Feature

A New K–12 Computer Science Curriculum

By Allen B. Tucker

Subject: Computer science

Grade Level: K–12 (Ages 5–18)

Technology: Programming languages

Audience: Administrators, technology coordinators, teachers

Standards: *NETS•S* 1 (http://www.iste.org/standards/)

Although there are numerous challenges for implementing a comprehensive K–12 computer science curriculum in the United States, the need is obvious, and the time to begin is now.

Computer science is a well-established academic discipline at the collegiate and postgraduate levels, as well as a distinct and important profession in the world economy. Unfortunately, the integration of computer science concepts into the K–12 curriculum has not kept pace in the United States. As a result, the public is not as well educated about computer science as it should be, and the current shortage of information technology workers at all levels threatens to continue in the future.

An effort sponsored by the Association for Computing Machinery (ACM) has produced a report, *A Model Curriculum for K–12 Computer Science,* that aims to help address this problem. Find the report at the ACM Web site, along with a detailed discussion of the issues and the model itself. (***Editor's Note:*** Find this and other URLs in the Resources section on p. 20.)

The report provides a framework by which school districts can begin to educate young people in computer science, thus better preparing them for effective citizenship in the 21st century.

Why It's Needed

Most teachers who now offer computer science in K–12 schools are experiencing a strong sense of isolation and vulnerability. This frustration has many roots, including the glacially slow pace of attitudinal and programmatic change, the battle to obtain adequate computing resources, the lack of acceptance of computer science among math and science colleagues, the absence of state curricular standards, the shortage of opportunities for inservice and preservice training in computer science, and the unusual vulnerability of computer science faculty and courses to budget cuts during times of fiscal restraint.

There is an urgent need to improve the level of public understanding of computer science as an academic and professional field, especially its distinctions from management information systems (MIS), information technology (IT), mathematics, and the natural sciences. Elementary and secondary schools have a unique opportunity and responsibility to address this need. Not only will a broad commitment to K–12 computer science education create such public understanding, but it will also help address the worldwide shortage of computer specialists. The creation of a viable model for a computer science curriculum and its implementation in K–12 is a necessary first step toward reaching these goals.

This model assumes that K–12 computer science curricula should enable students to learn:

- programming
- hardware design
- networks
- graphics
- databases and information retrieval
- computer security
- software design
- programming languages
- logic
- translation between levels of abstraction
- artificial intelligence
- the limits of computation
- applications in information technology and information systems
- social issues, such as Internet security, privacy, and intellectual property

Although the emerging IT curricula involves the proper use of technologies by which people manipulate and share information in its various forms, an IT curriculum is not nearly adequate as a substitute for computer science.

The development of K–12 computer science is making better progress internationally. In Israel, for example, a secondary school computer science curriculum was approved by the Ministry of Higher Education and implemented in 1998. In Canada, a comprehensive curriculum was implemented for all secondary schools in Ontario in 2001. In many other parts of the world, including Europe, Russia, Asia, South Africa, New Zealand, and Australia, computer science is being established and integrated into the K–12 curriculum. Thus, there should be a growing sense of urgency about K–12 computer science in the United States.

The Proposed Model

The report identifies a four-level framework for computer science, containing roughly the equivalent of four half-year courses (many of these can be taught as modules, integrated within existing science and mathematics courses). The first two levels contain subject matter that can be mastered by all students. The second two levels suggest topics that can be elected by students with special interest in computer science, whether they are college bound or not.

Level I (recommended for Grades K–8) provides elementary school students with foundational concepts in computer science by integrating basic skills in technology with simple ideas about algorithmic thinking. This can be best accomplished by adding short modules to existing science, mathematics, and social studies units. A combination of NETS and an introduction to algorithmic thinking (as offered, for instance, by Logo or other hands-on experiences) would ensure that students meet this goal.

Most teachers who now offer computer science in K–12 schools are experiencing a strong sense of isolation and vulnerability.

In addition to school- and districtwide workshops, state and regional events can be organized to bring teachers together as a community to learn and exchange ideas.

Level II students (Grade 9 or 10) should acquire a coherent and broad understanding of the principles, methodologies, and applications of computer science in the modern world. This can best be offered as a one-year course accessible to all students. For many students, this course will be their last encounter with computer science. Therefore, it should be considered essential preparation for the modern world.

Level III is for students (Grade 10 or 11) who elect to study more computer science in a class that would earn a math or science curriculum credit. This course continues the study begun at Level II, but it places particular emphasis on the scientific and engineering aspects of computer science—mathematical principles, algorithmic problem solving and programming, software and hardware design, networks, and social impact. Students will elect this course to explore their interest and aptitude for computer science as a profession.

