Computational Thinking

Jeannette M. Wing

President's Professor of Computer Science and Department Head
Computer Science Department
Carnegie Mellon University

OurCS Workshop
Carnegie Mellon University
4 March 2011

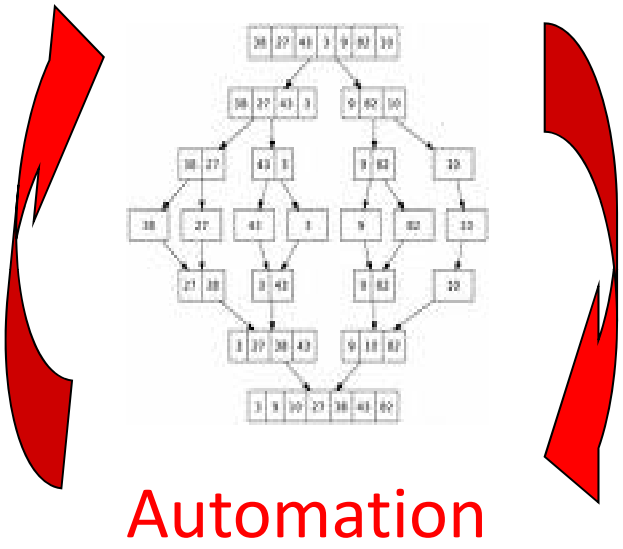
My Grand Vision

- Computational thinking will be a fundamental skill used by everyone in the world by the middle of the 21st Century.
  - Just like reading, writing, and arithmetic.
  - Incestuous: Computing and computers will enable the spread of computational thinking.
  - In research: scientists, engineers, …, historians, artists
  - In education: K-12 students and teachers, undergrads, …

J.M. Wing, "Computational Thinking," *CACM* Viewpoint, March 2006, pp. 33-35.
Paper off http://www.cs.cmu.edu/~wing/

Computing is the Automation of Abstractions

Abstractions



Automation

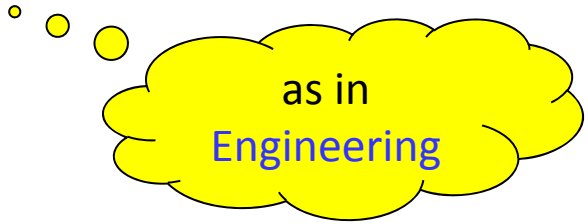
Computational Thinking focuses on the process of abstraction

- choosing the right abstractions
- operating in terms of multiple layers of abstraction simulta
- defining the relationships the between layers

as in Mathematics

guided by the following concerns…

Measures of a "Good" Abstraction in C.T.



- Efficiency
  - How fast?
  - How much space?
  - How much power?
- Correctness
  - Does it do the right thing?
    - Does the program compute the right answer?
  - Does it do anything?
    - Does the program eventually produce an answer? [Halting Problem]
- -ilities
  - Simplicity and elegance
  - Usability
  - Modifiability
  - Maintainability
  - Cost
  - …

Computational Thinking, Philosophically

- Complements and combines mathematical and engineering thinking
  - C.T. draws on math as its foundations
    - But we are constrained by the physics of the underlying machine
  - C.T. draws on engineering since our systems interact with the real world
    - But we can build virtual worlds unconstrained by physical reality

- Ideas, not artifacts
  - It's not just the software and hardware that touch our daily lives, it will be the computational concepts we use to approach living.

- It's for everyone, everywhere

Sample Classes of Computational Abstractions

- Algorithms
  - E.g., mergesort, binary search, string matching, clustering
- Data Structures
  - E.g., sequences, tables, trees, graphs, networks
- State Machines
  - E.g., finite automata, Turing machines
- Languages
  - E.g., regular expressions, …, VDM, Z, …, ML, Haskell, …, Java, Perl
- Logics and semantics
  - E.g., Hoare triples, temporal logic, modal logics, lambda calculus
- Heuristics
  - E.g., A* (best-first graph search), caching
- Control Structures
  - Parallel/sequential composition, iteration, recursion
- Communication
  - E.g., synchronous/asynchronous, broadcast/P2P, RPC, shared memory/message-passing
- Architectures
  - E.g., layered, hierarchical, pipeline, blackboard, feedback loop, client-server, parallel, distributed
- …

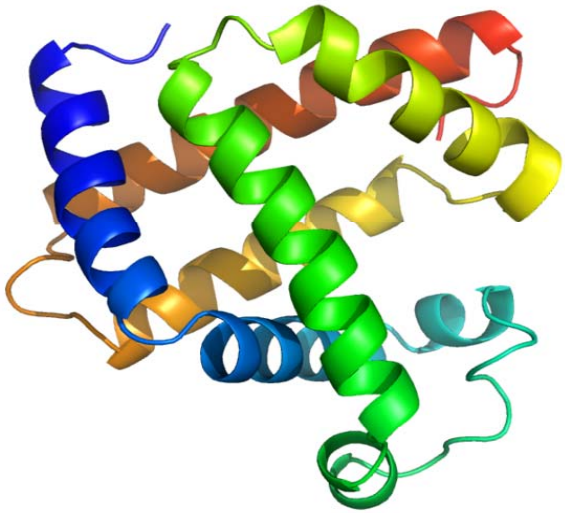
NOT

- Computer literacy, i.e., how to use Word and Excel or even Google
- Computer programming, i.e., beyond Java Programming 101

Examples of Computational Thinking in Other Disciplines

One Discipline, Many Computational Methods

Computational Thinking in Biology

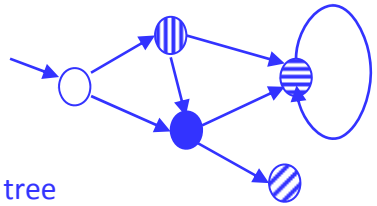


- Shotgun algorithm expedites sequencing of human genome

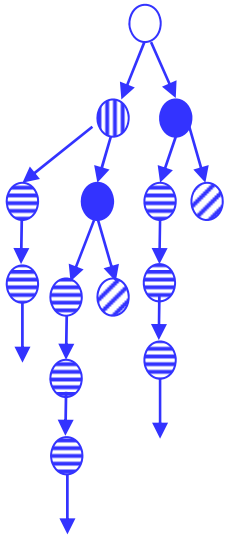
- DNA sequences are strings in a language
- Boolean networks approximate dynamics of biological networks
- Cells as a self-regulatory system are like electronic circuits
- Process calculi model interactions among molecules
- Statecharts used in developmental genetics
- Protein kinetics can be modeled as computational processes
- Robot Adam discovers role of 12 genes in yeast
- PageRank algorithm inspires ecological food web

Model Checking Primer

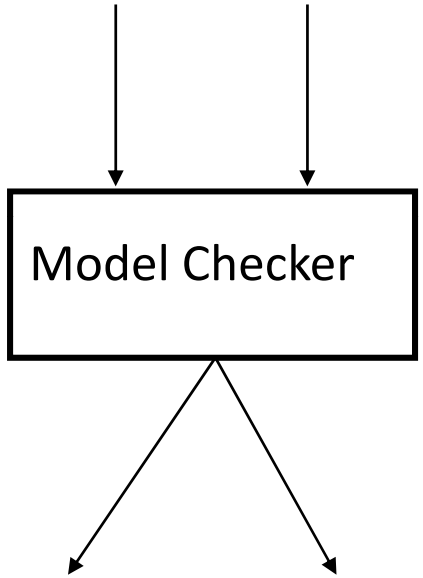
Finite State Machine
model M



M's computational tree



Temporal Logic
property Φ



$\Phi = \mathbf{AG\ p}$

AF p, EG p, EF p

yes

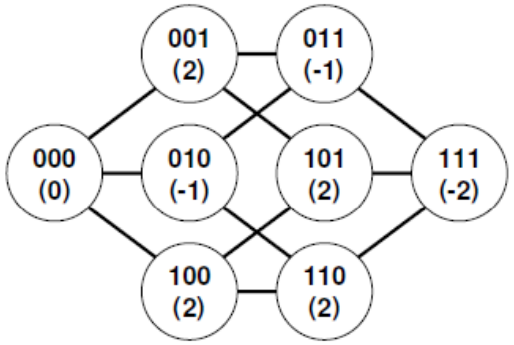
counterexample

Φ is falsified here.

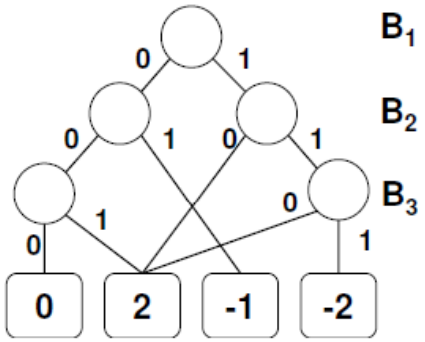
Model Checking in Biology

Goal: Predict Rate of Folding of Proteins

1. Finite State Machine **M** represents 3-residue protein

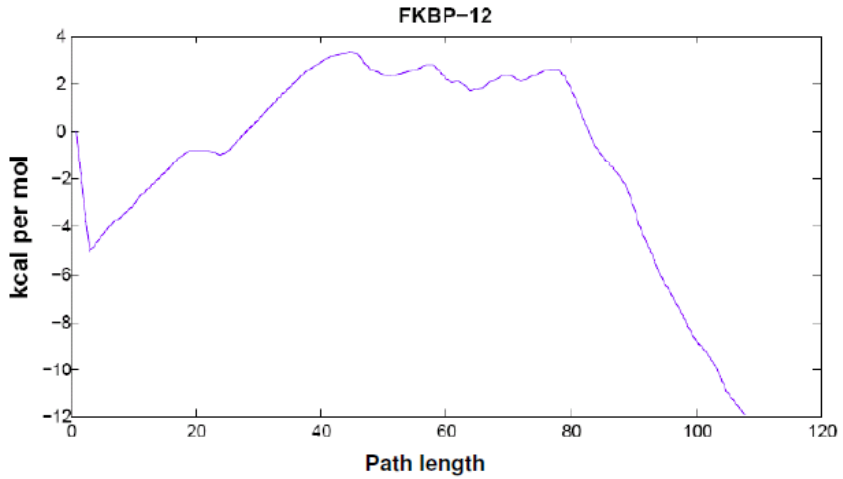


1'. BDD efficiently represents M



2. Temporal Logic Formula Φ
   a. Will the protein end up in a particular configuration?
   b. Will the second residue fold before the first one?
   c. Will the protein fold within *t* ms?
   d. What is the probability that (c)?
   e. Does the state *s* have *k* folded residues and have energy *c*?

Method easily handles proteins up to 76 residues.



Energy Profile for FKBP-12, Computed via Method

Model checking can explore state spaces as large as $2^{76} \approx 10^{23}$, 14 orders of magnitude greater than comparable techniques [LJ07].

