity

Intelligent Systems (IS)

IS == AI

CS 2013 - IS Draft Post-Review

Intelligent Systems (IS)

The field of artificial intelligence (AI) is concerned with the design and analysis of autonomous agents. ...

Current Outline

IS. Intelligent Systems: 2013

Tier 2 Core: 10 h.

IS/Fundamental Issues (1 hour)

IS/Basic Search (4 hours)
IS/Basic Representation and Reasoning (3 hours)
IS/Basic Machine Learning (2 hours)

Electives

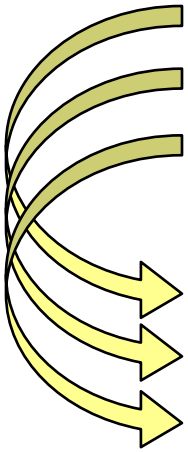
IS/Advanced Search
IS/Advanced Representation and Reasoning
IS/Advanced Machine Learning
IS/Reasoning Under Uncertainty
IS/Agents
IS/Natural Language Processing
IS/Robotics
IS/Perception and Computer Vision

Current Outline

IS. Intelligent Systems: 2013

Tier 2 Core: 10 h.

IS/Fundamental Issues (1 hour)



IS/Basic Search (4 hours)
IS/Basic Representation and Reasoning (3 hours)
IS/Basic Machine Learning (2 hours)

Electives

IS/Advanced Search
IS/Advanced Representation and Reasoning
IS/Advanced Machine Learning
IS/Reasoning Under Uncertainty
IS/Agents
IS/Natural Language Processing
IS/Robotics
IS/Perception and Computer Vision

Current Outline

IS. Intelligent Systems: 2013

2008		
1 hour	IS/Fundamental Issues	(1 hour)
5 hours	IS/Basic Search	(4 hours)
4 hours	IS/Basic Representation and Reasoning	(3 hours)
elective	IS/Basic Machine Learning	(2 hours)
elective	IS/Advanced Search	
elective	IS/Advanced Representation and Reasoning	
not present	IS/Advanced Machine Learning	
planning systems	IS/Reasoning Under Uncertainty	
elective	IS/Agents	
elective	IS/Natural Language Processing	
elective	IS/Robotics	
elective	IS/Perception and Computer Vision	

Tier 2 Core: 10 h.

Electives

Current Outline

IS. Intelligent Systems: 2013

2008	
1 hour	IS/Fundamental Issues (1 hour)
5 hours	IS/Basic Search (4 hours)
4 hours	IS/Basic Representation and Reasoning (3 hours)
elective	IS/Basic Machine Learning (2 hours)
elective	IS/Advanced Search
elective	IS/Advanced Representation and Reasoning
not present	IS/Advanced Machine Learning
planning systems	IS/Reasoning Under Uncertainty
elective	IS/Agents
elective	IS/Natural Language Processing
elective	IS/Robotics
elective	IS/Perception and Computer Vision

Example: *Basic Machine Learning*

IS/Basic Machine Learning [Core-Tier II]
Minimum core *coverage time:* ***2 hours***

Time (if any)

Topics:

- Definition and examples of machine learning for classification
- Inductive learning
- Simple statistical-based learning, Naive Bayesian Classifier
- Measuring classifier accuracy

Bulleted list of topics

Learning Outcomes:

1. Identify examples of classification tasks, including the available input features and output to be predicted. [***knowledge***]
2. Explain the difference between inductive and deductive learning. [***knowledge***]
3. Apply the Naive Bayesian Classifier to a classification task and measure the classifier's accuracy. [***application***]

Learning outcomes & Bloom levels

Break-out groups

Join up with 2-3 others.

Agree on 3 (or so) knowledge units of interest to the group:

IS/Fundamental Issues (1 hour)
IS/Basic Search (4 hours)
IS/Basic Representation and Reasoning (3 hours)
IS/Basic Machine Learning (2 hours)

IS/Advanced Search
IS/Advanced Representation and Reasoning
IS/Advanced Machine Learning
IS/Reasoning Under Uncertainty
IS/Agents
IS/Natural Language Processing
IS/Robotics
IS/Perception and Computer Vision

Look over those topic lists and the learning outcomes...

Discuss!

1) IS/Fundamental Issues [Core - Tier II]

Minimum core *coverage time: 1 hour*

Topics:

- Overview of AI problems, Examples of successful recent AI applications
- Philosophical questions (1) The Turing test (2) Searle's Chinese Room, (3) Ethical issues in AI
- Nature of environments
  (1) Fully vs partially observable (2) Single- vs multi-agent (3) Deterministic vs stochastic
  (4) Episodic vs sequential (5) Static vs dynamic (6) Discrete vs continuous
- Nature of agents
  (1) Autonomous vs Semi-Autonomous, (2) Reflexive, (3) Goal-based, and (4) Utility-based
- Fundamental definitions
- Rational vs non-rational reasoning
- Nature of human reasoning
- AI Programming = Representation + Reasoning

Learning Outcomes:

1. ***Describe*** Turing test and the "Chinese Room" thought experiment. [Knowledge]
2. ***Differentiate*** between the concepts of optimal reasoning/behavior and human-like reasoning/behavior. [Knowledge]
3. ***Describe*** a given problem domain using the characteristics of the environments in which intelligent systems must function. [Evaluation]