

## Transforming High School Computer Science: CS / 10,000 Project

If we are to build a globally competitive 21<sup>st</sup> century workforce and maintain our leadership in IT innovation, there is no stage in the academic pipeline more crucial than high school. It is true that students begin to lose interest in computing much earlier, probably in grades 4-5. Yet engagement programs for middle school students will not be effective if those students have no further opportunities during their four years of high school. Likewise, new and reinvigorated college computing programs cannot have a significant impact if there are too few interested and qualified students to show up at their doors. There are clear indications that college programs are already impacted. Since 2000, the percentage of incoming college freshman who intend to major in computing has decreased more than 70%; for women, the figure is closer to 80%. While some universities believe this trend may be leveling off or even turning around, the HERI data – a survey of incoming college freshman which has been extremely accurate in predicting degree attainment after four years – declined still further in 2008, with just 1% of students intending to major in computing.

High schools are key, yet they teach computing inadequately. Historically, computing classes taught keyboarding, and were consigned to Vocational Education. Surprisingly many schools continue that designation, which often, though not always, results in instruction in a very basic computing literacy course on a track rarely taken by college-bound students. A student in LA described the computing course at her high school as, “It’s like they make them type ... the students actually just type stuff and then they copy it and copy, paste, copy, paste.”<sup>1</sup> With the exception of the Advanced Placement (AP) CS course, few high schools offer any computing courses with college preparatory status.

The AP CS is a rigorous college preparatory course; however, nationally only 2000 computing teachers have passed the AP audit, indicating that it is being taught in less than 10% of our high schools. Even that course is not optimal: it is programming-centric; it is inaccessible to students with no prior experience; it does not focus on the fundamental concepts of computer science or computational thinking; and it does little to teach the breadth of application or “magic” of computing. Consequently, the AP CS A test was taken by only 14,529 students in 2008,<sup>2</sup> as compared to the 204,564 who took the Calculus AB exam, the 141,321 who took the Biology exam, or the 96,282 who took the Statistics exam. AP CS A also had the worst gender balance of any of the AP tests. Just 18.3% of the CS AP test takers were women in comparison to the Calculus AB test, where 48.7% were women, or Statistics, where 50.2% were women. Only 11.8% of the AP CS A test takers were underrepresented minorities.

Clearly, high school computing needs to change. We are serving too few of our students well. We propose

**CS/10,000 Project Goal:** *To develop an effective new high school curriculum for computing, taught in 10,000 high schools by 10,000 well-qualified teachers by 2015.*

---

<sup>1</sup> Private communication 2009, Jane Margolis, Senior Researcher, UCLA School of Graduate Education and Information Studies.

<sup>2</sup> The AP CS AB test, with more advanced material, was cancelled this year by the College Board because too few students were taking it; the breakeven point for an AP exam is 20,000 students.

The difficulty is not in developing such a curriculum but ensuring its adoption. K-12 education in the U.S. is decentralized, often making it necessary to proceed with change district-by-district, school-by-school. The only single point of national leverage that we have is the College Board AP courses, which are taught in more than 17,000 schools nationwide.<sup>3</sup> AP courses are attractive to schools – even under-resourced schools – that are trying to raise their academic standards. AP courses are also appealing to college-bound students because in most cases they carry college preparatory credit, they often have additional weight in computing grade point averages, and they are valued by college admissions officers. The existing AP CS course, however, is not what the CS/IT field needs.

Computing is not the only AP course that needs rethinking. In 2002, the National Research Council report, *Math and Science in U.S. High Schools*, recommended that AP courses be redesigned to “reflect what we know about how students learn, build students’ transferable, conceptual understanding and inquiry skills, and convey the content and unifying concepts of a discipline.” In addition, it cautioned that AP courses should not be designed solely to replicate introductory college courses, which “are not typically exemplary models.” In response, Biology, Physics, Chemistry, and Environmental Sciences initiated a joint project in 2005 to redesign their AP courses, funded by the National Science Foundation (NSF). The result has been an emerging set of frameworks – hierarchies of Big Ideas, Critical Concepts, and Enduring Understandings – that define the content of the new courses and form the basis for further curriculum development in each discipline. In 2008, the College Board began a similar process to develop an entirely new CS AP course. It will not be a derivative of first year college CS courses, but it will be the gold standard of high school computing courses.

### **Gold Standard CS AP Course**

The new AP course will not be programming-centric but will instead focus on the fundamental concepts of computation. It will be engaging, accessible, inspiring, and rigorous. It will be a course that is highly relevant to all students, regardless of their major, but especially crucial for students planning to major in STEM disciplines. We expect that this new course will be a target for K-8 curriculum development, as well as an impetus for college curriculum reform. It is intended that the course carry college credit, if not placement, at many schools.