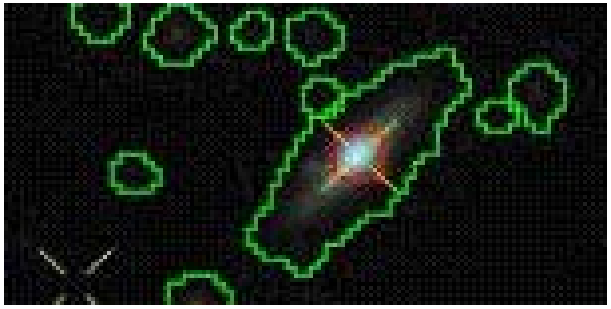
One Computational Method, Many Disciplines

Machine Learning has transformed the field of Statistics.

Machine Learning in the Sciences

Astronomy

- Brown dwarfs and fossil galaxies discovery via machine learning, data mining, data federation
- Very large multi-dimensional datasets analysis using KD-trees



Credit: SDSS

Medicine

- Anti-inflammatory drugs
- Chronic hepatitis
- Mammograms
- Renal and respiratory failure

Meteorology

- Tornado formation

Credit: LiveScience

Neurosciences

- fMRI data analysis to understand language via machine learning

Machine Learning Everywhere

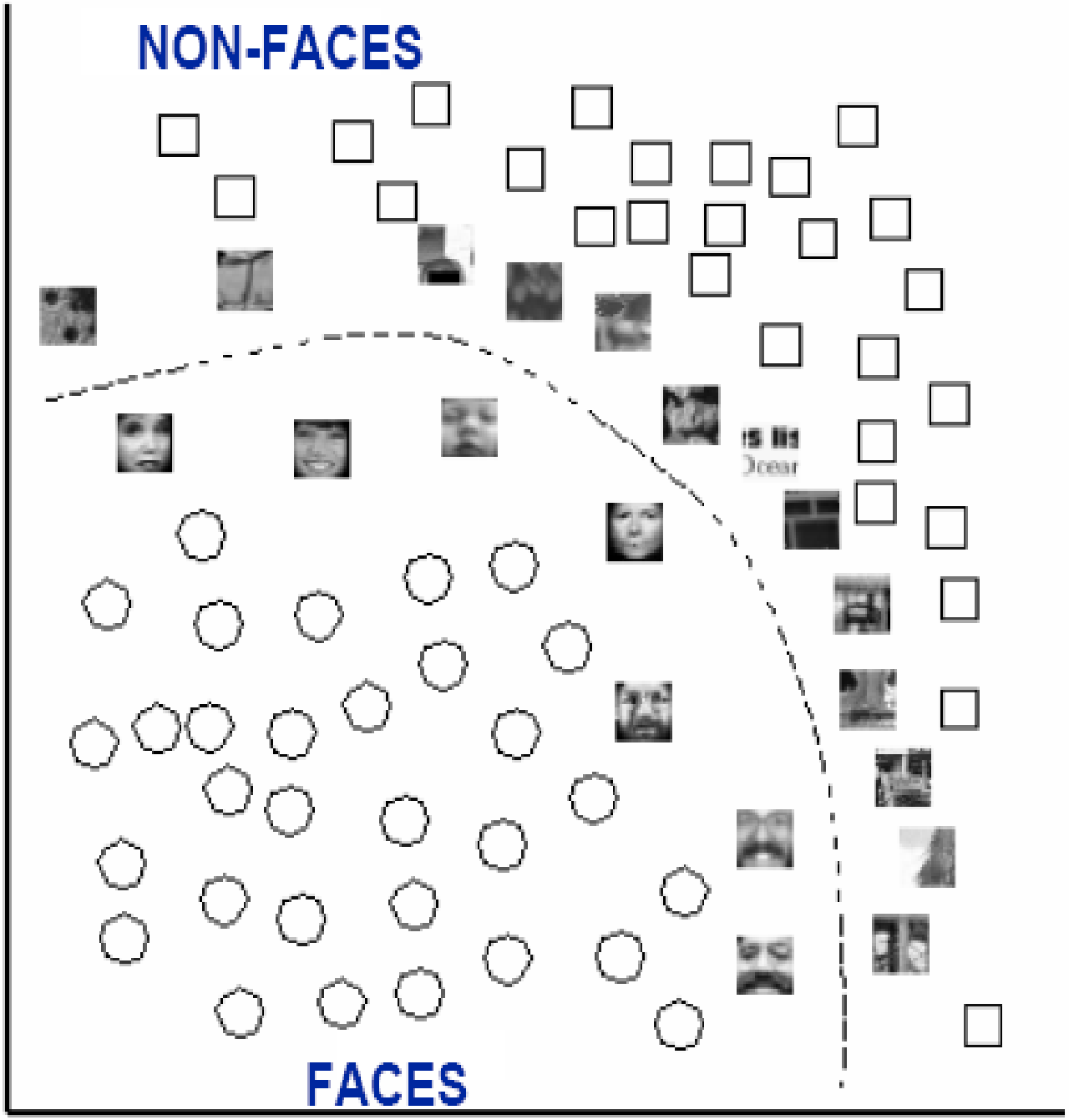
Credit Cards

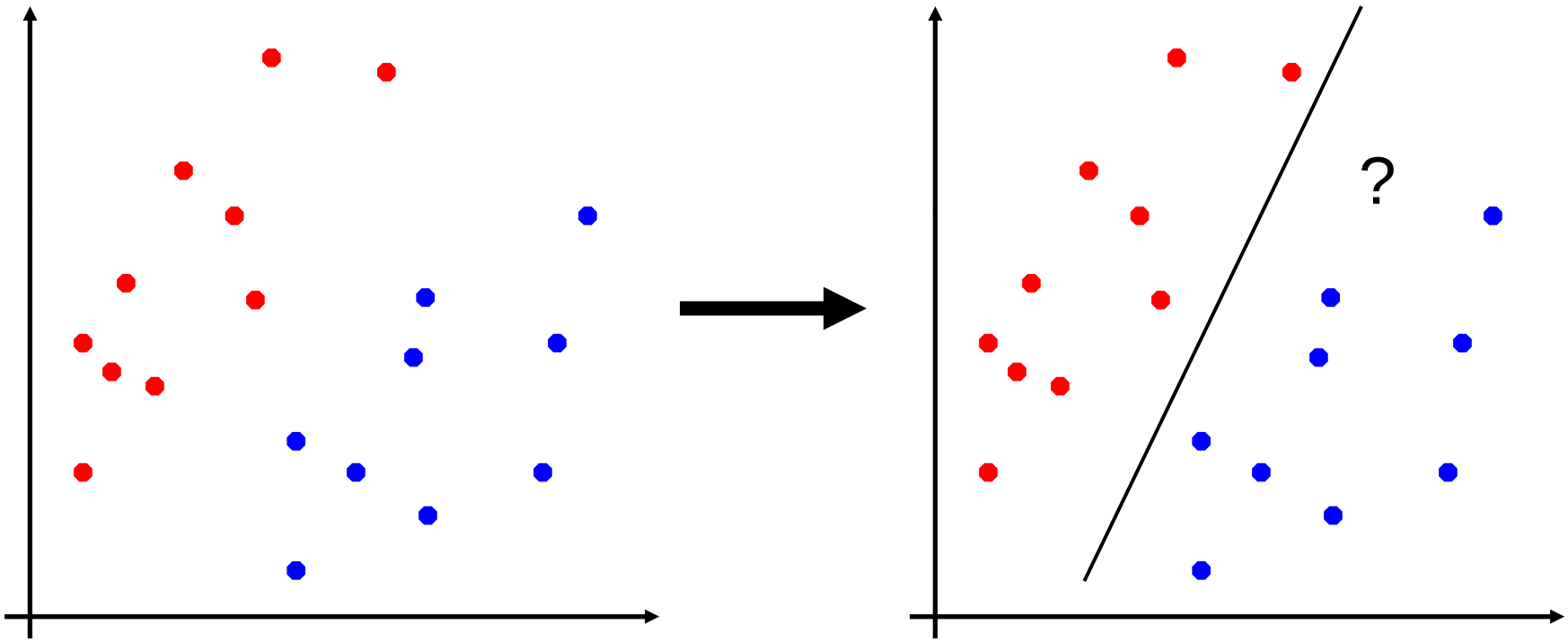
Supermarkets

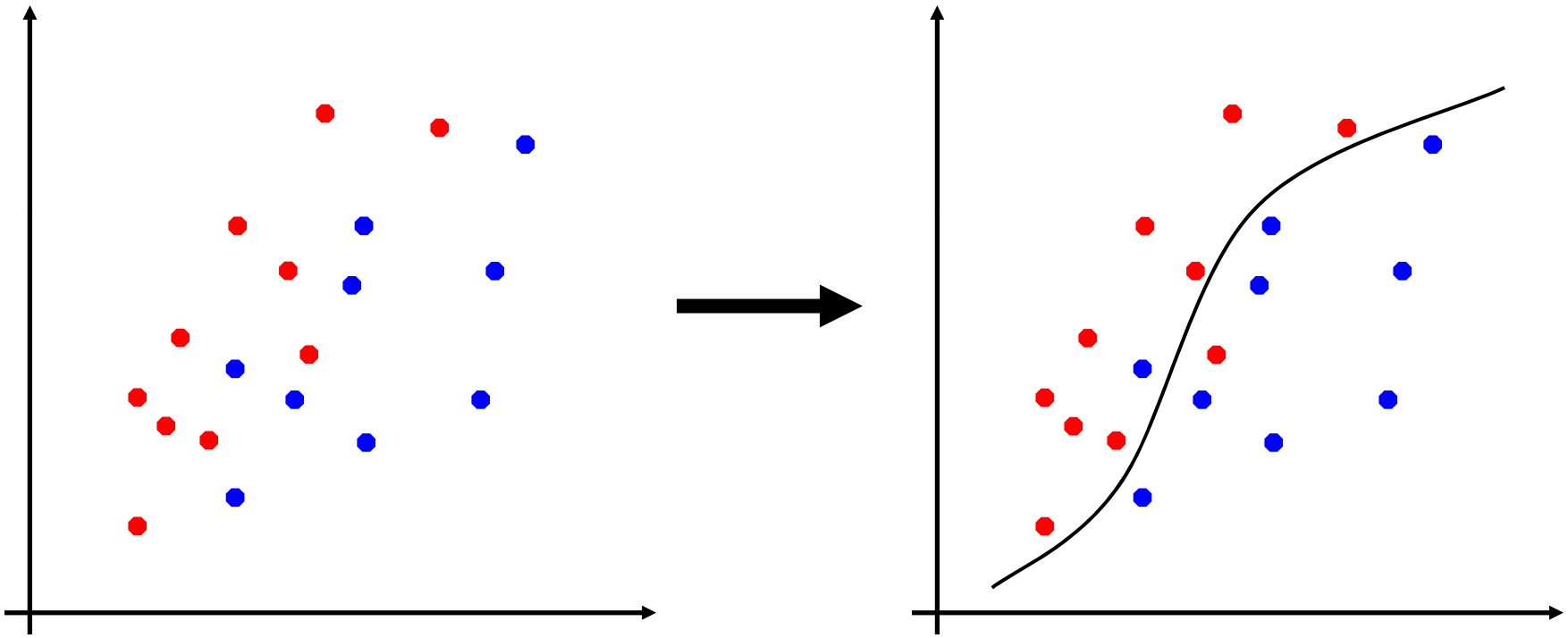
Wall Street

Entertainment:
Shopping, Music, Travel

Sports

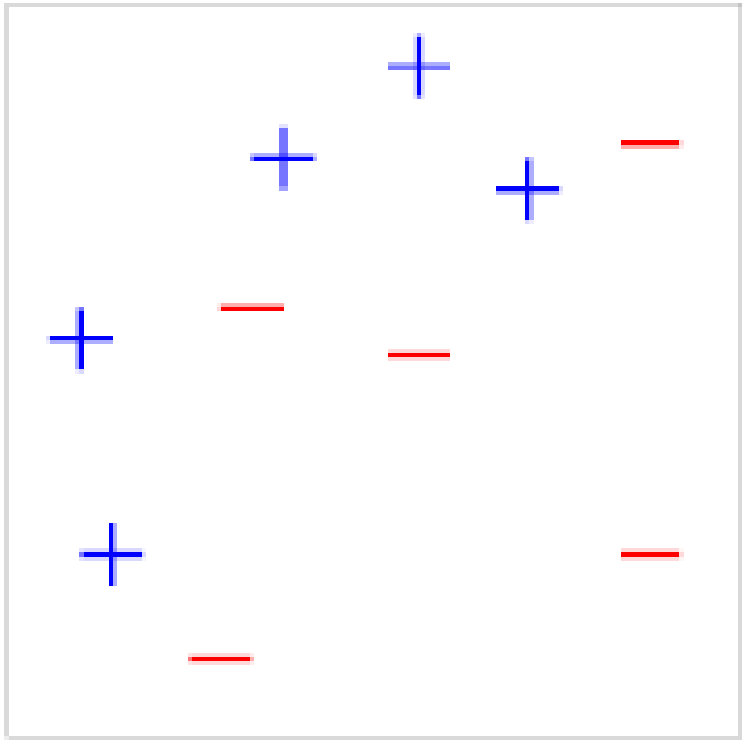

NON-FACES
FACES

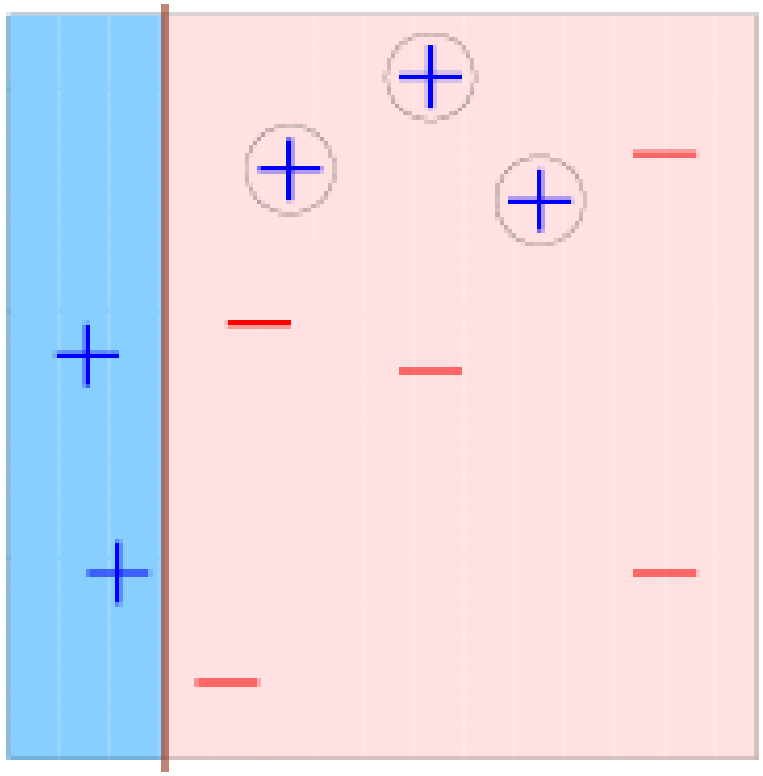


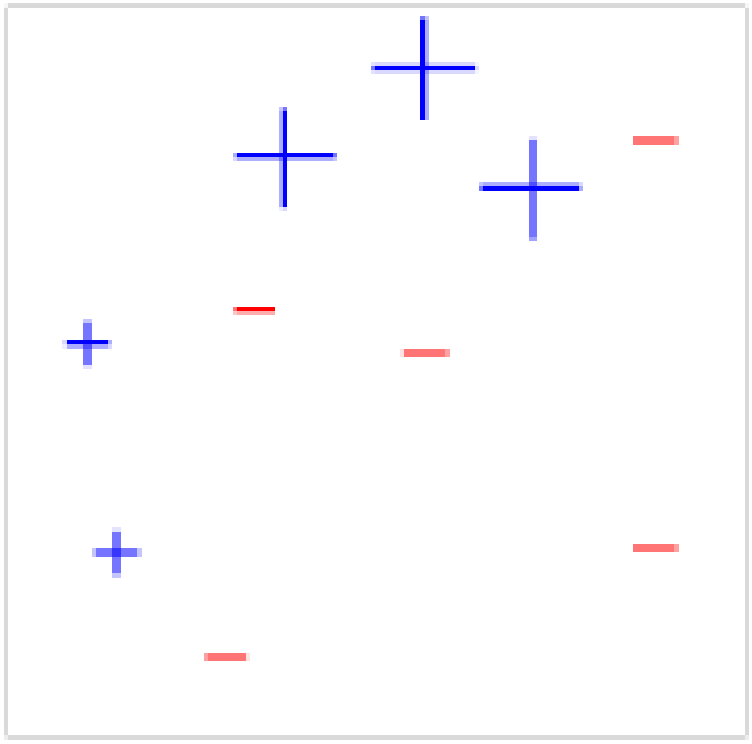


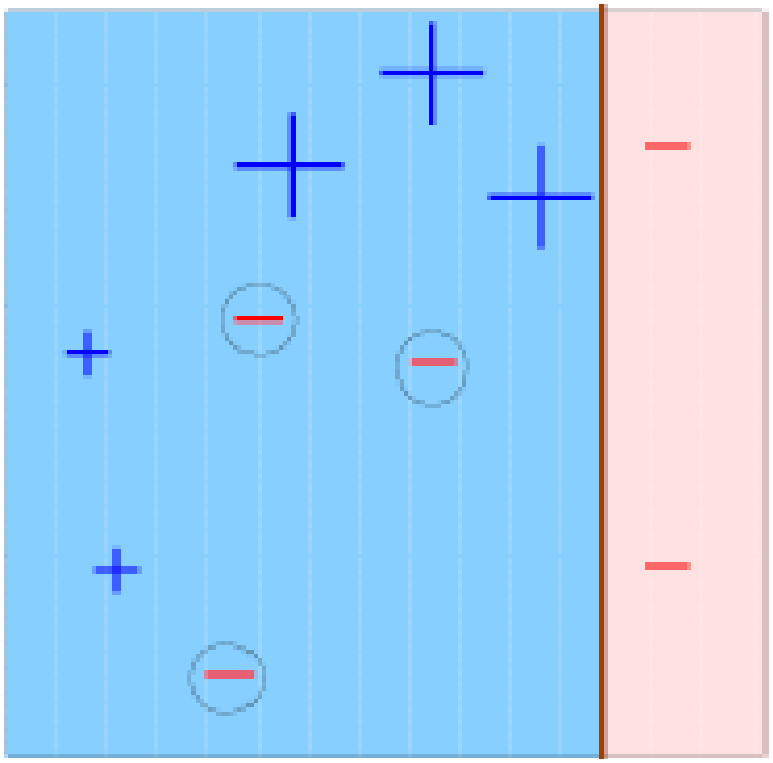
Question (Kearns): Can a Set of Weak Learners Create a Single Strong One?

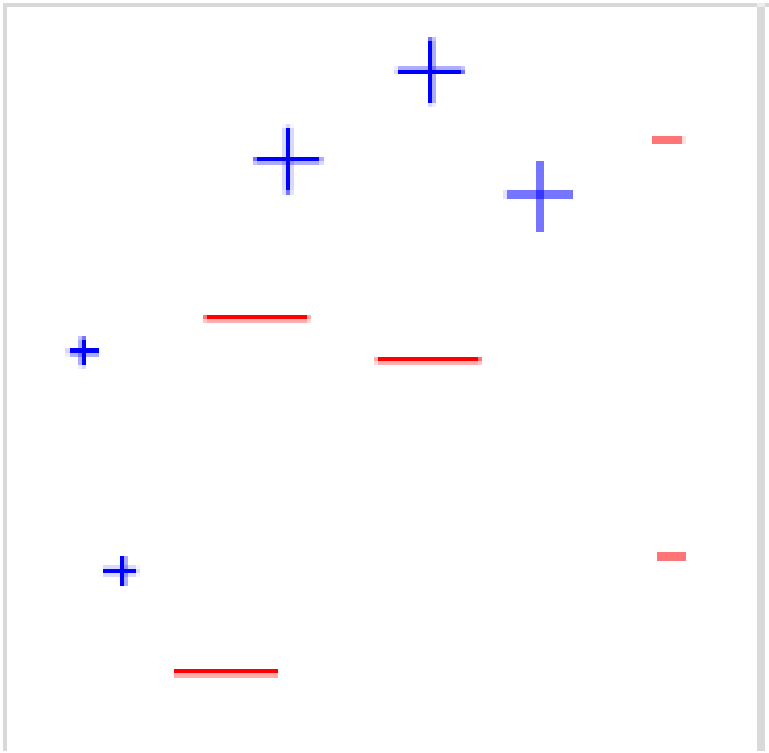
Answer: Yes, by *Boosting* Algorithms (e.g., [FS99])

