Computer Science in the Classroom: Challenges and Opportunities

Chris Stephenson
Executive Director
This material is based upon work supported by the National Science Foundation under Grant No. 0455403. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.
Agenda

• Computer science as a discipline
• What the research says
• Issues that affect us
• Key principles
• The need for change
• CSTA resources
Computer Science as a Discipline

• Definition difficulties:
  – Computer science is a new area within a very old educational cannon
  – Computer science as a field is evolving so quickly that it is difficult even for computer
    scientists to define its contents and delimit its boundaries
  – Deciding what students need to know in this field is like trying to hit a moving target

• What we know for sure:
  – Computer science is a scientific discipline and not just a technology that supports
    learning in other areas
  – There are recurring concepts and principles that students need to know (abstraction,
    complexity, modularity, and reusability)
  – Computer science is a discipline with a core set of scientific principles that can be
    applied to solve complex real-world problems and promote higher-order thinking

• Computer science is the study of computers and algorithmic processes, including
  their principles, their hardware and software design, their applications, and their
  impact on society. (ACM Model Curriculum pg. 13)
What the Research Tells Us

- Only 26% of schools require students to take a computer science course
- Only 40% of schools even offer AP CS
- Lack of time in the students’ schedules is the greatest impediment to students taking computer science courses
- 89% of high school computer science teachers say they experience a sense of isolation and a lack of collegial support in their schools and in their districts
- Noting the rapid changes in technology and in teaching, teachers indicated that their greatest professional development need was actually finding time for their own on-going learning
Why We’re in Trouble

- Computer science teachers feel isolated within their schools and districts
- Administrators do not understand that computing is a scientific discipline just like physics and biology
- There is no consistency in computing curriculum requirements from state-to-state, district-to-district, or even in some cases, school-to-school
- There is no consistency in teacher certification requirements
- Computing teachers do not receive the professional development they need to keep their teaching and technical skills current
- There is a disconnect between K-12 CS educators and their college/university colleagues
- Administrators, legislators, and congressional committees do not understand the link between supporting K-12 computer science education and economic/workplace issues (the national discourse example)
Why Enrollment Is Dropping

• Students and their parents do not understand the incredible scope of educational and career opportunities that computing provides

• Students want to be part of a discipline that is solving real problems, and they do not understand that computing is at the root of all of the new sciences

• Students’ schedules are so jammed packed that they do not have time to take elective courses

• The emphasis on standardized testing in core areas is pulling emphasis, funding, and good teachers away from computer science
The Truth About AP

- The College Board determines the content of the AP exam based upon surveys of universities and colleges
- When demand, applications, and enrollments were high, the APCS exam served as a convenient gate keeper
- While other AP exams represent exemplary high school courses, the AP CS exam is equivalent to a second year data structures course
- While other disciplines are able to scaffold knowledge and skills with multiple courses throughout K-12, the AP CS course is often the only CS course students can take
The Impact of NCLB

• This year, the No Child Left Behind legislation was implemented in high schools

• Under this legislation, federal funding is withdrawn from schools where students fail to reach specified performance levels on standardized tests in math and reading

• The result:
  – Non-core courses are being cancelled
  – Funds are being withdrawn from other programs
  – CS teachers are being pulled out of their classrooms to teach remedial mathematics (the Los Angeles example)
The Teacher Certification Mess

• CSTA research indicates that:
  – Certification requirements vary enormously from state to state
  – Many states require CS teachers to hold multiple certifications with CS as a secondary to some other discipline
  – Some states require CS teachers to take and pass praxis exams in other disciplines (math, business, vocational technology)
  – Teachers are ill-informed as to the requirements in their own state
  – Many DOE people responsible for certification are ill-informed as to the requirements in their own state (primarily because they do not know what computer science is)
  – In some states where there are clearly-stated requirements, there is no way for them to be met (the Florida example)
How Do We Change Things?

