June 14, 2010

Ms. Mary Maxon, Ph.D.
Deputy Executive Director
President’s Council of Advisors on Science and Technology (PCAST)
Office of Science and Technology Policy
Executive Office of the President
New Executive Office Building, Room 5224-5
Washington, DC 20502

Dear Ms. Maxon:

We are following up on the questions that several members of PCAST asked in response to remarks that ACM presented on May 21 regarding computer science education.

Computing is a driving force in our culture and our economy creating entire new industries and enabling innovation in existing ones. The discipline of computer science underpins the computing technology behind these transformative changes. Following the introduction of computer science into many high schools in the mid-1980’s, there has been -- over the past fifteen years -- a significant dilution of what is called “computer science” at the high school level. The reasons behind this are complex, with layers of federal, state and local policy playing varying roles. Two central reasons are:

- The deep confusion about what constitutes computer science education versus computer literacy/fluency, and
- The fact that computer science courses often are not considered part of or even eligible to be part of the core courses students must take to graduate from high school.

It is striking that the decline in computer science courses¹ is occurring even with a national movement to increase the capacity and quality of K-12 science, technology, engineering and mathematics (STEM) education in the United States. It is also occurring when computing-related job prospects are among the brightest for all STEM fields².

¹ Through surveys of K-12 schools, the Computer Science Teachers Association has found that introductory secondary school computer science courses have decreased in number by 17 percent from 2005 and the number of Advanced Placement Computer Science Courses has decreased by 33 percent.

² For breakdown of job openings vs. degrees granted in several STEM fields see, http://www.acm.org/public-policy/08-18%20chart.jpg
We offer the following recommendations for PCAST to consider as part of its STEM education report:

- Clearly and explicitly define K-12 computer science education as part of STEM education programs
- Support state planning and implementation grants to improve computer science education
- Appoint a blue ribbon commission to review the computer science teacher certification crisis
- Create pre-service and professional development opportunities for computer science teachers
- Enlist NITRD to coordinate programs that could seek to improve K-12 computer science education

We have attached a detailed justification of these recommendations and detailed answers to the questions raised by PCAST members.

Thank you again for the opportunity to comment on these issues. Please feel free to contact us with further questions.

Sincerely,

Robert Schnabel
Chair, ACM Education Policy Committee
Dean, School of Informatics and Computing
Indiana University

Cameron Wilson
Director
Office of Public Policy
Association for Computing Machinery
Recommendations

- Clearly and explicitly define K-12 computer science education as part of STEM education programs

Educators and policy makers consistently confuse the use of technology and teaching of technology literacy with teaching computer science as a core academic discipline within the STEM fields. In fact, this confusion is a fundamental reason behind many of the policy issues we have identified. ACM and the Computer Science Teachers Association (CSTA) offer the following definition of computer science as a discipline and elements that can be considered appropriate for computer science curricula:

**Computer science** means the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society; and, **Computer science education** includes the following elements: design (both software and hardware), creation of digital artifacts, abstraction, logic, algorithm development and implementation, programming paradigms and languages, theoretical foundations, networks, graphics, databases and information retrieval, information security and privacy, artificial intelligence, the relationship between computing and mathematics, the limits of computation, applications in information technology and information systems, and social impacts of computing.

We have found that while policy makers and educators have recognized STEM education is critical for the 21st Century workplace, K-12 computer science is often marginalized within the STEM reform discussions. For example, federal K-12 STEM education programs, such as those established by the Elementary and Secondary Education Act and America COMPETEs, generally focus resources on the “core” curriculum implemented in schools. These biases are usually subtle ranging from requiring teacher certification (which is highly problematic for computer science teachers as often they are certified “out of field,” meaning they can be certified in “business” even while teaching computer science courses) to requirements that improvements from baseline assessments in “science” and in “mathematics” be measured (such as in the National Science Foundation’s Math and Science program). Federal education programs -- particularly those in STEM education -- need to recognize explicitly that K-12 computer science proposals are eligible.

- Support state planning and implementation grants to improve computer science education

We need to incentivize states to develop specific plans to improve computer science education. Upcoming research by ACM and CSTA reveals:
• Very few states have defined K-12 learning standards for computer science education, and those that do exist are focused exclusively on developing computing skills instead of teaching computer science concepts;
• Few states allow computer science courses to “count” in the high school graduation requirement rubric;
• Computer science teacher certification programs are largely broken; and
• There are significant gender and equity issues regarding access to computer science education that must be addressed.

The federal government needs to catalyze action in the states through targeted grants to address the lack of computer science standards, assessments, courses counting toward graduation and teacher preparation, professional development and certification.