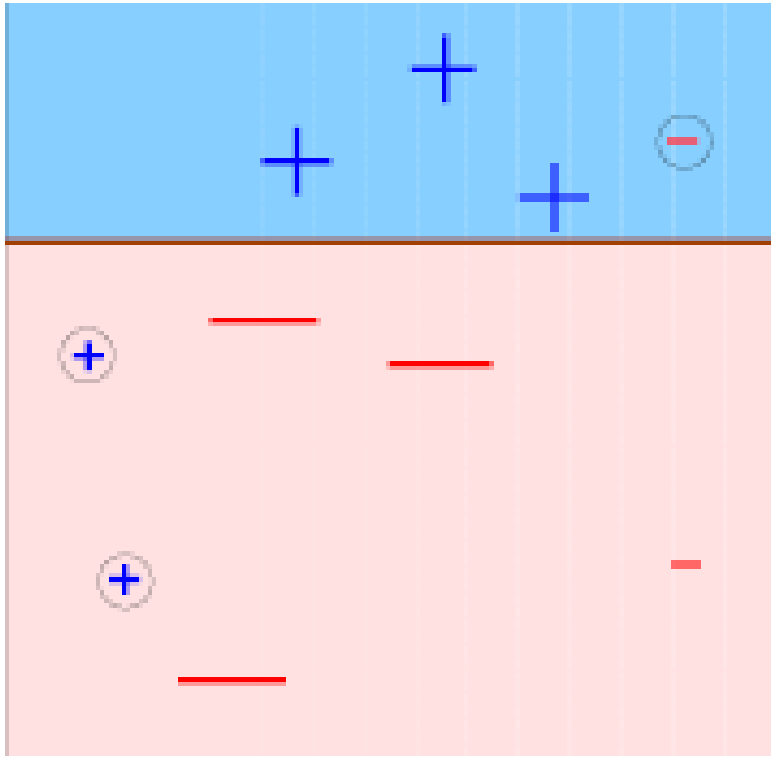


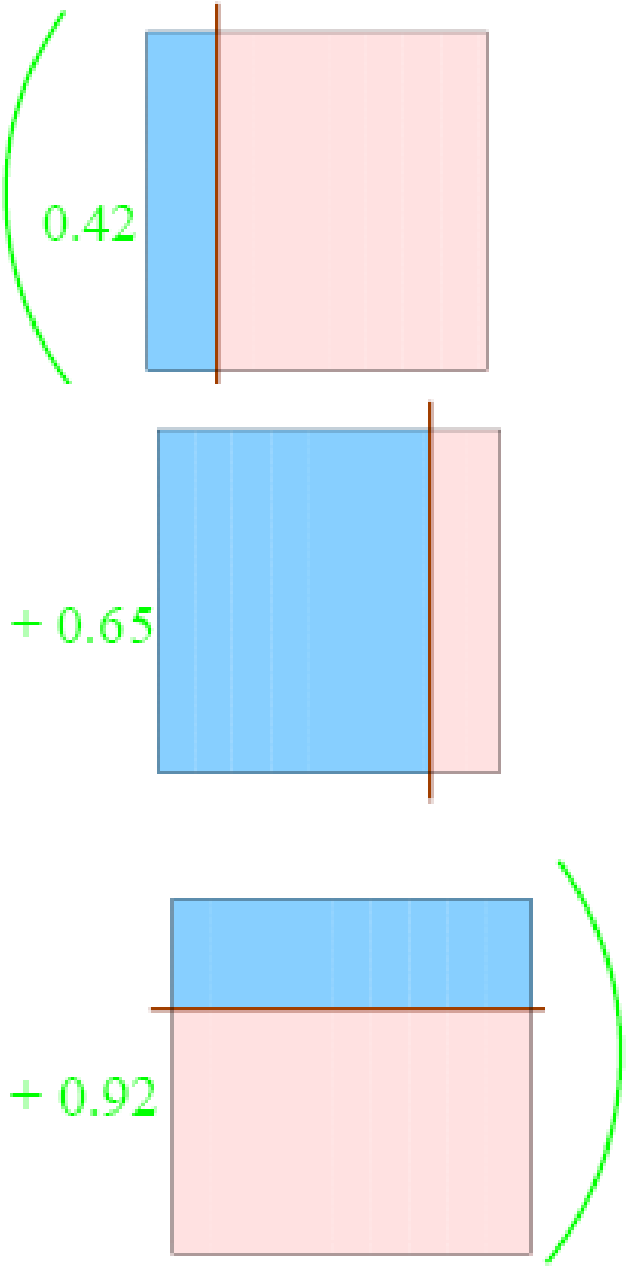


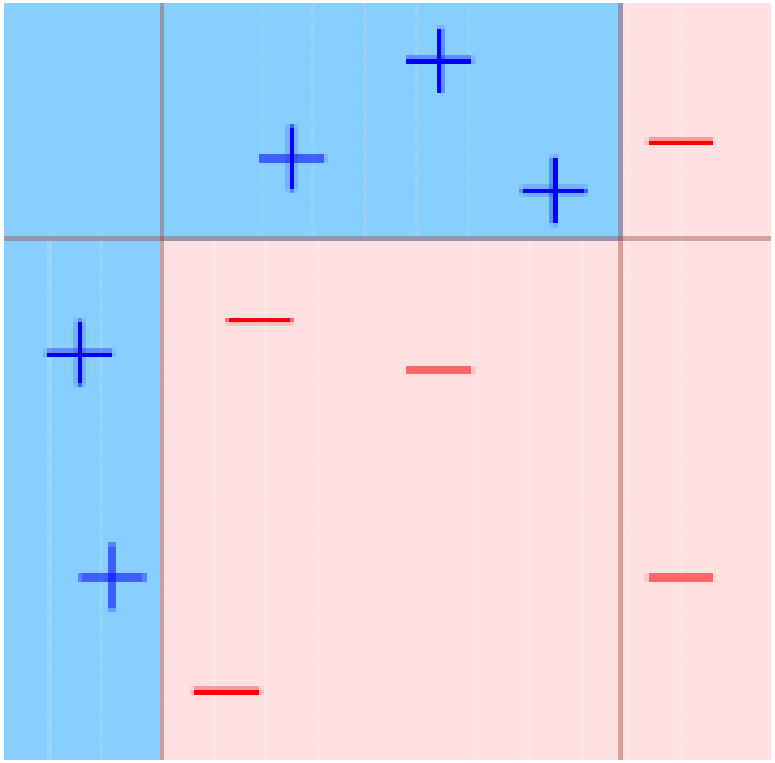








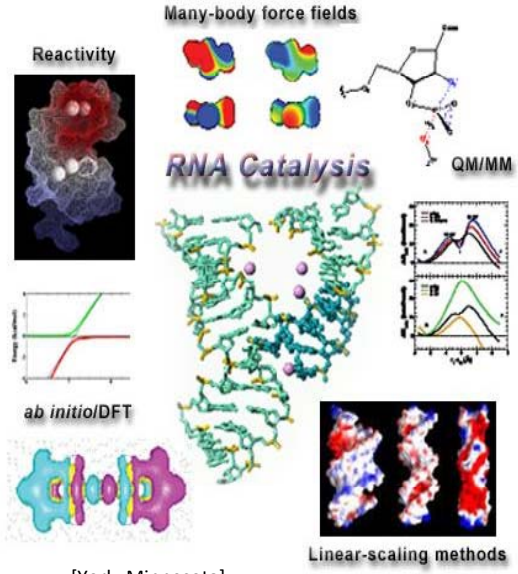

0.42
+ 0.65
+ 0.92



Computational Thinking in the Sciences and Beyond

CT in Other Sciences

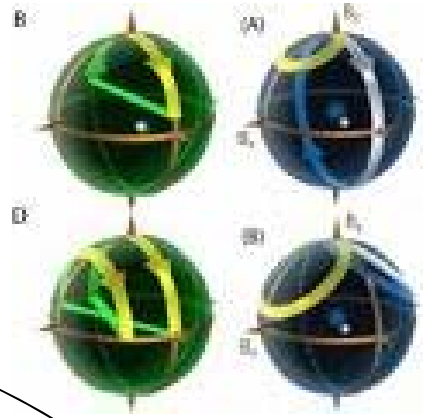
Chemistry



[York, Minnesota]

- Atomistic calculations are used to explore chemical phenomena
- Optimization and searching algorithms identify best chemicals for improving reaction conditions to improve yields

Physics



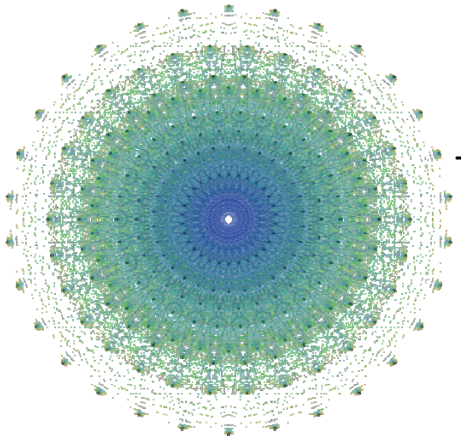
- Adiabatic quantum computing: How quickly is convergence?
- Genetic algorithms discover laws of physics.

Geosciences

- Abstractions for Sky, Sea, Ice, Land, Life, People, etc.
  - Hierarchical, composable , modular, traceability, allowing multiple projections along any dimension, data element, or query
- Well-defined interfaces

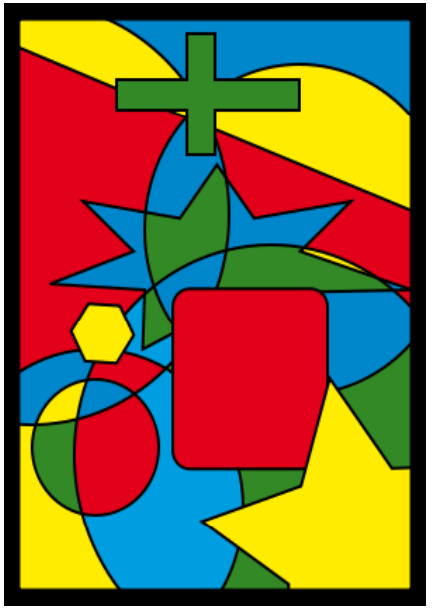
CT in Math and Engineering

Mathematics



Credit: Wikipedia

- Discovering E8 Lie Group:
  18 mathematicians, 4 years and 77 hours of supercomputer time (200 billion numbers).
  Profound implications for physics (string theory)
- Four-color theorem proof



Credit: Wikipedia

Engineering (electrical, civil, mechanical, aero & astro,…)

- Calculating higher order terms implies more precision, which implies reducing weight, waste, costs in fabrication
- Boeing 777 tested via computer simulation alone, not in a wind tunnel

Credit: Boeing

CT for Society

Economics

- Automated mechanism design underlies electronic commerce, e.g., ad placement, on-line auctions, kidney exchange
- Internet marketplace requires revisiting Nash equilibria model
- Use intractability for voting schemes to circumvent impossibility results

Law

- Inventions discovered through automated search are patentable
- Stanford CL approaches include AI, temporal logic, state machines, process algebras, Petri nets
- POIROT Project on fraud investigation is creating a detailed ontology of European law
- Sherlock Project on crime scene investigation

Healthcare

- Algorithmic medicine
- Software design principles and debugging applied to prescriptions of painkillers
- ONC SHARP Program, NSF Smart Health and Wellness Program, NITRD Senior Steering Group on Health IT

CT for Society

Archaeology

- eHeritage Project, Microsoft Research Asia
- Digital Forma Urbis Romae Project, Stanford
- Cathedral Saint Pierre, Columbia

Journalism

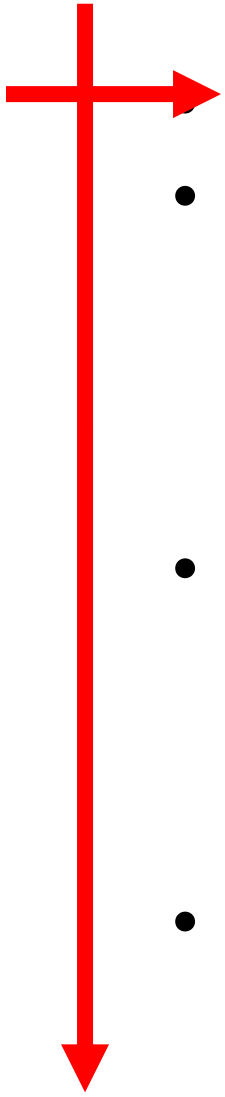
- Crowd sourcing as a new way of getting news tips from sources
- Algorithmic approach to validate credibility of sources
- Digital Media and Learning Initiative, MacArthur Foundation

Humanities

- Digging into Data Challenge: What could you do with a million books?
  Nat'l Endowment for the Humanities (US),
  JISC (UK), SSHRC (Canada)
- Music, English, Art, Design, Photography, …

Educational Implications

Pre-K to Grey



- K-6, 7-9, 10-12
- Undergraduate courses
  - Freshmen year
    - "Ways to Think Like a Computer Scientist" aka Principles of Computing
  - Upper-level courses
- Graduate-level courses
  - Computational arts and sciences
    - E.g., entertainment technology, computational linguistics, …, computational finance, …, computational biology, computational astrophysics
- Post-graduate
  - Executive and continuing education, senior citizens
  - Teachers, not just students

Education Implications for K-12

Question and Challenge for the Computing Community:

What is an effective way of learning (teaching) computational thinking by (to) K-12?