The course is being developed by an AP Commission, headed by Owen Atraschan of Duke University. All of the commissioners have had extensive AP experience and are enthusiastic about this project. In addition, the Commission has the active involvement of an illustrious Advisory Board:

---

<sup>3</sup> This counts schools that have at least one AP offering, not necessarily a CS offering. The International Baccalaureate (IB) Program is taught in fewer schools (approximately 1,000) than AP in the U.S. and tends not to be taught in under-resourced schools because of its cost. The CS / 10,000 Initiative will include representatives from the IB in all of its development efforts, however, in the hopes of influencing their curriculum redesign as well.

Duane Bailey (Williams College)	Mark Guzdial (GA Tech)	Jim Kurose (UMass)
Gail Chapman (CSTA)	Susanne Hambrusch (Purdue)	Richard Pattis (UC Irvine)
Tom Cortina (CMU)	Michelle Hutton (High school, TX)	Eric Roberts (Stanford)
Wanda Dann (Ithaca)	Rich Kick (High school, CA)	Larry Snyder (UW)
Stephen Edwards (VA Tech)	Deepak Kumar (Bryn Mawr)	Cameron Wilson (ACM)
Juan Gilbert (Auburn)		

Together the Commission and the Advisory Board have designed the first two levels of the CS framework. The first level has seven Big Ideas; the second level (shown in the accompanying document) has Critical Concepts for each of those big ideas.

### **Big Ideas in Computing**

1. Computing is a creative activity that draws on a wide variety of fields, such as the natural sciences, mathematics, engineering, social sciences, business, and the arts.
2. Abstraction is a central problem-solving technique in computer science.
3. Algorithms are the essence of computational problem solving.
4. Writing programs is an integral part of solving computational problems.
5. Theoretical and practical limitations affect what can be solved computationally.
6. Computing enables and empowers innovation, exploration, and the creation of knowledge.
7. Computing drives and is driven by economics, culture, society, and ethics.

### **A High School Curriculum**

The new AP course is only one part of proposed high school curriculum. The entire curriculum will be a three-course sequence:

1. **Introductory Course (Pre-AP):** This introductory course is intended for students who have minimal experience with computing. It will be designed for all students, even those who are not planning on STEM careers. It will go beyond computer literacy to fluency in the fundamentals of computing and computational thinking. It will use an inquiry-based instructional approach and engage students with exciting, 21<sup>st</sup> century applications. Possible models for this course already exist and a meeting has been planned for Spring 2009 to establish a common framework for them.
2. **New AP, Gold Standard Course**
3. **Existing (possibly modified) AP CS A Course:** The current Java-based programming course will be maintained for stakeholders who choose not to relinquish the existing AP course and for those students who want to go beyond the Gold Standard course and have an intensive programming experience, the current Java-based programming course will be maintained. That course mirrors

the first course taught in many CS departments and serves a purpose in providing placement credit for more advanced students. Once it has the first two courses behind it, however, it could potentially become a much more interesting, project-based course. (Potential modifications to that course are not part of this proposed effort.)

With NSF funding, there are already several projects in place to build the curriculum and materials for the first two courses, which we expect to have widely available by the time the new AP test comes online in 2015. Curriculum alone, however, will not be sufficient. To complete the project, we will need to prepare teachers and establish the course sequence in high schools.

### **10,000 Teachers in 10,000 Schools**

We are aggressively planning for 10,000 teachers in 10,000 schools by 2015. That is an ambitious goal and will need to be pursued on many fronts: in-service teacher preparation, preservice teacher preparation, ongoing professional development, and entrée into the school districts. In each case, we will need local partnerships with universities, community colleges, government agencies, nonprofits, and industry.

- **In-service teacher preparation.** Many of the needed 10,000 teachers can be drawn from the ranks of existing teachers but they will need extensive in-service preparation. There are 36,000 teachers<sup>4</sup> teaching computing in high schools in the U.S., but the quality and content of the courses they teach varies widely. Only 2,000 of them have passed the AP audit for CS AP and even among those teachers, there are very few who majored in computing-related fields. Current high school computing teachers have little background in the topics of computability, complexity, and the range of applications of IT that are part of the new course. To reach 10,000 teachers, we will need to develop a Professional Development (PD) curriculum aligned with the new course. It will include materials – both for on-line and face-to-face delivery – organized as modules that could be customized to the needs of specific school districts. We will also need to build an infrastructure of trained or master teachers to deliver these PD courses in each state. Teachers and trainers will have to be compensated for their time.
- **Preservice teacher preparation.** We will need to cultivate new teachers both to reach the goal of 10,000 and to sustain and increase that number over time. This will require working with college students from both math and the computing-related fields, and it will require working with both traditional and alternative certification programs. We will need collaborations between Colleges of Education and Departments of Computing to develop computing methods courses, to recruit CS students, and to partner with their local high schools. Some of the recruiting efforts should focus on colleges and universities with high numbers of minority students. We should work closely both with innovative programs such as UTEACH<sup>5</sup> and the Teacher Residency<sup>6</sup> programs, and with

---

<sup>4</sup> CSTA Teacher Survey, 2008.

