



Voice

MENTORING

The Voice of K–12 Computer Science Education and its Educators

Volume 5, Issue 1

March 2009

Inside This Issue

FEATURES

- Invitation
- Culture-Specific Mentoring to Broaden Participation*
- Filling the Pipeline*
- Proximal Mentoring*
- Get the Conversation Started*

COLUMNS

- Research Review*
- Committee News*
- Classroom Tools*
- Curriculum in Action*
- Spotlight*
- Show me the Numbers*

INFO BRIEFS

- CSTA Thanks Staff*
- CSTA Membership Renewal*
- Contribute to CSTA*
- Contact Info*
- CSTA Source*
- Meet the Authors*
- Mark Your Calendar*
- Resources*

IN THE NEXT ISSUE OF THE VOICE

Building a CS Program



You are invited!

10th Annual CSTA Computer Science & Information Technology Symposium

Saturday, June 27, 2009

*Renaissance Hotel
999 Ninth Street
Washington, D.C.*

CS & IT provides professional development opportunities for high school and middle school computer science and computer applications teachers who need practical, relevant information to help them prepare their students for the future.

*See our terrific list of speakers and sessions
Register for the event*

www.csitsymposium.org

Culture-Specific Mentoring to Broaden Participation

Daniel Frost

What are the interactions between culture and a young person's choice of college major and career? Modern science and technology are largely products of Western culture—does that imply a hurdle for someone who lives in the United States and thinks of

himself or herself as the inheritor of a non-Western tradition? How can our nation broaden participation in computing and embrace a variety of cultures?

These questions are being investigated by the American Indian Summer Institute

CSTA wishes to thank Margot Phillipps

for her role in reshaping
the advocacy content
of *The New Educational
Imperative* giving it
a broader international
perspective.

Staff

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

Pat Phillipps
Editor

Phone: 1-608-436-3050
Fax: 1-928-855-4258
cstapubs@csta.acm.org

Executive Officers

Michelle Hutton
President
mfh@pobox.com

Stephen Cooper
Vice-President
scooper@sju.edu

Committees

Certification
cstacertification@csta.acm.org

Curriculum
cstacurriculum@csta.acm.org

Grants
cstagrants@csta.acm.org

International
cstainternational@csta.acm.org

Membership
cstahelp@csta.acm.org

Professional Development
cstapd@csta.acm.org

Research
cstaresearch@csta.acm.org

CULTURE-SPECIFIC MENTORING

continued from page 1

in Computer Sciences (AISICS) at the University of California, Irvine, in regard to American Indian high school students.

American Indian teenagers are both 21st century digital natives and rising stewards of their traditional cultures. They play video games, use Facebook, listen to stories told by elders from their tribes, and often adopt a worldview informed by Native American culture. They live in cities, suburbs, rural areas, and on reservations. Many want a deeper understanding of both digital culture and their traditional culture.

The primary goal of AISICS is to bridge these two worlds—to inspire interest in computer science (CS) by making it cultural-ly relevant. In July

2008, 18 high-school age American Indian students lived in campus dormitories for three weeks. In the mornings they attended lectures from CS fac-

ulty, covering topics such as programming, computer graphics, interactivity, and software engineering. Afterwards in labs, the participants used MIT's Scratch environment to build interactive, sprite-based projects.

In the afternoons, AISICS students met with two American Indian instructors for study and projects on Native Culture, psychology, crafts, and communication; some afternoon time was also devoted to information sessions on applying to college and finding sources of financial aid. In the evenings, students relaxed, listened to stories told by Native Storytellers, and inter-

acted with the three AISICS counselors, who were all American Indian college students. An important component of AISICS is to expose high school students to computer scientists and to successful, college-educated or college-enrolled American Indian role models. The students' Scratch projects forged a dynamic, creative link between CS and native culture. Each student selected a story from his or her tribal background and turned it into an interactive Scratch project. Many of the stories were creation myths. In most students' projects, the player guides the hero through the narrative by selecting items, making decisions, or solving simple puzzles. In the process of making interactive retellings of their traditions, AISICS students learned

An important component of AISICS is to expose high school students to computer scientists and to successful, college-educated or college-enrolled American Indian role models

that computer systems and traditional knowledge and beliefs are not unrelated or antagonistic, but can support each other.