- What concepts can students (educators) best learn (teach) when?
  What is our analogy to numbers in K, algebra in 7, and calculus in 12?

- We uniquely also should ask how best to integrate The Computer
  with teaching the concepts.

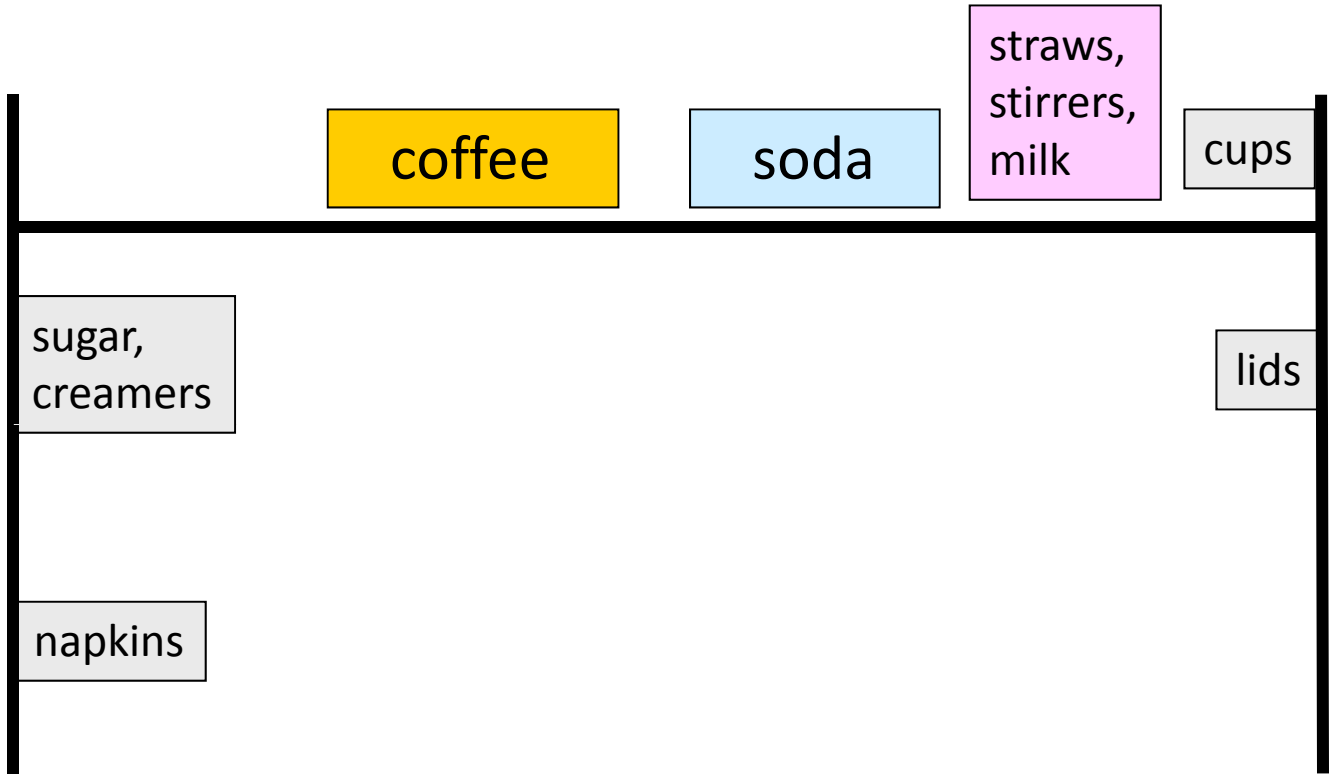
Computer scientists are now working with educators and cognitive learning scientists to address these questions.

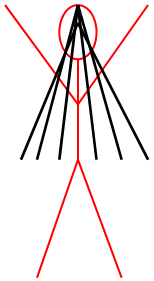
Computational Thinking in Daily Life

Simple Daily Examples

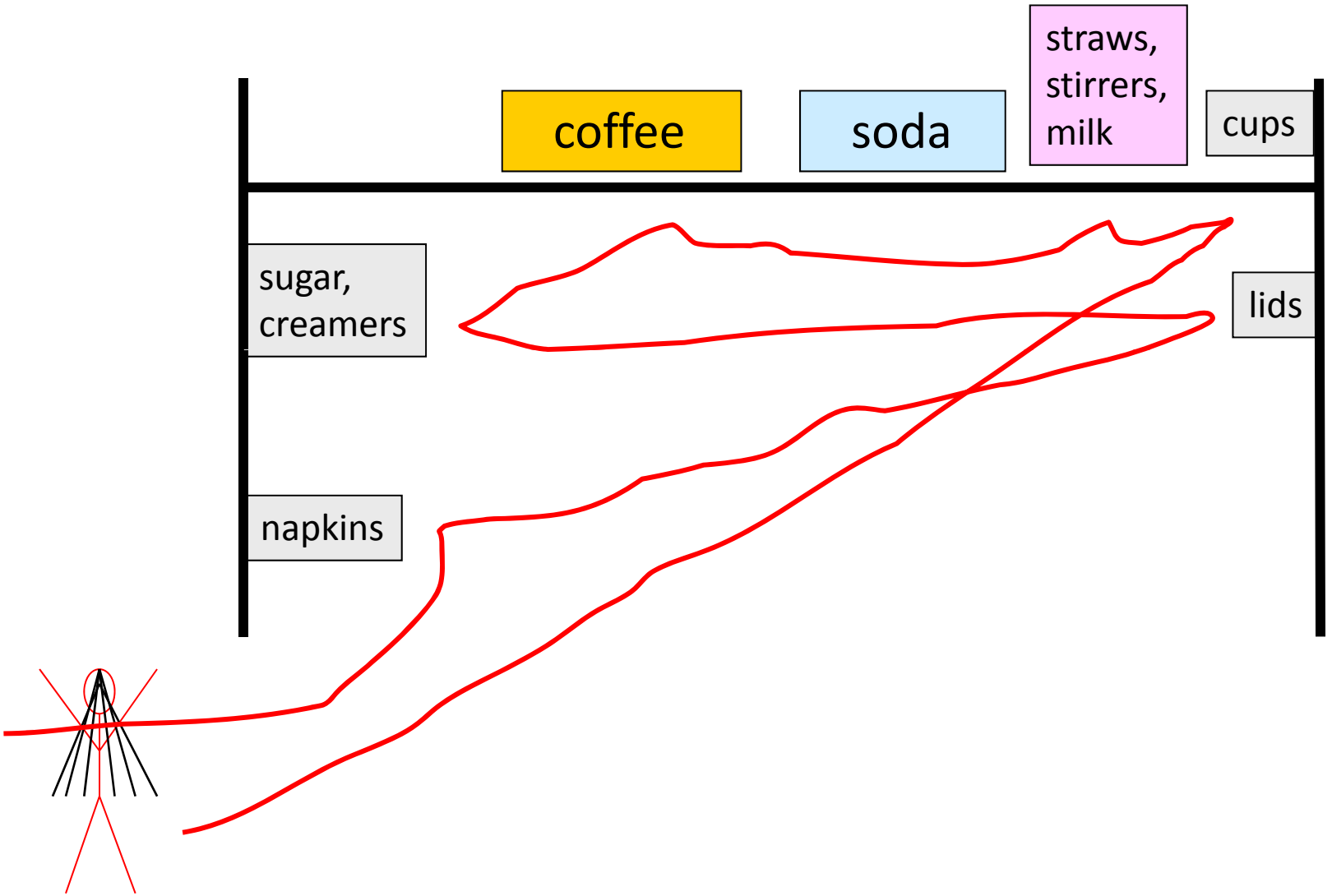
- Looking up a name in an alphabetically sorted list
  - Linear: start at the top
  - Binary search: start in the middle
- Standing in line at a bank, supermarket, customs & immigration
  - Performance analysis of task scheduling
- Putting things in your child's knapsack for the day
  - Pre-fetching and caching
- Taking your kids to soccer, gymnastics, and swim practice
  - Traveling salesman (with more constraints)
- Cooking a gourmet meal
  - Parallel processing: You don't want the meat to get cold while you're cooking the vegetables.
- Cleaning out your garage
  - Keeping only what you need vs. throwing out stuff when you run out of space.
- Storing away your child's Lego pieces scattered on the LR floor
  - Using hashing (e.g., by shape, by color)
- Doing laundry, getting food at a buffet
  - Pipelining the wash, dry, and iron stages; plates, salad, entrée, dessert stations
- Even in grade school, we learn algorithms (long division, factoring, GCD, …) and abstract data types (sets, tables, …).