Level IV (Grade 11 or 12) is an elective that provides depth of study in one particular area of computer science. This may be an AP computer science course that offers depth of study in programming and data structures. Alternatively, it could be a project-based course in multimedia design or a vendor-supplied course that leads to professional certification. Any Level IV course will naturally require the Level II course as a prerequisite, and some will require the Level III course as well.

These four levels have the following general goals:

- Students should understand the nature of computer science and its place in the modern world.
- Students should understand that computer science interleaves principles and skills.
- Students should be able to use computer science skills (especially algorithmic thinking) in their problem-solving activities in other subjects.
- The computer science curriculum should complement IT and AP computer science curricula in schools where they are already offered.

The appendix to the report provides "proof of concept" by describing activities that are currently being taught in different school districts at each of the four levels.

Challenges

The recommendations in the report are not made in a vacuum. It acknowledges the serious budgetary constraints that school districts are now operating under and the uphill battle that computer science faces in the light of other academic and nonacademic priorities. Thus, the recommendations are intended to initiate a long-term evolution of computer science in K–12 schools rather than suggesting a quick fix. Efforts will be needed in three principal areas to sustain any momentum that the report may have created: teacher preparation, state-level content standards, and curriculum materials development.

Teacher Preparation

For students to learn the fundamentals of computer science, teachers must acquire both a mastery of the subject matter and the pedagogical skills that will enable them to present the material to students at appropriate levels. Thus, state departments of education and other appropriate agencies must recognize the discipline of computer science, so that appropriate standards for teacher certification can be established. This can then be followed by the establishment of teacher preparation programs with a prescribed course of study in computer science education, so that perspective teachers will gain the skills and knowledge necessary to meet certification standards.

The need for teacher preparation standards has recently been addressed by the National Council for Accreditation of Teacher Education (NCATE), a coalition of 33 specialty professional associations of teachers, teacher educators, content specialists, and local and state policy makers. NCATE oversees the professional accreditation of schools, colleges, and departments of education.

NCATE has recently defined accreditation standards for secondary school computer science teacher education programs. It is anticipated that these standards can be implemented through a teacher preparation endorsement program, roughly equivalent to providing prospective teachers with a minor in computer science (including at least 18 semester hours of college-level computer science coursework). The prerequisite for this program is a foundation in educational technology.

As states develop teacher certification requirements and content standards for K–12 computer science education, schools will be prompted to implement relevant computer science curricula. Most importantly, schools of education will be motivated to introduce preservice programs in computer science education. With computer science becoming a recognized discipline, schools of education will need to set up such preservice programs.

For the minority of states that offer teachers an endorsement in computer science, requirements vary widely—

some require teachers to have a background in data processing, while others require them to have a business background. Thus, state departments of education should review their certification standards for professional educators so that they identify computer science as a distinct discipline. As the requirements for certification and preservice programs are developed, they must also maintain the view that computer science is a rapidly evolving discipline.

As with other subjects, inservice education is important to help current teachers adapt and integrate new computer science curriculum elements. Inservice education at the early stages of this curriculum implementation can take many forms. In addition to school- and districtwide workshops, state and regional events can be organized to bring teachers together as a community to learn and exchange ideas. These events can be used to disseminate to the teachers and school administrators new curricular recommendations and guidelines as they evolve. Another important event would be to include short workshops addressing timely issues in computing during the implementation of the new curriculum.

Professional recognition is important for the current cohort of teachers of computer science. These teachers typically have original credentials in mathematics, science, business, or English, but they've self-educated to teach many different types of computing courses, including AP computer science. One way to provide such recognition is for states to develop standard core competencies for computer science teachers and endorse those teachers who can demonstrate these competencies. Development of core competencies can be accomplished through new inservice training initiatives.

State-Level Content Standards

Recently, efforts have increased to develop national and state content standards for computer science. Curriculum standards serve to define the skills and knowledge of the discipline to be acquired by every student. For this to happen, school curricula must be aligned with these standards. Content standards for computer science education need to be developed and adopted in a way that parallels what routinely happens in disciplines such as science, mathematics, and language arts. Curriculum frameworks aligned with these content standards can then be developed for the classroom.

In the design of state standards, it is important to ensure distinction between the teaching of IT skills (especially in service to the sciences and mathematics) and the teaching of computer science itself. Computer science must be considered as subject matter, and IT should be viewed as a tool that cuts across all subject areas, including computer science. Existing IT standards, thus, are not substitutable for computer science standards.

Curriculum Development

The report presents a model for computer science education, but not a complete curriculum. Additional steps need to be taken to formulate content standards, define professional development needs, develop curriculum (textbooks and laboratory materials), and disseminate to students in the classroom. For all this to happen, teachers must play a substantial and leading role in the formulation of curriculum components. This will also require the participation of university faculty and professional organizations to serve as facilitators and support a process that will yield a deliverable and effective curriculum.