Measuring the success of a program such as AISICS is difficult. We conducted pre- and post-testing and interviews, and plan on following up with the participants over many years. One encouraging data point was that the number of students who said they were likely to major in CS went up from 12% to 30%. AISICS is scheduled to continue in 2009 with 30 participants and in 2010 with 50 participants. AISICS is funded by NSF grant CNS-0739304.

CSTA Voice ISSN: 1555-2128

CSTA Voice is a publication of the Computer Science Teachers Association.

CSTA Voice is a quarterly publication for members of the Computer Science Teachers Association. It provides analysis and commentary on issues relating to K-12 computer science education, resources for educators, and information for members. The publication supports CSTA's mission to promote the teaching of computer science and other computing disciplines.

Change of Address and Membership Questions: Contact Member Services via email at cstahelp@csta.acm.org, or call 1-800-342-6626 (U.S. & Canada) or +1-212-626-0500 (Global).

Reproduction Rights Information: No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording or information storage and retrieval system, without permission in writing from the publisher. Exception: permission to photocopy (individual) items for internal or personal use is hereby granted by CSTA.

Criteria for submitting articles: Potential writers for CSTA should send a brief description of the proposed article, estimated word count, statement of value to members, author's name & brief bio/background info, and suggested title to the editor at cstapubs@csta.acm.org. The final length, due date and title will be negotiated for chosen articles.

Notice to Authors Contributing to CSTA Newsletter: By submitting your article for distribution in this publication, you hereby grant to CSTA the following non-exclusive, perpetual, worldwide rights:

- to publish in print on condition of acceptance by the editor
- to digitize and post your article in the electronic version of this publication
- to allow users to copy and distribute the article for noncommercial, educational or research purposes

However, as a contributing author, you retain copyright to your article and CSTA will make every effort to refer requests for commercial use directly to you.

Filling the Pipeline

Students Mentoring Students

Jeff Gray

According to a recent Lemelson-MIT Invention Index, a majority of teens may be turned off to technology careers because they “do not know anyone who works in these fields or understand what people in these fields do” (web.mit.edu/invent/). Mentors have opportunities to improve the understanding and awareness of a career in computing.

Starting a mentoring program, however, is fraught with many challenges. For example, students often lack a basic knowledge of computational concepts and hold many misconceptions. Over the past five years, I have found the following three-phased approach beneficial for building a sustainable mentoring pipeline through layered stages of interaction.

Phase 1 - Interest and Awareness: The first phase of interaction provides an information session that can influence a large contingent of students. The goal of this phase is to introduce core elements of computational thinking through a mixture of “roadshows” (visiting students in their classroom) and field trips (hosting student visits to a computer science (CS) department). On average, a dozen roadshows and a handful of field trips can provide an opportunity to reach over 400 students each year in grades K–12.

Each 90-minute roadshow or 3-hour field trip may introduce topics such as:

- Activities from CS Unplugged
- “Why Study Computer Science” presentations
- Example robotics exercises
- A brief introduction to Alice

Phase 2 - Initial Exploration: A subset of students from the first phase can be invited to participate in the second segment of the pipeline, which equips them with basic skills through more intense interaction. Students participating in this phase are given an opportunity to become

familiar with a programming language (e.g., Alice for elementary/middle school, and Java for high school) and to learn more about other aspects of CS (e.g., computer forensics). Example interventions may include attending a multi-week summer computer camp or enrolling in an introductory course at a local university. Our outreach involves a series of state-wide competitions that are offered for all grade levels. The students completing this mentoring phase will have the skills to compete in these competitions with peers from other schools. This phase typically provides interaction with 50–75 students in grades 5–12 over one or more weeks.