• **Appoint a blue ribbon commission to review the computer science teacher certification crisis**

Certification programs and requirements for computer science teachers must be reformed or built from the ground up. Often they do not exist, and when they do they are typically not connected to actual computer science content knowledge. We recommend the Secretary of Education establish a blue ribbon commission of state officials, discipline experts, practitioner organizations and practitioners to review state computer science teacher certification requirements and share best practices.

• **Create pre-service and professional development opportunities for computer science teachers**

Despite the existence of NCATE accreditation requirements for computer science, very few pre-service teacher preparation programs have the current capacity or coursework developed to prepare computer science teachers. For example, methodology courses in computer science needed to prepare teachers are almost non-existent. Federal programs should specifically seek to address this issue by targeting grants to develop new partnerships between schools of education and computer science departments to develop coursework leading to the K-12 certification of computer science teachers.

• **Enlist NITRD to coordinate programs that could seek to improve K-12 computer science education**

The National Information Technology Research and Development (NITRD) program cuts across numerous federal agencies with a mission to carry out and coordinate investments in IT R&D. While it has a research focus, part of its mission is supporting education. NITRD should address K-12 computer science education and diversity issues as part of its strategic planning and road-mapping process. We also note that the Department of Education is no longer a member of the NITRD program. We recommend that the Administration move to reengage the Department of
Education in the NITRD program to use its resources and influence in addressing computer science education issues.

Responses to PCAST Member Questions

What is the relationship between computer science education and “computing” education? How does computer science curriculum fit into making students career or college ready?

These two questions were asked independently but are deeply intertwined, so we have combined the response.

Despite the incredible diversity of the United States workforce, it is clear that most of today’s jobs depend on some knowledge of, and skills to use, computing technologies. It is also clear that this trend is growing as computing becomes embedded more deeply in everyday commerce and society. What is unclear to many educators is what curriculum will support this growing trend.

Various studies and groups advocate for differing curriculum and standards that further cloud these waters. Some have broad definitions of “technology literacy” that encompass many fields (National Assessment’s Governing Board’s Technology and Engineering Literacy Framework for the 2014 National Assessment of Educational Progress), some link technology and engineering education (National Academies Engineering in K–12 Education: Understanding the Status and Improving the Prospects), and some focus on computing or information technology certification (Cisco Networking Academy), some focus on ensuring technology is used throughout the curriculum (International Society for Technology Education’s NETS’ standards). And we argue for specific computer science standards and courses that are academically rigorous, widely engaging, founded upon the body of knowledge for the discipline, and that are part of the educational core for all students.

This swirl of terms is daunting to anyone trying to understand the integration of computing and technology in STEM fields and the work place. This confusion is one of the fundamental reasons why computer science standards, course classification and credits, teacher certification programs, and curriculum are inadequate or non-existent. We hope that the following definitions will provide some assistance for educators, policy makers, and parents who are trying to cope with this confusion and complexity.

Computer science -- An academic discipline that means the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society. In the sections below we detail the ACM/CSTA standards that underpin the teaching of computer science. These standards serve as the baseline for measurement against existing state standards in this study.
Technology literacy and fluency -- A spectrum of curricula ranging from literacy (understanding how to use technology) to fluency (the ability to express ideas creatively, reformulate knowledge, and synthesize new information and technology).

Information technology courses -- A broad and diverse set of topics, but typically focused on teaching to apply the components of information technology, such as network or database administration.

Educational technology or computing across the curriculum -- This is the integration of technology into teaching in order to advance student learning across academic disciplines.

Computing education -- Another broadly used term that, depending on the educational context may encompass only one of the noted areas above to all of them and more.

The reality is all of these efforts are worthwhile and have a place in K–12 education, but we must be clear about how they are related and how they are distinct to ensure adequate workforce and college preparation.

Goals to embed technology use in education or to ensure technology literacy are often focused on ensuring a basic knowledge of IT or a specialized knowledge (in the case of IT courses) of a narrow tool. These are clearly important skills considering computing’s everyday uses in today’s economy.

Our research has shown that most states are focused on lower-level skills instead of deeper computer science concepts and capabilities. Further, only 10 states now allow computer science courses, if they exist, to count as a core mathematics or science requirement. But states currently require students to take some form of technology literacy or computing application courses. As our economy becomes more dependent on computing technologies, it is clear that simply having a facial knowledge of computing is not enough.