Getting Morning Coffee at the Cafeteria

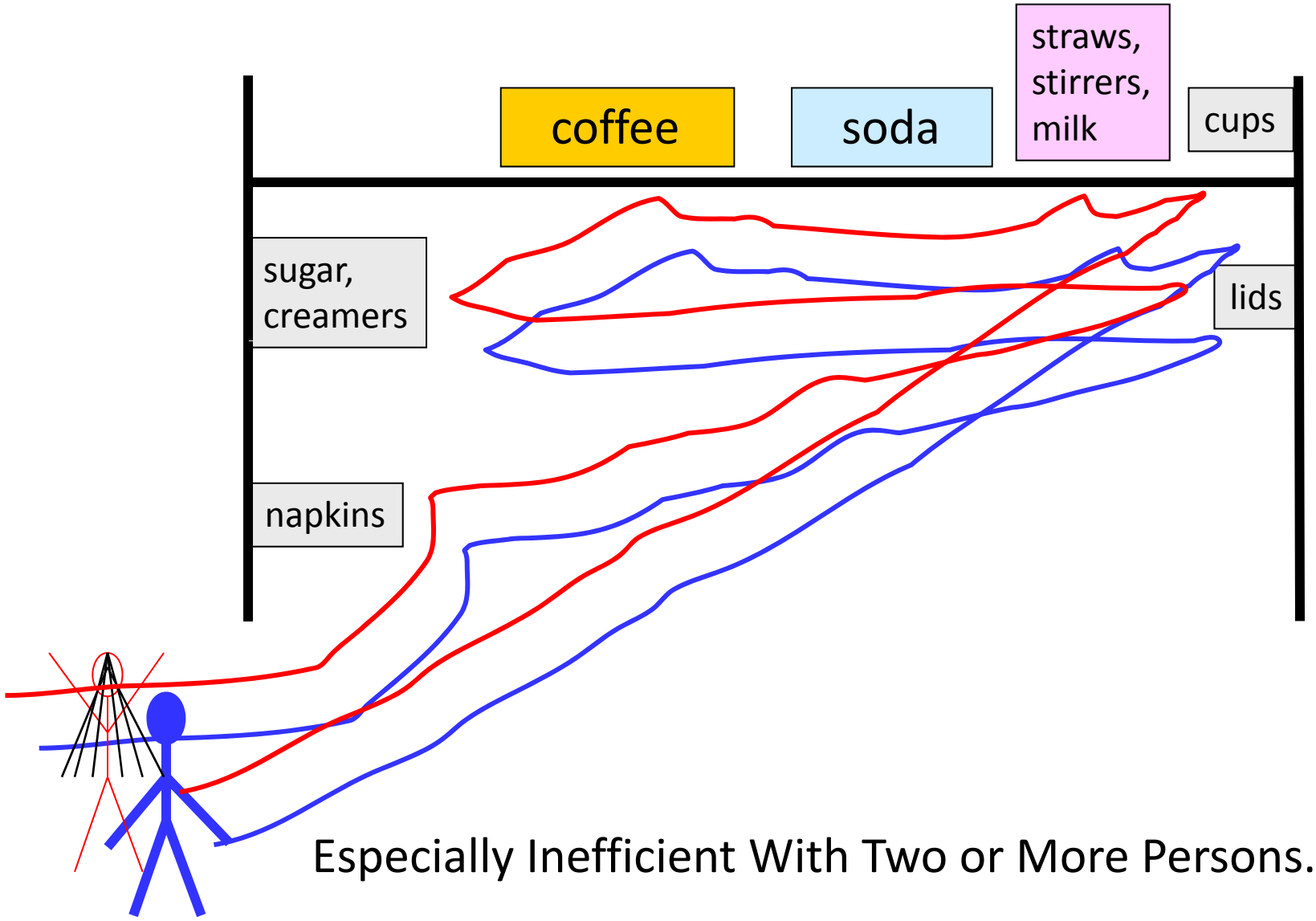




Getting Morning Coffee at the Cafeteria

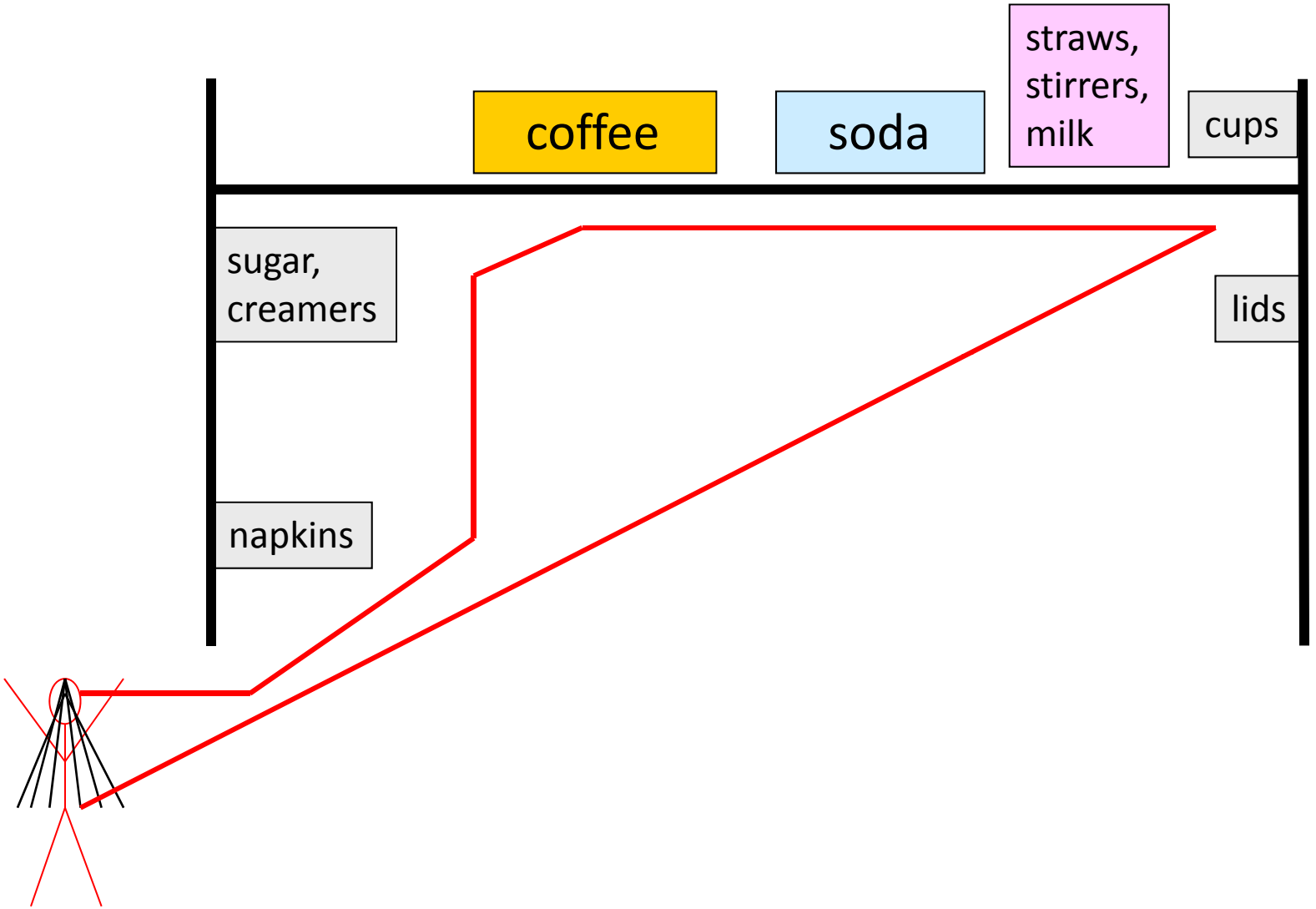


Getting Morning Coffee at the Cafeteria



Especially Inefficient With Two or More Persons…

Better: Think Computationally—Pipelining!



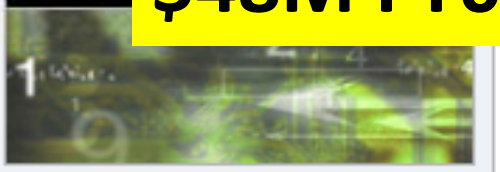
Computational Thinking at NSF

National Science Foundation
WHERE DISCOVERIES BEGIN

SEARCH
NSF Web Site

HOME | FUNDING | AWARDS | DISCOVERIES | NEWS | PUBLICATIONS | STATISTICS | ABOUT | FastLane

$48M FY08 to $100M in FY11 Budget Request



- Find Funding
- A-Z Index of Funding Opportunities
- Recent Funding Opportunities
- Upcoming Due Dates
- Advanced Funding Search
- Interdisciplinary
- How to Prepare Your Proposal
- About Funding

Proposals and Awards

- Proposal and Award Policies and Procedures Guide
  - Introduction
  - Proposal Preparation and Submission
    - Grant Proposal Guide
    - Grants.gov Application Guide
  - Award and Administration

Cyber-Enabled Discovery and Innovation (CDI)

Computational Thinking for Science and Engineering

CONTACTS

Name	Email	Phone	Room
Eduardo Misawa	cdi@nsf.gov	(703) 292-8080	
Thomas Russell	cdi@nsf.gov	(703) 292-8080	
Kenneth Whang	cdi@nsf.gov	(703) 292-8080	

Drs. Misawa, Russell, and Whang are being assisted by a multidisciplinary team of Program Officers drawn from throughout NSF. CDI team members include: Kile [illegible] William [illegible] (SBE/OAD), [illegible]S), Dan Lubin (OD/OPP), Manish Parashar (OD/OCI), David Rockcliffe (BIO/MCB), Nigel Sharp (MPS/AST), Carl Taylor (BIO/DBI), Rita Teutonico (SBE/OAD), Susan Winter (OD/OCI), William Wiseman (OD/OPP), and Eva Zanzerkia (GEO/EAR).

PROGRAM GUIDELINES

Solicitation 10-506

SYNOPSIS

Cyber-Enabled Discovery and Innovation (CDI) is NSF's bold five-year initiative to create *revolutionary* science and engineering research outcomes made possible by innovations and advances in computational thinking. Computational thinking is defined comprehensively to encompass computational concepts, methods, models, algorithms, and tools. Applied in challenging science and engineering research and education contexts, computational thinking promises a profound impact on the Nation's [illegible]

Range of Disciplines in CDI Awards

- Aerospace engineering
- Astrophysics and cosmology
- Atmospheric sciences
- Biochemistry
- Biomaterials
- Biophysics
- Chemical engineering
- Civil engineering
- Communications science and engineering
- Computer science
- Cosmology
- Ecosystems
- Genomics
- Geosciences
- Linguistics
- Materials engineering
- Mathematics
- Mechanical engineering
- Molecular biology
- Nanocomputing
- Neuroscience
- Proteomics
- Robotics
- Social sciences
- Statistics
- Statistical physics
- Sustainability
- …

… advances via Computational Thinking

National Science Foundation
WHERE DISCOVERIES BEGIN

SEARCH

HOME | FUNDING | AWARDS | DISCOVERIES | NEWS | PUBLICATIONS | STATISTICS | ABOUT | FastLane

Funding

Find Funding
A-Z Index of Funding Opportunities
Recent Funding Opportunities
Upcoming Due Dates
Advanced Funding Search
Inter
How
About Funding

Proposals and Awards

Proposal and Award Policies and Procedures Guide
Introduction
Proposal Preparation and Submission
• Grant Proposal Guide
• Grants.gov Application Guide

Email Print Share

Division of Computer and Network Systems

Computing Education for the 21st Century (CE21)

CONTACTS

Name	Email	Phone	Room
Janice Cuny	jcuny@nsf.gov	(703) 292-8489	1175
Jim Hamos	jhamos@nsf.gov	(703)-292-4687	835N
Joan Peckham	jpeckham@nsf.gov	(703) 292-8970	1160

PROGRAM GUIDELINES

"to develop competencies in computational thinking"

Full Proposal Target Date: February 22, 2011
Planning proposals ONLY.
Last Tuesday in February, Annually Thereafter

Full Proposal Deadline Date: April 27, 2011
Type I and Type II proposals ONLY
Last Wednesday in April, Annually Thereafter

Full Proposal Target Date: July 28, 2011
Planning proposals ONLY.
Last Thursday in July, Annually Thereafter