Phase 3 - Deep Skill Building through a Research Experience: The final phase of mentoring develops a close relationship with a very small group of students. Often, the objective of this mentoring phase is to provide a research experience for each student with a project that is competitive at a local science fair. We provide students in this phase with office space and a desktop alongside Ph.D. students in my lab, where students work 5–10 hours per week. A college freshman we mentored with this approach recently presented a research paper (based on his high school experience) at a regional ACM conference—a first in the 30-year history of the conference. This intense level of mentoring offers an opportunity to interact with two to three students in grades 10–12 over a full academic year.

In this staged approach of mentoring, the early phase has a short period of engagement across a large number of students, and the later phases have a longer period of mentoring for a small number of students. Details about some of the activities used across the mentoring pipeline are available at www.cis.uab.edu/gray/outreach/outreach.html.

Correction: Beth Kraemer, Information Technology division, University of Kentucky Libraries, authored *Virtual College Fair* (January 2009). She and Kelly Czarnecki were co-organizers of the college fair held in Second Life.



When you receive your email notice, renew your **FREE CSTA membership.**

Contribute to the CSTA Voice

The editorial board of the *CSTA Voice* is dedicated to ensuring that this publication reflects the interests, needs, and talents of the CSTA membership. Please consider sharing your expertise and love for computer science education by contributing newsletter content.

Potential writers for the *CSTA Voice* should send a brief description of the proposed article, estimated word count, statement of value to members, author's name and brief bio/background info, and suggested title to the editor at: cstapubs@csta.acm.org. The final length, due date, and title will be negotiated for chosen articles. Please share your knowledge.

Volunteer today!

The *CSTA Voice* welcomes your comments.

E-MAIL: cstapubs@csta.acm.org

PHONE: 1-608-436-3050

FAX: 1-928-855-4258

Letters to the Editor are limited to 200 words and may be edited for clarification.



ACM founded CSTA as part of its commitment to K–12 computer science education.

CSTA
Source
Web
Repository

Your source
for K–12
CS Teaching
Materials

Search for resources

- ♦ Shared by members
- ♦ Pedagogically valid
- ♦ Relevant & current
- ♦ Technically sound

Contribute resources

- ♦ Help your peers
- ♦ Quick & easy
- ♦ Retain the copyright
- ♦ Submit online

csta.acm.org

Proximal Mentoring

Teachers Mentoring Teachers

Susan E. Gunn

Some school districts are experiencing both a financial crisis and an experience crisis. Financially, many school districts are being forced to cut back on a number of programs. In my school district, one of the items on the financial chopping block for next school year is the new-teacher mentoring program. This is devastating in a field where more than 30% of new teachers quit in the first three years and at least 50% by the fifth year of teaching.

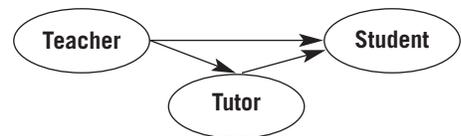
At the same time, many school districts are also seeing the retirement of their most experienced teachers, leaving a huge gap in the experience level of current teachers. This gap in the experience level further confounds the successful integration of novices into the teaching discipline.

It may be time to rethink the current mentoring paradigm in education. What if teachers in their second and third years of teaching become what have been labeled as Proximal Mentors to the brand-new teachers, while still having the traditional mentor with more than 15 years of experience guiding all of them? The current research on this theory of mentoring in graduate education suggests that providing Proximal Mentors to novices improves the perceptions of the novice while allowing the Proximal Mentor to become more experienced within the field under the guidance of the traditional mentor. Perhaps turning the focus of the second or third year teacher towards helping a new teacher become successful will help stem the tide of teachers who leave within those first five years.