• Change is a long-term process, not a short term intervention

• Curriculum must have both bottom up and top-down support

• Successful curriculum implementation requires buy-in at every level (federal policy makers, state policy makers, school district administrators, principals, teachers, colleges and universities, schools of education, business and industry)

• Major change agents must be in place at all levels

• Adequate resources must be made available throughout all of the stages of curriculum design and implementation (funding for resources, hardware and software, learning materials, professional development)
What If We Don’t Change?

• Our students will fail to receive the education they require to compete in an increasingly technological global economy

• Businesses and industries will continue to be unable to find the people with the skills they need for the jobs that exist now and in future

• Our economic power as a nation depends upon our ability to build the tools the rest of the world needs and while computer science education languishes in the U.S., other countries are driving toward the future by providing computer science education for their students

• Computer science is driving innovation in every single field of science and we will continue to lose the innovation edge
Seven Systemic Changes to Improve Teaching

• **Mastery of Knowledge:** New high school computer science teachers should be required to have completed an undergraduate degree in computer science or a comparable degree program

• **Standardized Pre-service Programs:** All teacher preparation programs should be required to adhere to the National Council for Accreditation of Teacher Education (NCATE) standards for high school computer science educators

• **Certification standards:** State teacher certification requirements for high school computer science teachers should adhere to a consistent (and enforced) national standard that would allow for greater clarity and mobility from state to state.

• **Professional Development:** School districts should provide regular professional development for computer science teachers to allow them to keep their knowledge and skills current

• **Focus on Teaching:** School districts should employ a sufficient number of technical specialists with responsibility to ensure that computer hardware, networks, and software is maintained, freeing teachers to concentrate on their teaching

• **Competitive Compensation:** Salaries for computer science teachers should be commensurate with those offered in industry to ensure that the best possible candidates prepare and apply for teaching positions

• **Professional Affiliation:** All high school computer science teachers should be members of professional associations that support their discipline-based knowledge and provide a teaching community that mentors and celebrates them
Ten Principles of Curriculum Design

• Focus on underlying scientific principles
• Develop student familiarity with abstraction, complexity, modularity, reusability
• Focus on problem solving and critical thinking
• Help students develop a range of capabilities/skills independent of technologies
• Give a broad overview of the field (history, computing in other disciplines)
• Deal explicitly with design, maintenance, and analysis
• Enable students to scaffold new ideas, concepts, and skills across a series of courses with age-appropriate outcomes
• Use teaching strategies that make the content engaging to all students
• Interweave conceptual and experimental issues
• Don’t confuse computer science with computer literacy
Five Implementation Requirements

• **Support:**
  – The initiative must have top-down and grassroots support and agents must be in place at all levels to ensure continued enthusiasm and support

• **Stakeholder buy-in:**
  – External groups must have a role in the review process (teachers unions, professional associations, parent councils, universities/colleges, business/industry)

• **Resources:**
  – Schools, teachers, and students must be provided with the resources they need for successful implementation (hardware, software, textbooks, reference materials, manipulatives)

• **Professional development:**
  – Teachers must receive training to allow them to master the curriculum content and effective teaching strategies

• **Timeframe:**
  – Every step takes time and real system change takes up to ten years. Giving less time than truly needed to accomplish any step along the implementation path from vision to reality can condemn the entire process to failure
Five Qualities of Exemplary Teachers

• **Problem-solving Approach:** Exemplary computer science teachers use a problem-solving approach that allows students to examine problems from different angles and perspectives and formulate solutions.

• **Real World Focus:** Exemplary computer science teachers motivate students by having them create real-world artifacts with an intended audience and encouraging them to understand the essential link between the problem, the user, and the solution.

• **Explicit Emphasis on Design:** Exemplary computer science teachers explicitly teach and use the software design process, ensuring that students master the steps involved in designing, creating, testing, and debugging software.