Where computer science education differs from basic technology literacy/IT goals is that it teaches fundamental concepts of computing, just as an academic course in physics will teach a student the fundamental laws of motion and energy. Computer science teaching should sit on a continuum from basic computing concepts that can be attained at elementary and middle school levels to deeper knowledge, skills, and practices more appropriate for secondary school. Some of its topics overlap with technology literacy and IT curriculum, while some are completely different. For example, the complexity of algorithms is a fundamental idea in computer science but probably would not appear in a technology literacy or IT curriculum. Gaining a deeper knowledge of computer science and its fundamental aspects is essential not only to have a clear understanding of “what is going on under the hood” of software or hardware, but also to develop critical thinking skills that will serve a student throughout his or her career.
The future workforce picture (see chart) provides an even clearer case why schools need to move toward improving computer science education. The United States Bureau of Labor Statistics projects that the computing sector will have 1.5 million job openings over the next ten years making this one of the fastest growing economic fields. While there are many pathways into these jobs, a deeper look at the fastest growing occupations within this field (such as computer software engineers) shows they either will require a computer science or related degree or greatly benefit from the knowledge and skills imparted by computer science courses. Improving the outlook for students going into computing careers is only a small reason for addressing K–12 computer science education issues. The knowledge and skills imparted by computer science enables innovation and opens doors. Many fields of science and business depend on computer science. Modeling brings together many of the fundamental concepts of computer science and is used on a daily basis in many fields. New subfields of science, such as computational physics or computational biology, have been created by the fusing of computer science with an existing scientific discipline. If K–12 schools are seeking to make students college and career ready, computer science must be at the core of their curriculum.

What federal policies are barriers to improving computer science education?

Because of accountability provisions in No Child Left Behind and the focus of states on that Act’s “core” disciplines in developing high school graduation requirements, investments in curriculum, pedagogy and professional development are focused on “core” courses. We also have seen a pronounced shift toward the presumed state adoption of the work of the “Common Core Standards Initiative” and its “college and career ready standards” in the Race to the Top competitive grant guidance and in the President’s proposed Fiscal Year 2011 budget. In practice, this means schools, states and federal programs emphasize mathematics, reading and natural sciences. Therefore, well-meaning federal legislation intended to improve STEM education broadly often does not include computer science at the state and local levels, since it is not typically considered part of this “core.”

This same issue plays out in programs authorized by the COMPETEs Act and the Elementary and Secondary Education Act, putting K-12 computer science education in a classic “Catch-22.” Because computer science is not part of the core it does not have the same level of assessments or teacher support as core programs that education policy makers seek to improve course offerings, but it is difficult to develop these without being in the core.

For example, NSF’s Math and Science Partnership has five types of awards, including Targeted Partnerships intended for “a specific disciplinary focus in mathematics or the sciences.” At a high level, the program seems broadly STEM focused. In fact, the COMPETEs Act amended what was then current law to clarify the scope of the program to include all of the STEM disciplines. However, guidance to grant applicants asks specifically for baseline data on how the proposals will improve student achievement in math and/or science standards. This focus—intended or not—is discouraging to applicants hoping to address computer science and creates a barrier for our field. Similarly, the significant public investments in math and science
assessments rarely address computer science. Therefore, computer science proposals have difficulty meeting the baseline data requirements. This puts computer science proposals at a distinct disadvantage relative to math and science proposals, and deters would-be applicants.

This same confusion stymies teacher certification processes and complicates the status of “highly qualified” computer science teachers, which has implications for multiple COMPETEs Act programs. For example, the MSP award category Teacher Institutes for the 21st Century was created by the COMPETEs Act. The scope of the program is to serve STEM teachers who “are considered highly qualified.” The COMPETEs Act references the underlying definition of “highly qualified” in the Elementary and Secondary Education Act (ESEA). One of the requirements for teachers to be highly qualified is certification and demonstrated knowledge in the subject area in which they teach. (Other programs in COMPETEs that rely on the highly qualified criteria include Teachers for a Competitive Tomorrow and the Robert Noyce Teacher Scholarship Program.) These same issues appear as barriers for schools distributing Title II funding under the Elementary and Secondary Education Act.

Do we have national standards for what constitutes computer science education?

ACM and CSTA have published A Model Curriculum for K–12 Computer Science: Final Report of the ACM K–12 Task Force Curriculum Committee to define a series of standards that were appropriate for students at the elementary and secondary level. We have attached a short document that details these standards by grade level.

---

**Recommended Grade Level**

<table>
<thead>
<tr>
<th>K–8</th>
<th><strong>Level I</strong>—Foundations of Computer Science</th>
</tr>
</thead>
<tbody>
<tr>
<td>9 or 10</td>
<td><strong>Level II</strong>—Computer Science In the Modern World</td>
</tr>
<tr>
<td>10 or 11</td>
<td><strong>Level III</strong>—Computer Science as Analysis and Design</td>
</tr>
<tr>
<td>11 or 12</td>
<td><strong>Level IV</strong>—Topics in Computer Science</td>
</tr>
</tbody>
</table>

Figure 1. Structure of a K–12 Computer Science Curriculum

---

3 [http://www.csta.acm.org/Curriculum/sub/ACMK12CSModel.html](http://www.csta.acm.org/Curriculum/sub/ACMK12CSModel.html)
The standards are intended to give educators a four-part framework around which they can design innovative, rigorous, and engaging computer science curriculum.