The concept of Proximal Mentoring stems from the work of Russian Psychologist Lev Vygotsky on learning and development. He identified two levels of development: actual and proximal. The actual developmental level represented those skills that learners know and/or can demonstrate on their own. The proximal developmental level is a state of readiness in which the learner is prepared to learn. Vygotsky named this state the

Zone of Proximal Development (ZPD).

In our current educational paradigm, the teacher and/or other adults who interact with the students in an academic role represent “adult guidance” whereas the “more capable peers” are the peer tutors or learning partners. A visual representation of this relationship can be expressed as:



In attempting to extend the ZPD into the teacher mentoring paradigm, there was no equivalent for the academic tutor in the mentoring paradigm. Traditionally, a mentor is defined as someone with 15 or more years of experience in her/his area of expertise who guides a mentee to become successful within a discipline. The mentee is a novice, someone who is new to the field. Gunn (2008) proposed a new construct in the teacher mentoring paradigm—the Proximal Mentor:



A teacher Proximal Mentor does not have the years of expertise in the domain that the traditional teacher mentor has. However, the Proximal Mentor has more experience than the mentee (new teacher). A teacher Proximal Mentor provides assistance with content knowledge, much like a tutor, while also providing guidance and serving as a role model much like a Mentor. In addition, the teacher Proximal Mentors can bring a sympathetic understanding of what it is like to be new to a profession. They can provide a unique perspective of the difficulties of the here-and-now, guiding the mentee in

navigating difficulties that the traditional mentor may never have experienced.

For both the traditional mentor and the Proximal Mentor, the personal goal of the mentoring process is the successful integration of the novice into the discipline. Another goal for the Proximal Mentor is to gain additional experience within the domain while still under the guidance of the traditional mentor.

We do not have to wait until someone decides to implement a new mentoring paradigm. We can start with a grass-roots effort amongst ourselves. If you are new,

volunteer to help someone who is even newer to the profession. If you are experienced, suggest to your second or third year mentees that they take on the role of Proximal Mentor to someone who has less experience. There is no rule that says you have to have a lot of knowledge in order to share what you know.

References: Gunn, S. E. (2008). An exploratory study of proximal mentoring in graduate education. Retrieved January 18, 2009 from www.msgunn.com/Proximal_Mentoring/PM.htm.

Get the Conversation Started

NCWIT Talking Points

Lecia Barker

Talking Points is a series of easy-to-use conversation cards designed to encourage girls to consider Information Technology (IT) careers by providing facts and dispelling myths related to women in IT. The first in the series targets parents and other adults who influence girls. It includes discussion topics important to girls when planning their careers.

- Meaningful work
- Security and salaries
- Flexibility, variety, and creativity
- Work environments
- Preparation strategies

The details in the *Talking Points* are based in research on girls, parents, and other adults. Research suggests that many young people, especially those who will be first in their families to attend college, think most often of “doctor” and “lawyer” when they think of high-status or high-salary careers. Readers learn that a bachelor’s degree in CS can provide a high, entry-level salary. Because many parents and counselors believe that offshoring has reduced the number of IT jobs, job growth predictions from the U.S. Department of Labor are included. By using photographs of people working together and describing the team-oriented nature of IT work, we also hope to overcome the myth that IT careers are solitary, isolating jobs.

We assume girls will be a secondary audience (i.e., parents might leave it on the kitchen counter and their daughters will see it), so the card is designed to be visually appealing and engaging to both adults and girls. The photographs include people of many races and ethnicities in the hope that readers will see in them a “possible self.”

Look for future *Talking Points* to debunk myths about technology jobs

NCWIT Talking Points, which are printed on a single 5 × 8 card, are easy to read and share. They are free and can be downloaded at www.ncwit.org.

Videos and information on successful women and various IT careers, free downloadable software for getting started (e.g., Scratch), and other information are also available on the NCWIT site. A Spanish-language version *Talking Points* will be available soon.