• **A Welcoming Environment:** Exemplary computer science teachers make their classroom a welcoming environment for all students (especially young women and minority students) and find creative ways to engage all students with examples and exercises that are relevant to their lives.

• **Modeling Life-long Learning:** Exemplary computer science teachers serve as role models for their students by continuing to enhance their own teaching and technology skills and by exploring new ideas and new technologies.
CSTA’s Goals and Objectives

Creating a community of individuals and organizations working together to address critical issues in K-12 computer science education.

- **Promote a Better Understanding of Computer Science**: Provide visibility, influence policy, and generate resources that illuminate computer science as an essential academic discipline.
- **Develop Research and Resources**: Conduct original research and serve as a direct-to-practitioner channel for the dissemination of research and resources that addresses current knowledge gaps.
- **Support National Standards**: Facilitate the implementation of national curriculum and teacher certification standards to support consistent excellence in learning and teaching.
- **Support Teacher Excellence**: Provide multiple levels of professional development to improve teachers’ technical knowledge and pedagogical skills.
- **Opportunities**: Promote computer science as a field of study and as a career destination that provides a wealth of opportunities to students regardless of their gender, race, or socio-economic status.
CSTA Curriculum Solutions

- *The ACM Model Curriculum for K-12 Computer Science*
  
  [http://csta.acm.org/Curriculum/sub/ACMK12CSModel.html](http://csta.acm.org/Curriculum/sub/ACMK12CSModel.html)

- Online resource materials to support the ACM Model Curriculum: The Outlines and Objectives Documents
  
  [http://csta.acm.org/Curriculum/sub/ACMK12CSModel.html](http://csta.acm.org/Curriculum/sub/ACMK12CSModel.html)

- *The New Educational Imperative: Improving High School Computer Science Education*: a comprehensive white paper bringing together U.S. and international research to provide practical solutions for achieving long-term systemic improvement
  
  [http://csta.acm.org/Publications/Publications.html#dbottom](http://csta.acm.org/Publications/Publications.html#dbottom)
Teacher Preparation Solutions

• JETT: Java Engagement for Teacher Training workshops offered in partnership with colleges and universities across the country (53 workshops for 650 teachers held to date)
  
  http://jett.acm.org/

• TECS: Teacher Engagement for Computer Science introductory CS workshops offered in partnership with colleges and universities across the country (11 workshops for 200 teachers held to date)
  
  http://tecs.acm.org/

• The annual Computer Science and Information Technology Symposium (professional development for over 700 teachers across the country). 8th CSIT Symposium: June 28th in Atlanta
  
Resources & Information

• The Teacher Certification database: a state-by-state list of computer science teacher certification requirements and contacts (now under construction)

• The CSTA web repository: A national repository of resources and learning materials (now under construction)
  http://csta.acm.org/Resources/sub/WebRepository.html

• National research initiatives providing cutting edge data on the state of K-12 computer science education
Resource & Information cont.

- Careers in Computing Poster (for high school and middle school classrooms): in partnership with ACM-W and ASCA
  [http://csta.acm.org/Careers/Careers.html](http://csta.acm.org/Careers/Careers.html)

- Careers in Computing Lesson Plan
  [http://csta.acm.org/Careers/Careers.html](http://csta.acm.org/Careers/Careers.html)

- The ACM Computing Degrees and Careers brochure and website
  [http://computingcareers.acm.org](http://computingcareers.acm.org)
Resource & Information cont.

- The CSTA *Voice*: a quarterly newsletter focusing on key issues and resources for computer science educators
  
  http://csta.acm.org/Publications/Publications.html#ptop

- CSTA *Advocate*: a blog for discussion of key organizational issues and programs
  
  http://blog.acm.org/csta/

- CSTA Information brochure for policy-makers
  
  http://csta.acm.org/About/sub/AdvocacyOutreach.html
Contact Information

Chris Stephenson
Executive Director, CSTA
Phone: 1-800-401-1799
Fax: 1-541-687-1840
cstephenson@csta.acm.org