SYNOPSIS

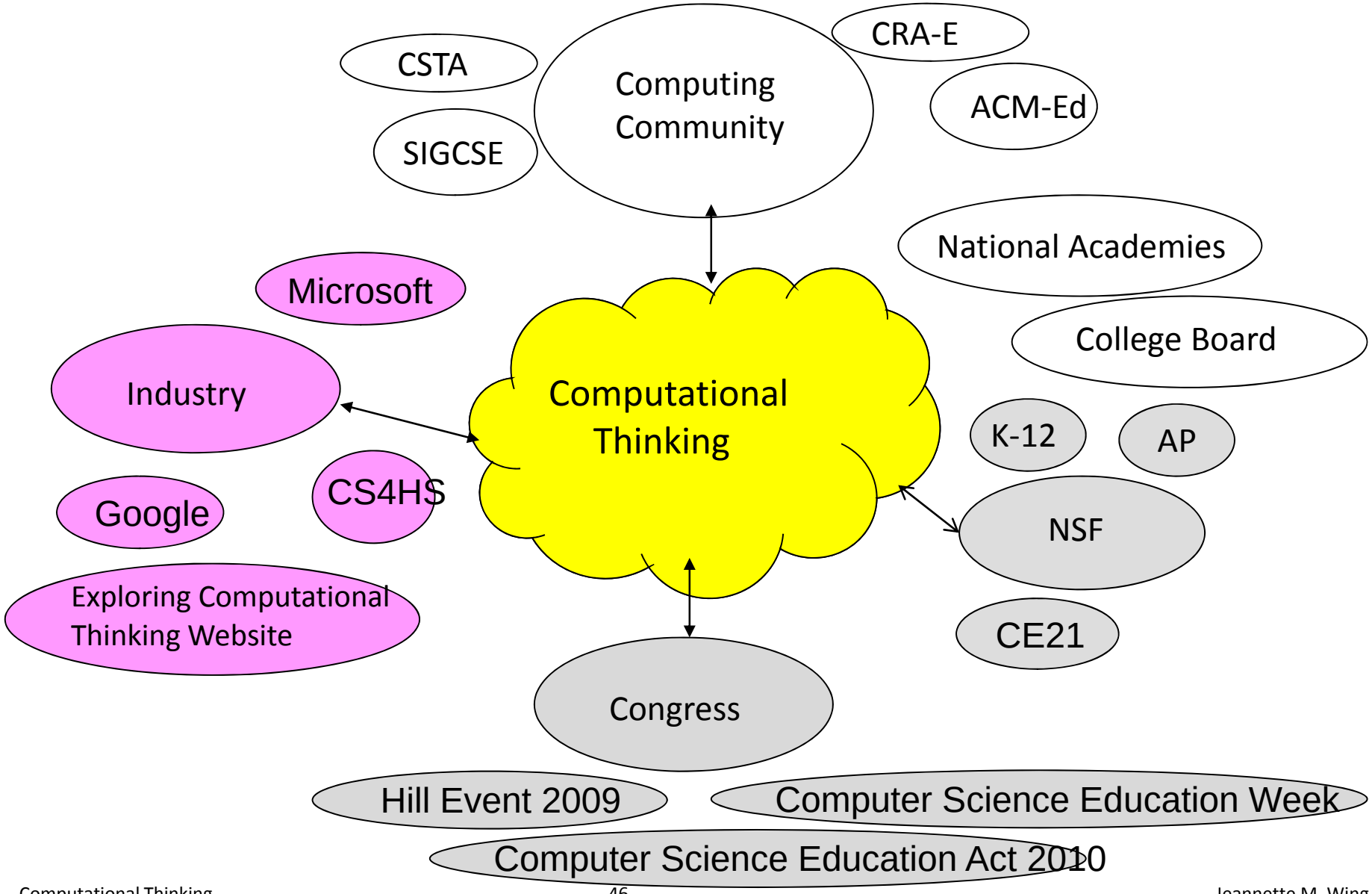
Computational Thinking in Education and Beyond

CMU and Other Colleges and Universities

- CMU: Redesign of Intro Courses
  - "15-110: Principles of Computer Science. An introduction to computer science, based on the principles of computational thinking. Many taking this course will be nonmajors, but we will also use it as the entry point for any entering student with limited programming experience." [Bryant, Stehlik, Sutner, Introductory Computer Science Education at Carnegie Mellon University: A Deans' Perspective, CMU-CS-10-140, August 2010]

- Examples: Brown, Bryn Mawr, Colorado State University, Columbia, Eastern Michigan University, Georgetown, Georgia Tech, Harvard, Haverford, Harvey Mudd, Kent State, MIT, New York City College of Technology, Northwestern, Princeton, Rochester Institute of Technology, St Joseph's U, U of Alabama-Birmingham, U of Florida, UNC-Charlotte, U of Puerto Rico, UTexas-Arlington, U of Wisconsin-La Crosse, Vanderbilt, Villanova, William & Mary,…

C.T. in Education: National Efforts

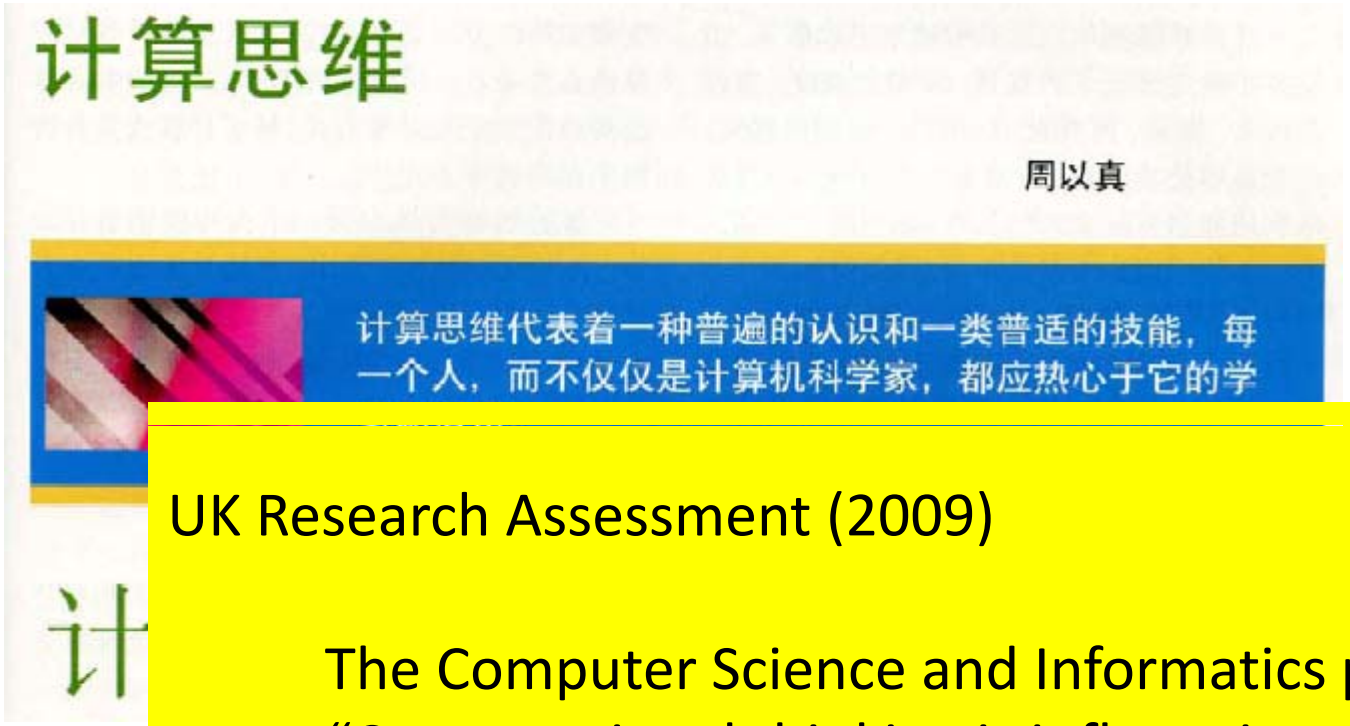


CMU and Other Colleges and Universities

- CMU: Redesign of Intro Courses
  - "15-110: Principles of Computer Science. An introduction to computer science, based on the principles of computational thinking. Many taking this course will be nonmajors, but we will also use it as the entry point for any entering student with limited programming experience." [Bryant, Stehlik, Sutner, Introductory Computer Science Education at Carnegie Mellon University: A Deans' Perspective, CMU-CS-10-140, August 2010]

- Examples: Brown, Bryn Mawr, Colorado State University, Columbia, Eastern Michigan University, Georgetown, Georgia Tech, Harvard, Haverford, Harvey Mudd, Kent State, MIT, Northwestern, New York City College of Technology, Princeton, Rochester Institute of Technology, St Joseph's U, Towson, U of Alabama-Birmingham, U of Florida, UNC-Charlotte, U of Puerto Rico, UTexas-Arlington, U of Wisconsin-La Crosse, Vanderbilt, Villanova, William & Mary,…

Computational Thinking, International



UK Research Assessment (2009)

The Computer Science and Informatics panel said
"Computational thinking is influencing all disciplines…."

par Jeannette M. Wing

Cet article fait suite aux divers interviews que nous avons faits et qui nous invitaient à une réflexion sur les fondements de notre discipline et ses aspects philosophiques et épistémologiques. Aujourd'hui l'article de Jeannette Wing nous conduit à réfléchir sur l'utilité et l'ubiquité de la pensée informatique et ses implications, mais aussi sur l'essence même de cette pensée.

Academic Organizations

- College Board http://csprinciples.org
  - With NSF support, revision of CS AP courses
  - Five universities pilots this year: UNC-Charlotte, UC Berkeley, Metropolitan Sate College of Denver, UC San Diego, and University of Washington
  - More schools–high schools, community colleges, and universities–to participate next year.

- National Academies Computer Science and Telecommunications Board
  - Report of a Workshop on the Scope and Nature of Computational Thinking, National Research Council, 2010 http://www8.nationalacademies.org/cp/projectview.aspx?key=48969

Computer Science Organizations

- SIGCSE sessions
- ACM Educational Council discussions
- CSTA: http://www.csta.acm.org/
  - *Computational Thinking Resource Set: A Problem-Solving Tool for Every Classrooom*
- CRA-E
  - "Creating Environments for Computational Researcher Education," August 9, 2010. http://www.cra.org/uploads/documents/resources/rissues/CRA-E-Researcher-Education.pdf
  - Includes recommendations for computational thinking courses for non-majors.

Congress

- Event on the Hill: May 29, 2009
  - Sponsored by ACM, CRA, CSTA, IEEE, Microsoft, NCWIT, NSF, and SWE , called for putting the "C" (computer science) into "STEM."
- Computer Science Education Week: http://www.csedweek.org/
  - Sponsored by ABI, ACM, BHEF, CRA, CSTA, Dot Diva, Google, Globaloria, Intel, Microsoft, NCWIT, NSF, SAS, and Upsilon Pi Epsilon.
- Computer Science Education Act
  - On July 30, 2010 Rep. Jared Polis (D-CO) introduced the Computer Science Education Act (H.R.5929) to strengthen K-12 computer science education.

Industry Support

- CS4HS
  - Initiated in 2006 at CMU, with support from Google and later Microsoft,
  - 2007, spread to UCLA and UW
  - By 2010, under the auspices of Google, CS4HS spread to 20 schools in the US and 14 in Europe, the Middle East, and Africa.