Look for future *Talking Points* to debunk myths about technology jobs, describe attractive IT careers, emphasize the usefulness of IT skills in a variety of careers, and provide concise information on why women’s participation is important.

Meet the Authors

Mordechai Ben-Ari

Weizmann Institute of Science, Israel
Moti Ben-Ari is an associate professor. He received the 2004 SIGCSE Award for Outstanding Contributions to CS Education.

Lecia Barker

NCWIT

Lecia is a research scientist for the National Center for Women & Information Technology at the University of Texas – Austin, TX.

Daniel Frost

University of California - Irvine

Dan is a Lecturer in the Informatics and Computer Science departments.

Jeff Gray

University of Alabama - Birmingham

Jeff is an Associate Professor of CIS and recipient of the NSF CAREER award. He was recently named the Alabama Professor of the Year.

Susan Gunn

Clark County School District, NV

Susan teaches advanced computer subjects at the Advanced Technologies Academy.

Ronit Ben-Bassat Levy

Weizmann Institute of Science, Israel

Ronit teaches high school CS and mathematics and is a Ph.D. student at the Weizmann Institute of Science.

Patricia Medina

Clear Creek ISD, TX

Patricia teaches and has presented at the Texas Computer Education Association Conference.

Karen North

Houston ISD, TX

Karen is a Technology Specialist at Piney Point Elementary and is working on the *ACM K-8 CS Model Curriculum*.

Margot Phillipps

Lynfield College, Auckland, New Zealand

Margot teaches mathematics and CS and is the International Representative on the CSTA Board of Directors.

Research Review

CS in Texas

Karen North

Recent data from the Texas Education Agency (TEA) suggest that computer science (CS) education can help prevent students from dropping out of school before completing their high school education.

The TEA data shows that students who take just one CS class have a 17% better chance of graduating from high school than those who do not take a CS course. The high school dropout rate of CS course takers is 1.4%, compared to the Texas state average of 8.8%. Although this data does not establish a cause-and-effect relationship, it provides interesting information about a possible relationship between CS education and student academic success.

The data also raises interesting questions about the importance of providing students with an opportunity to take courses that emphasize conceptual knowledge, algorithmic thinking, and problem-solving skills. Statistics show many students drop out because they lack challenge and motivation. CS has the advan-

tage of offering reasonable success to average students, and presenting a scope that is broad enough and deep enough to continually challenge students. For gifted students, CS offers an opportunity to challenge their capabilities and to utilize their creativity in ways that many other courses may not.

Recently, Texas has taken important steps to support K–12 CS education by ensuring that rigorous CS classes count toward a student’s core graduation requirements in math. Texas students now have the choice of AP CS in the 4 × 4 plan to satisfy the required fourth year of math. This change came about as the result of extensive advocacy by Texas educators at both the K–12 and post-secondary level. The first step in changing the policy involved educating state decision makers on the difference between technology literacy and CS. The next step was demonstrating the connections between CS, computation, and mathematics.

These results in Texas point to the critical importance of advocacy on behalf of CS education and to the continued need to help administrators understand the potential of CS to teach higher-level problem-solving skills and creative thinking that are key to student engagement and academic success.

Texas Longitudinal Completion Rates, Grades 9-12, Students Taking and Passing at Least One CS Course During High School, Class of 2006

Group	Class	GRADUATED		RECEIVED GED		DROPPED OUT	
		Number	Rate (%)	Number	Rate (%)	Number	Rate (%)
CS course takers	13,388	12,967	96.9	41	0.3	181	1.4
State	283,698	227,975	80.4	6,456	2.3	24,975	8.8

“When compared to all students, those who took and passed at least one CS course had a much higher rate of graduation and a much lower rate of receiving GEDs ... or dropping out.”

Karen Kahan, Director, Educational Technology, Texas Education Agency

Committee News

The New Educational Imperative: International Version

In an effort to better serve its more than 1100 international members, CSTA has released a new version of *The New Educational Imperative: Improving High School Computer Science Education*.