The first part is a course of study that teaches basic computer or technology literacy skills in elementary and middle schools (K-8). It is followed by a progression of courses (parts 2 and 3, with the option of a fourth) that go deeper into the breadth and rigor of the field for high school students (9-12).

The high school model curriculum can be taught in a two- or three-course framework. The first course is accessible to all students, whether they are college-bound or intend to enter the workforce immediately after high school. Its goal is to provide all students an introduction to the principles of computer science and its place in the modern world. Concepts such as basic computing components (software), algorithmic problem-solving, computer networks, connections between elements of mathematics and computer science (including binary numbers, logic, sets, and functions), and exposure to careers in computing are part of this introductory course.

The second and/or third course(s) increase in rigor, giving students a deeper understanding of algorithm development, problem solving, programming while using software engineering principles, interface design, the limits of computers, and societal and ethical issues of software engineering.

**What international standards exist for computer science education?**

We have not done extensive research on the topic of international standards for K-12 computer science instruction. However, a 2005 study by CSTA found the following:

“Looking at the experiences of other countries that have designed and implemented a high school computer science curriculum can provide valuable lessons as to the factors that must be in place to ensure that our efforts to improve computer science education are successful. An examination of the experiences of an international panel of computer educators from Canada, Israel, Scotland, South Africa, and the United States support the argument that such curriculum initiatives will be successful only to the extent to which they meet the following criteria:

- There is a link between the outcome required and the strategies used.
- Change is driven by real learning needs and not politically manufactured needs.
- Educational change must be seen in the context of larger social and economic forces.

---

All of the stakeholders must agree to the need for change and on the strategies put in place to achieve it.

Change requires the commitment of adequate resources through all phases of the design, implementation, and testing of the new curriculum.

Change is a long-term process, not a short-term intervention.”

We have attached two pages that describe the efforts in these countries.

**What data do we have on why students are or are not interested in computer science courses?**

While enrollments for computer science majors have started to slowly increase\(^5\), the computing community is deeply concerned about data showing that incoming freshman are not interested in majoring in computer science and about the widening gender gap for computer science majors. (see chart below)

<table>
<thead>
<tr>
<th>Freshman Intent to Major in Computer Science</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>Female</td>
</tr>
<tr>
<td>Male</td>
</tr>
</tbody>
</table>

Source: UCLA, Higher Education Research Institute Survey of Incoming Freshman

There are several prevailing theories: students never get exposed to exciting computer science in K-12, the myth that off shoring of jobs leaves little opportunity, or that the false image that computer science majors sit in a cubical all day long programming, etc. Much of the data we have is focused on the image aspects and, in particular, the major gender and ethnicity diversity challenges for computer science.

As part of the *New Image in Computing* project, ACM has partnered with WGBH Educational Foundation to research\(^6\) the attitudes held by high school students toward the study of computer science and test messages about the discipline. The first study confirmed a significant gender gap between girls and boys in their opinions of computing as a possible college major or career. The research found that 74 percent of boys—regardless of race or ethnicity—reported that a college major in computer science was a “very good” or “good” choice for them, but only 10 percent of girls rated it as a “very good” choice and 22 percent rated it as “good.”

The gender gap extended to computer science as a potential career choice as well as a field of study. From a selection of 15 possible careers, computer science came in fourth among the respondents, with 46 percent rating it “very good” or “good.” However, while 67 percent of all


\(^{6}\) [http://www.acm.org/membership/NIC.pdf](http://www.acm.org/membership/NIC.pdf)
boys rated computer science highly as a career choice, only nine percent of girls rated it “very
good” and 17 percent rated it “good.”

All survey respondents were asked, “What word comes to mind when you see or hear the word
computing?” The most common words (“software,” “programming,” “technology,” etc.) were
similar between boys and girls. But gender differences emerged in the secondary words. For
example, boys tended to use words such as “design,” “games,” “video,” etc., with more
frequency than girls. By contrast, the secondary words used by the girls tended to take on a
more negative tone—with “boring,” “hard,” and “nerd” being used more frequently.

In an unexpected finding, the research showed little racial/ethnic differentiation in young
people’s attitudes toward computer science. In fact, computer science was held in high regard
by college-bound African American and Hispanic boys, but these two groups remain
underrepresented in both academia and the computer science workforce. The report concluded
that the image issue might not apply in these cases.

ACM and WGBH are now seeking additional funding from NSF to further explore this finding.