One vehicle for mobilizing these efforts would be to seek grant support from federal agencies and private foundations. Ideally, a summer institute in K–12 computer science education could be established and teachers would be chosen to participate in the development of curriculum content and teaching modules. The institute would have working groups for the four levels identified in the report, and could be held at multiple locations throughout the country for two or three weeks each summer. Participants would come together the following summer to discuss results and plan follow-up activities; participant travel costs and stipends would be covered at the outset by grants from agencies such as the NSF.

Sustainability

People in leadership positions need to acknowledge the importance of computer science education for the future of our society. Although a handful of states have begun to establish computer science content standards, define models for teacher certification, provide inservice training in computer science, and experiment with developing new curricular materials, a much more comprehensive level of commitment is now needed. Professional organizations in computer science—the ACM, the IEEE Computer Society, ISTE, and other national and local teacher organizations—can help facilitate progress toward this goal.

The report proposes a model, but not teaching materials, lesson plans, a trained teaching cohort, or an operational budget to deliver K–12 computer science among schools. The following additional steps are needed to begin this process of implementation.

For students to learn the fundamentals of computer science, teachers must acquire both a mastery of the subject matter and the pedagogical skills that will enable them to present the material to students at appropriate levels.

Content standards for computer science education need to be developed and adopted in a way that parallels what routinely happens in disciplines such as science, mathematics, and language arts.

Buy-In. The report should be endorsed by all organizations that have a stake in its recommendations: ACM SIGCSE, ISTE's Special Interest Group for Computer Science (SIGCS), directors of curriculum in school districts (ASCD), state boards of education, the National Education Association (NEA), the National Association of Secondary School Principals (NASSP), and the National School Board Association.

Curriculum and Course Development. Funding sources should be approached to assist teams of K–12 teachers and other computer science educators to develop pilot courses along the lines suggested in this report. Concurrently, textbook and Web-based publishers should be encouraged to invest in these experimental courses so that the resulting teaching materials can be widely disseminated and used elsewhere.

Professional Societies. The establishment of a national computer science teachers association—a new professional society for K–12 computer science teachers—has recently been proposed by ACM. Similarly, ACM SIGCSE and ISTE should continue to broaden their missions and conferences to better accommodate the professional interests of K–12 computer science teachers. State and regional organizations should provide support and collaboration for K–12 computer science teachers at the local level.

Culture. If computer science is to become a meaningful and effective academic culture within the mainstream of K–12 education, big changes must take place and priorities must be set that value the computer science curriculum. Until then, teachers will continue to struggle to keep any sort of computer science presence in their school's curriculum, and schools that do not now offer computer science will probably never consider making such a change.

Conclusion

Additional steps will still be needed to sustain this work beyond curriculum development and dissemination. For example, new certification standards and programs usually pass through a complex and sometimes bureaucratic administrative process before they are adopted. But collaboration among professional organizations in education and computing, colleges and universities, state education departments, and teachers can help facilitate progress. Consequently, a coordinating entity that supports and sustains the long-term interests of K–12 computer science education must emerge.

Computer science is a mainstream discipline that can no longer be ignored by public schools. The report provides a model that states, schools of education, and individual school districts can use to begin implementing a coherent computer science curriculum available to all students. Much work needs to be done to translate this model into teaching and laboratory materials that are pedagogically viable and widely accessible. Corporations, foundations, and other external sources should support this effort by providing incentives that will enable such a curriculum development effort to succeed.

Resources

A Model Curriculum for K–12 Computer Science—Report of the ACM K–12 Task Force Curriculum Committee: http://www.acm.org/education/k12

AP Course Description—Computer Science: http://www.collegeboard.com/ap/students/compsci/

Computer Science Unplugged: http://www.unplugged.canterbury.ac.nz

Gal-Ezer, J. & Harel, D. (1999). Curriculum and course syllabi for a high-school CS program. *Computer Science Education 9*(2), 114–147.

NETS•T: http://www.iste.org/standards/

State of Maine Learning Results: http://www.state.me.us/education/lres/lres.htm

National Research Council, Committee on Information Technology Literacy, *Being Fluent with Information Technology:* http://www.nap.edu/catalog/6482.html

Program for Initial Preparation of Teachers of Educational Computing and Technological Literacy, and Secondary Computer Science Education: http://www.ncate.org/standard/programstds.htm

Mindstorms: Children, Computers, and Powerful Ideas, and more information: http://el.media.mit.edu/logo-foundation/products/books.html#learn

2000–2001 Taulbee Survey: http://www.cra.org/CRN/articles/march02/bryant.vardi.html

Allen Tucker is the Anne and Robert Bass Professor and Chair of the Computer Science Department at Bowdoin College. He earned a BA from Wesleyan University and an MS and PhD in computer science from Northwestern University. For his many contributions to curriculum design, he received the ACM Outstanding Contribution Award in 1991 and the ACM SIGCSE Outstanding Contribution to CS Education Award in 2001.