According to CSTA’s International Committee Chair Margot Phillipps, this publication looks at the issues and research from a broader international perspective, attempting to frame the discussion in a way that focuses on computer science (CS) as an international educational concern.

“When the original *New Imperative* report was written, we framed the discussion about high school CS as a U.S. concern” say Phillipps. “Our goal now, however, is to reexamine the key issues from a broader international perspective and to determine a common set of actions that can be taken in any country to improve the opportunity for students to develop a strong founda-

tion in CS before they move on to post-secondary programs.”

Because the original research for the publication was international in scope, it was obvious that the document provided a broad perspective on the issues relating to CS education. Revising the document therefore involved taking a close look at the use of U.S.-specific terminology and assumptions about the role and breadth of advocacy efforts that could support and improve CS education in other countries.

According to CSTA Executive Director Chris Stephenson, the new document was really made possible by the dedicated work of the association’s international volunteers. “Margot (New Zealand) played an essential role in helping us reshape the advocacy sections and both she and our international Advisory Council member Judith Gal-Ezer (Israel) helped us to identify much of the U.S.-focused terminology” said Stephenson.

To make the document more accessible for international members, CSTA also decided to make it freely available as a pdf. The document can be downloaded directly from the CSTA Web site at: csta.acm.org/Resources/sub/Documents.html

Classroom Tools

Teaching Introductory Programming with Jeliot

Ronit Ben-Bassat Levy and Mordechai Ben-Ari

It is not news to computer science (CS) teachers that a deep understanding of even simple programming concepts can be difficult for students to acquire. However, student understanding can be improved by using a program animation system that dynamically

illustrates the execution of a program or segment of code so that students can “see” what is happening in the computer.

Jeliot is a code animation system that accepts a Java program as input and automatically creates an animation of its execution including memory allocation, control flow, and expression evaluation. The user interface consists of one window split into two panels; the left panel displays the source code for editing and for highlighting the source line that is being animated, and the animation takes place in the right panel. VCR-like buttons control the execution of the program animation.

Jeliot produces instantaneous animation, allowing teachers to demonstrate how to modify programs in response to questions and to review concepts with a history of recent animation frames. Teachers can also add annotations to point out important events, for example, “leaving the while loop.”

Students can use the visualizations to build a vocabulary for talking about programs and for self-study. Jeliot has proved to be especially suited for improving the learning of average students who are capable of understanding abstract CS concepts, but need some help to deepen their comprehension.

Jeliot was developed as a collaborative project supervised by Mordechai Ben-Ari at the Weizmann Institute of Science in Israel and Erkki Sutinen at the University of Joensuu in Finland. It is cur-

rently in use in Israel, Finland, Germany, Spain, the U.S., and Brazil.

Ronit Ben-Bassat Levy uses it in classes at the Weizmann Institute as a demonstration tool for explaining concepts and as a development environment in student labs in courses with a syllabus similar to the Advanced Placement courses. Ronit believes that Jeliot encourages students to think analytically when developing and designing algorithms. It also allows the instructor to hide the source code and ask the students to write a program that gives rise to the execution shown in the animation.

Jeliot is freely available under the GNU GPL (both as a stand-alone application and as a BlueJ plugin) from cs.joensuu.fi/jeliot/. There are links to examples and tutorials, research papers, an introductory video, and a set of learning objects.

Curriculum in Action

Gaming: The Next Step

Patricia Medina

Long before the tardy bell rings, students stream into the classroom, immediately head for their computers, and begin work on their latest assignment. Soon students are moving from one computer to another, looking at other students’ work, and sharing ideas. The instructor Patricia Medina moves around the room answering questions, stops to look at a student’s work, and, in some cases, offers a helpful hint. This is the scene in Medina’s computer science (CS) gaming class at Clear Brook High School in Friendswood, Texas.