<sup>5</sup> UTEACH: <http://uteach.utexas.edu/>

<sup>6</sup> Teacher Residency Programs: Creating and Sustaining Urban Teacher Residencies: A New Way to Recruit, Prepare, and Retain Effective Teachers in High-Needs Districts. Aspen Institute, 2008. <http://www.aspeninstitute.org/policy-work/education-society>

alternative certification organizations like Math for America,<sup>7</sup> Teach for America,<sup>8</sup> Transitions to Teaching (Troops to Teachers),<sup>9</sup> and Teaching Fellows.<sup>10</sup>

- **Ongoing Professional Development.** We will need to provide all of the teachers with the ongoing professional development that will allow them to keep current with the material. This professional development structure could perhaps be modeled on the National Writing Project (which provides materials, online support, professional development workshops, and online social networking).<sup>11</sup> Support should also include coaching, accessibility of master teachers, and the assistance of GK-12 students,<sup>12</sup> IT professionals, and retirees. It might also be possible to build a "Computing Student Corps" modeled on the STARS Student Leadership Corps<sup>13</sup> that could send teams of undergraduates into the schools to assist with in-school and out-of-school labs. We will need to expand the Computer Science Teachers Association (CSTA),<sup>14</sup> work with professional development programs such as CS4HS,<sup>15</sup> and collaborate with Mathematics Teacher's associations.
- **Entrée into schools.** Introduction of these courses is impeded not only by the lack of prepared teachers and curriculum but also by a lack of time in crowded schedules, the positioning of computing as an "elective" which often does not carry college preparatory credit, the absence of uniform standards and certification, the lack of appropriate facilities, the problem of getting adequate technical support for equipment, and the difficulties in keeping labs open during out of school hours. Some of these issues can be addressed locally, others by engaging professional organizations at the state level. We will also include efforts to engage students, partnering with informal education opportunities where appropriate, and including "AP Incentives" for teachers and students, perhaps by partnering with the NMSI (National Math and Science Initiative)<sup>16</sup> to include "AP Incentives" for teachers and students. In approaching teenage students, we will also make use of the marketing strategy that is being developed by the New Image of Computing effort by WGBH of Boston and the ACM.<sup>17</sup>

---

<sup>7</sup> MFA: <http://www.mathforamerica.org/home>

<sup>8</sup> TFA: <http://www.teachforamerica.org/>

<sup>9</sup> Transitions to Teaching: <http://www.ed.gov/programs/transitionteach/index.html>

<sup>10</sup> See, for example, <http://www.dcteachingfellows.org/overview.html> or <http://www.chicagoteachingfellows.org/>

<sup>11</sup> National Writing Project: <http://www.nwp.org/>

<sup>12</sup> GK-12: [http://www.nsf.gov/funding/pgm\\_summ.jsp?pims\\_id=5472&from=fund](http://www.nsf.gov/funding/pgm_summ.jsp?pims_id=5472&from=fund)

<sup>13</sup> STARS: <http://www.starsalliance.org/>

<sup>14</sup> CSTA: <http://www.csta.acm.org/>

<sup>15</sup> CS4HS: <http://www.cs.cmu.edu/cs4hs/>

<sup>16</sup> NMSI: <http://www.nationalmathandscience.org/index.php/about-nmsi/about-nmsi.html>

<sup>17</sup> New Image of Computing Project Targets High School Students. The WGBH Educational Foundation and ACM are working together on a project, entitled "New Image of Computing" (with NSF funding) to improve the understanding of and image of computing among high school students with a wide-ranging national outreach and communications plan to spread the word about the rewards and benefits of a life in computing.

## Progress to Date

We have already begun the initial phases of this project. The design of the new AP course is underway. Several models of the preAP course already exist. This spring and summer, NSF and Google are cohosting a working group to further develop the preAP course description and select exemplars. This fall, NSF and Google are cohosting a meeting of “Thought Leaders” – directors of science and mathematics in large school districts – to work on issues for the large scale roll out of the new curriculum. We expect that several school districts will begin NSF-funded pilot programs by next fall. A preAP course has already been developed and is being taught in Los Angeles, where their motivation, difficulties, and progress has been documented in a new book, *Stuck in the Shallow End: Education, Race, and Computing*.<sup>18</sup> The Computer Science Teachers Association (CSTA) is involved in all these efforts and will continue to be a close collaborator. The ACM Ed Policy Board has offered to work with us on issues where state-level advocacy will be needed to make the necessary changes in the jumble of computing-related certification, standards, and course classifications.

## Partnerships for Progress

In its entirety, the CS / 10,000 project will require building substantial collaborations beyond NSF. We are seeking to build a consortium of partners – government agencies, community groups, private foundations, and industry leaders – to assist in funding and all aspects of the effort.

**For further information**, contact Jan Cuny, Program Officer for Broadening Participation in Computing, National Science Foundation ([jcuny@nsf.gov](mailto:jcuny@nsf.gov), 703-292-8489).

---

<sup>18</sup> Jane Margolis, Rachel Estrella, Joanna Goode, Jennifer Jellison Holme, and Kimberly Nao, *Stuck in the Shallow End: Education, Race, and Computing*, MIT Press, 2008.