- Microsoft Research-Carnegie Mellon Center for Computational Thinking: http://www.cs.cmu.edu/~CompThink/
  - Since 2007, the Center supports both research and educational outreach projects.

- Google's Exploring Computational Thinking website, launched Oct 2010: http://www.google.com/edu/computational-thinking/index.html
  - Wealth of links to further web resources, including lesson plans for K-12 teachers in science and mathematics.

Spread the Word

- Help make computational thinking commonplace!

To fellow faculty, students, researchers, administrators, teachers, parents, principals, guidance counselors, school boards, teachers' unions,
congressmen, policy makers, …

Thank you!

References (Representative Only)

- Computational Thinking
  - University of Edinburgh, http://www.inf.ed.ac.uk/research/programmes/comp-think/
  - [Wing06] J.M. Wing, "Computational Thinking," *CACM* Viewpoint, March 2006, pp. 33-35, http://www.cs.cmu.edu/~wing/
- Model Checking, Temporal Logic, Binary Decisions Diagrams
  - [Br86] Randal Bryant, "Graph-Based Algorithms for Boolean Function Manipulation," *IEEE Trans. Computers*, 35(8): 677-691 (1986).
  - [CE81] E. M. Clarke and E. A. Emerson, "The Design and Synthesis of Synchronization Skeletons Using Temporal Logic," *Proceedings of the Workshop on Logics of Programs*, IBM Watson Research Center, Yorktown Heights, New York, Springer-Verlag Lecture Notes in Computer Science, #131, pp. 52–71, May 1981.
  - [CES86] E. M. Clarke, E. A. Emerson, and A. P. Sistla, "Automatic Verification of Finite State Concurrent Systems Using Temporal Logic Specifications," *ACM Trans. Prog. Lang. and Sys.*, (8)2, pp. 244-263, 1986.
  - [CGP99] Edmund M. Clarke, Jr., Orna Grumberg and Doron A. Peled, *Model Checking*, MIT Press, 1999, ISBN 0-262-03270-8.
  - [Ku94] Robert P. Kurshan, *Computer Aided Verification of Coordinating Processes: An Automata-theoretic Approach*, Princeton Univ. Press, 1994.
  - [Pn77] Amir Pnueli, "The Temporal Logic of Programs," *Foundations of Computer Science*, FOCS, pp. 46-57, 1977.
  - [QS82] Jean-Pierre Queille, Joseph Sifakis, "Specification and verification of concurrent systems in CESAR," *Symposium on Programming*, Springer LNCS #137 1982: 337-351.
  - [VW86] Moshe Y. Vardi and Pierre Wolper, "An Automata-Theoretic Approach to Automatic Program Verification (Preliminary Report)," *Logic in Computer Science*, LICS 1986: 332-344.
- Computational Thinking and Biology
  - Allessina and Pascual, "Googling Food Webs: Can an Eigenvector Measure Species' Importance for Coextinctions?", PLoS Computational Biology, 5(9), September 4, 2009. http://www.ploscompbiol.org/article/info:doi%2F10.1371%2Fjournal.pcbi.1000494
  - Executable Cell Biology, Jasmin Fisher and Thomas A Henzinger, *Nature Biotechnology*, Vol. 25, No. 11, November 2007. (See paper for many other excellent references.)
  - [LJ07] Predicting Protein Folding Kinetics via Temporal Logic Model Checking, Christopher Langmead and Sumit Jha, WABI, 2007.
  - Systems Biology Group, Ziv Bar-Joseph, Carnegie Mellon University, http://www.sb.cs.cmu.edu/pages/publications.html

References (Representative Only)

- Machine Learning and Applications
  - Christopher Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
  - [FS99] Yoav Freund and Robert E. Schapire, "A short introduction to boosting." *Journal of Japanese Society for Artificial Intelligence*, 14(5):771-780, September, 1999.
  - Tom Mitchell, *Machine Learning*, McGraw Hill, 1997
  - Symbolic Aggregate Approximation, Eamonn Keogh, UC Riverside, http://www.cs.ucr.edu/~eamonn/SAX.htm (applications in Medical, Meteorological and many other domains)
  - The Auton Lab, Artur Dubrawski, Jeff Schneider, Andrew Moore, Carnegie Mellon, http://www.autonlab.org/autonweb/2.html (applications in Astronomy, Finance, Forensics, Medical and many other domains)
- Computational Thinking and Astronomy
  - J. Gray, A.S. Szalay, A. Thakar, P. Kunszt, C. Stoughton, D. Slutz, J. vandenBerg, "Data Mining the SDSS SkyServer Database," in Distributed Data & Structures 4: Records of the 4th International Meeting, W. Litwin, G. Levy (eds), Paris France March 2002, Carleton Scientific 2003, ISBN 1-894145-13-5, pp 189-210.
  - Sloan Digital Sky Survey @Johns Hopkins University, http://www.sdss.jhu.edu/
- Computational Thinking and Archaeology
  - Columbia University: http://www.cs.columbia.edu/~atroccol/
  - Digital Forma Urbis Romae Project: http://www.graphics.stanford.edu/projects/forma-urbis/
  - eHeritage Project, MSRA: http://research.microsoft.com/en-us/collaboration/global/asia-pacific/programs/eheritage.aspx
  - See also Marc Levoy's digital archaeology projects: http://www-graphics.stanford.edu/~levoy/
  - See also UK universities: http://en.wikipedia.org/wiki/Computational_archaeology#Research_groups_and_institutions
- Computational Thinking and Chemistry
  - [Ma07] Paul Madden, Computation and Computational Thinking in Chemistry, February 28, 2007 talk off http://www.inf.ed.ac.uk/research/programmes/comp-think/previous.html
- Computational Thinking and Economics
  - Abraham, D., Blum, A. and Sandholm, T., "Clearing algorithms for barter exchange markets: enabling nationwide kidney exchanges," *Proc. 8th ACM Conf. on Electronic Commerce*, pp. 295–304. New York, NY: Association for Computing Machinery, 2007.
  - Conitzer, V., Sandholm, T., and Lang, J., When Are Elections with Few Candidates Hard to Manipulate? *Journal of the ACM*, 54(3), June 2007.

References (Representative Only)

- Conitzer, V. and Sandholm, T., Universal Voting Protocol Tweaks to Make Manipulation Hard. In Proceedings of the *International Joint Conference on Artificial Intelligence (IJCAI), 2003.*
- Michael Kearns, Computational Game Theory, Economics, and Multi-Agent Systems, University of Pennsylvania, http://www.cis.upenn.edu/~mkearns/#gamepapers
- Algorithmic Game Theory, edited by Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani, September 2007, http://www.cambridge.org/us/catalogue/catalogue.asp?isbn=9780521872829
- David Pennock, Yahoo! Research, Algorithmic Economics, http://research.yahoo.com/ksc/Algorithmic_Economics

- Computational Thinking and Journalism
  - MacArthur Foundation Digital Media and Learning Initiative: http://www.macfound.org/site/c.lkLXJ8MQKrH/b.946881/k.B85/Domestic_Grantmaking__Digital_Media__Learning.htm
  - Kim Pearson, Poynter Online, eMedia Tidbits, May 2009: http://www.poynter.org/column.asp?id=31&aid=164084
  - Georgia Tech: http://www.computation-and-journalism.com/main/
- Computational Thinking and Law
  - The Poirot Project, http://www.ffpoirot.org/
  - Robert Plotkin, Esq., *The Genie in the Machine: How Computer-Automated Inventing is Revolutionizing Law and Business*, forthcoming from Stanford University Press, April 2009, Available from www.geniemachine.com
  - Burkhard Schafer, Computational Legal Theory, http://www.law.ed.ac.uk/staff/burkhardschafer_69.aspx
  - Stanford Computational Law, http://complaw.stanford.edu/
- Computational Thinking and Medicine and Healthcare
  - The Diamond Project, Intel Research Pittsburgh, http://techresearch.intel.com/articles/Tera-Scale/1496.htm
  - Institute for Computational Medicine, Johns Hopkins University, http://www.icm.jhu.edu/
  - See also Symbolic Aggregate Approximation, Eamonn Keogh, UC Riverside, http://www.cs.ucr.edu/~eamonn/SAX.htm
  - SM Belknap, H Moore, SA Lanzotti, PR Yarnold, M Getz, DL Deitrick, A Peterson, J Akeson, T Maurer , RC Soltysik, GA Storm, and I Brooks, Application of Software Design Principles and Debugging Methods to an Analgesia Prescription Reduces Risk of Severe Injury From Medical Use of Opioids, *Clinical pharmacology & Therapeutics,* Vol 84 No. 3 , September 2008, pp. 385-392.
- Computational Thinking and Meteorology
  - Yubin Yang, Hui Lin, Zhongyang Guo, Jixi Jiang, "A data mining approach for heavy rainfall forecasting based on satellite image sequence analysisSource," *Computers and Geosciences,* Volume 33 , Issue 1, January 2007, pp. 20-30, ISSN:0098-3004.
  - See also Symbolic Aggregate Approximation, Eamonn Keogh, UC Riverside, http://www.cs.ucr.edu/~eamonn/SAX.htm

References (Representative Only)

- Computational Thinking (especially Machine Learning) and Neuroscience
  - Yong Fan, Dinggang Shen, Davatzikos, C., "**Detecting Cognitive States from fMRI Images by Machine Learning and Multivariate Classification,"** Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06, June 2006, p. 89.
  - T.M. Mitchell, R. Hutchinson, R.S. Niculescu, F.Pereira, X. Wang, M. Just, and S. Newman, "Learning to Decode Cognitive States from Brain Images,"*Machine Learning*, Vol. 57, Issue 1-2, pp. 145-175. October 2004.
  - X. Wang, R. Hutchinson, and T. M. Mitchell, "Training fMRI Classifiers to Detect Cognitive States across Multiple Human Subjects ," *Neural Information Processing Systems 2003*. December 2003.
  - T. Mitchell, R. Hutchinson, M. Just, R.S. Niculescu, F. Pereira, X. Wang, "Classifying Instantaneous Cognitive States from fMRI Data," *American Medical Informatics Association Symposium*, October 2003.
  - Dmitri Samaras, Image Analysis Lab, http://www.cs.sunysb.edu/~ial/brain.html
  - Singh, Vishwajeet and Miyapuram, K. P. and Bapi, Raju S., "*Detection of Cognitive States from fMRI data using Machine Learning Techniques," IJCAI, 2007.*

- Computational Thinking and Sports
  - Synergy Sports analyzes NBA videos, http://broadcastengineering.com/news/video-data-dissect-basketball-0608/
  - Lance Armstrong's cycling computer tracks man and machine statistics, website

Credits

- Copyrighted material used under Fair Use. If you are the copyright holder and believe your material has been used unfairly, or if you have any suggestions, feedback, or support, please contact: jsoleil@nsf.gov

- Except where otherwise indicated, permission is granted to copy, distribute, and/or modify all images in this document under the terms of the GNU Free Documentation license, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation license" (http://commons.wikimedia.org/wiki/Commons:GNU_Free_Documentation_License)

Computational Thinking, in Summary

- Computational thinking is the *thought processes* involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information processing agent. [Cuny, Snyder, Wing 2010, in progress]