“Over 92 percent of high school students play video games,” said Medina. “If our students are already playing games, why not use that to teach CS concepts?”

According to Medina, students come into this class wanting to learn instead of having to learn, which is motivating for both the

students and the teacher. “Wouldn’t we all like to teach a class where students are anxious for the next lesson?”

Medina came upon the idea for the game class in search of a better way to teach CS in the Clear Creek Independent School District. At the Texas Computer Education Association conference (TCEA), she discovered the book, *Python Programming for the Absolute Beginner*, by Michael Dawson. After reading the book, Medina decided she could use gaming to teach CS.

Medina and colleague Anne Woolweaver, who teaches at Clear Creek High School in League City, Texas, developed a curriculum for the class aligned with Texas state standards. The curriculum includes more than creating games; students explore the process from development of an idea to marketing. Students develop a business plan, explore the market place to assess consumer interest, and navigate copyright issues.

“While creating their program, the students are constantly challenging themselves,” she said. “They want their characters to be unique and have special features, so they learn the CS concepts of attributes, parameters, variables, controls, and much more.”

Spotlight

A Challenge For Girls in New Zealand

Margot Phillipps

The Young Women’s Programming Contest initiative from 20 years ago was recently revitalized as New Zealand’s (Alice) Programming Challenge for Girls (PC4G) at Auckland University of Technology (AUT). The November 2008 PC4G event drew 40 girls from 14 high schools across New Zealand. In two hours they learned basic Alice programming concepts and strategies, and then applied them in a two-hour challenge activity.

The latest educational research on teaching technology to girls was applied. Leaders selected problems girls could easily relate to, taught the content with inclusive language, and fostered social collaboration by organizing the learners into teams.

The lessons and activities focused on “Beetles” playing music. During the challenge, the teams of two were tasked with creating an aerobics dance routine for five “people” (Alice objects) to a choice of three supplied pieces of music. The girls applied problem-solving techniques as they created a solution storyboard and then programmed the solution using Alice. Gold, silver and bronze medals were awarded, and prizes included small purses, book tokens, lip gloss, and funky stationery. Prizes were provided by the Manukau Institute of Technology and AUT’s Women on Campus.

Alice was chosen as the programming environment because it is highly motivational. It provides instant feedback, the need for technical knowledge is minimal, and problem solutions can be created quickly.

The forerunner to this challenge was called the Young Women’s Programming Contest. It was initiated in 1988 by Margot Phillipps and ran for 11 years. The event was taught using a specially written assembly-level programming language (SPLAT) developed by Phillipps and Gordon Grimsey from AUT. Over the years, the contest was refined by Dr. Gay Costain. The current PC4G is the project of Anne Phillpott, Jacqueline Whalley, Costain, Phillipps and other volunteers.

SHOW ME THE NUMBERS			
Advanced Placement Exam Takers			
	2003	2008	% change
APCS-A & AB	21,745	20,532	-6
AP Biology	103,944	154,504	+49
AP Calculus	212,794	291,938	+37
AP Chemistry	65,698	100,586	+53
AP Physics	71,436	98,276	+38
AP Statistics	58,230	108,284	+86
Note: AP CS has increased in the last 3 years 2006: 3%, 2007: 3%, 2008: 2%			
SOURCE: College Board: professionals.collegeboard.com/educator/K-12-teacher			



We're on the Web! csta.acm.org

MARK YOUR CALENDAR

SIGCSE 2009

March 4–7, 2009 in Chattanooga, Tennessee
www.cs.arizona.edu/groups/sigcse09

Computer Using Educators (CUE)

March 5–7, 2009 in Palm Springs, California
www.cue.org/

Richard Tapia Celebration of Diversity in Computing

April 1–4, 2009 in Portland, Oregon
tapiaconference.org/2009

Consortium for Computing Sciences in Colleges

(CCSC: Mid-South)
April 3–4, 2009 in Martin, Tennessee
www.ccsc-ms.org

Consortium for Computing Sciences in Colleges

(CCSC: Central Plains)
April 3–4, 2009 in Bolivar, Missouri
www.ccsc.org/centralplains

Consortium for Computing Sciences in Colleges

(CCSC: Southwestern)
April 3–4, 2009 in San Diego, California
www.ccsc.org/southwestern/2009

Consortium for Computing Sciences in Colleges

(CCSC: South Central)
April 24–25, 2009 in Hammond, Louisiana
www.sci.tamucc.edu/ccsc

Consortium for Computing Sciences in Colleges

(CCSC: Northeastern)
April 24–25, 2009 in Plattsburg, New York
www.ccscne.org/2009

International Conference on the Foundations of Digital Games

April 26–30, 2009 leaving from Port Canaveral, Florida
foundationsofdigitalgames.org

UAB High School Programming Contest (HSPC)

May 10, 2008 in Birmingham, Alabama
www.cis.uab.edu/programs/hspc/

Oregon Game Programming Challenge (OGPC)

May 16, 2009

UAB Alice Festival

May 16, 2009; Birmingham, Alabama
www.cis.uab.edu/programs/alice-festival

ACSL All-Star Contest

May 23, 2009 in Huntsville, Alabama
www.acsl.org

School Programming summer camps

June–July 2009 in Birmingham, Alabama
www.cis.uab.edu/programs/camps

Computer Science and Information Technology Symposium (CS & IT)

June 27, 2009 in Washington, D.C.
www.csitsymposium.org

NECC 2009

June 28–July 1, 2009 in Washington, DC
center.uoregon.edu/ISTE/NECC2009

The 21st International Olympiad in Informatics

August 8–15, 2009 in Plovdiv, Bulgaria
ioinformatics.org

CSTA INSTITUTIONAL MEMBER K–12 OUTREACH PROGRAMS

TechTopia Challenge - Neumont University

July 31, 2009 in Jordan, Utah
www.tech-topia.com

RESOURCES

Here's more information on topics covered in this issue of the *CSTA Voice*.

Page 1: CS & IT Symposium www.csitsymposium.org

Page 1: American Indian Summer Institute in CS (AISICS) www.cfep.uci.edu/airp/aisics.php

Page 1: Scratch scratch.mit.edu

Page 3: Lemelson-MIT Program web.mit.edu/invent

Page 3: UAB K–12 Outreach Activities www.cis.uab.edu/gray/outreach/outreach.html

Page 3: UAB CS Summer Camps www.cis.uab.edu/programs/camps

Page 3: UAB Alice Film Festival www.cis.uab.edu/programs/alice-festival

Page 3: UAB HS Programming Contest www.cis.uab.edu/programs/hspc

Page 3: CS Unplugged csunplugged.org

Page 4: Proximal Mentoring www.msgunn.com/Proximal_Mentoring/PM.htm

Page 5: NCWIT www.ncwit.org

Page 5: NCWIT Talking Points www.ncwit.org/pdf/TP_ParentsResource_cards.pdf

Page 5: Gotta Have IT www.ncwit.org/ghit

Page 5: Outreach-in-a-Box www.ncwit.org/resources.res.box.html

Page 5: Promising practices for K–12 CS www.ncwit.org/resources.res.practices.php

Page 6: Texas Education Agency www.tea.state.tx.us

Page 6: The Political Landscape: Advocating for CS csta.acm.org/About/sub/CSTAPresentations.html

Page 6: The New Educational Imperative: Improving HS CS Education
csta.acm.org/Communications/sub/Documents.html

Page 7: Alice www.alice.org

Page 7: College Board www.collegeboard.com

Page 7: Python Programming Language www.python.org

Page 7: Jeliot cs.joensuu.fi/jeliot



CSTA report on certification:

Ensuring Exemplary Teaching in an Essential Discipline

csta.acm.org/Communications/sub/DocsPresentationFiles/CertificationFinal.pdf