

VOICE

CSTEACHERS.ORG
COMPUTER SCIENCE TEACHERS ASSOCIATION

THE VOICE OF K-12 COMPUTER SCIENCE EDUCATION AND ITS EDUCATORS

Volume 12 / Issue 5
November 2016

CSEdWeek

FEATURES

CSEdWeek...Don't Miss It!
CSEdWeek: Join the Biggest Learning Event in History
The 2016 CSTA CS Standards
Updated Unplugged Resources from New Zealand
Computational Thinking Video Series
Congratulations, CSTA Member

COLUMNS

Professional Development
Classroom Tools
Member Recognition
Reflecting on CS Education
Teaching About CS Careers
Sponsor News

INFO BRIEFS

Keep Up with CSTA!
Staff
Contribute to the Voice
Contact Info
Meet the Authors
Mark Your Calendar

IN THE NEXT VOICE
CSTA Chapters



CSED WEEK...DON'T MISS IT!

Dave Reed

As the calendar begins to wind down on 2016, it's time to get excited about your plans for Computer Science Education Week (CSEdWeek). CSEdWeek—this year from December 5 to 11—is an annual event designed to inspire students and raise awareness of the importance of K-12 CS education.

It's the perfect time to recognize CS students, promote computing clubs, and educate the community. More than 200,000 educators have supported CSEdWeek activities since it was first celebrated in 2009. If you are not one of those educators, it's time to get involved! You'll also find lots of ideas on the CSEdWeek website (csedweek.org).

The Hour of Code is a very important part of CSEdWeek. Since 2013, Code.org has challenged schools to engage their students in at least one hour of CS content. To date, more than 100 million students from more than 180 countries have taken part in Hour of Code activities. The Hour of Code website (hourofcode.com) provides coding tutorials for all age groups and unplugged activities that introduce students to computational thinking and CS.

This year, CSEdWeek will have special significance for CSTA. CSTA will announce the inaugural recipients of the Infosys Foundation USA/ACM/CSTA Awards for Teaching Excellence in Computer Science. This new award program will recognize up to ten outstanding K-12 CS teachers from around the world. Awardees will each receive a \$10,000 prize, thanks to the generosity of Infosys Foundation USA (www.csteachers.org/?page=CSTeachingAwards)

Celebrate CS education!



KEEP UP WITH CSTA!

CSTEACHERS.ORG

Twitter

(csteachersorg)

Facebook

(Computer Science Teachers Association)

LinkedIn

(search for computer science teachers association)



BOARD OFFICERS

David Reed, *Chair*

DaveReed@creighton.edu

STAFF

Mark R. Nelson, *Executive Director*

Phone: 518.281.2975

m.nelson@csteachers.org

Lissa Clayborn, *Deputy Executive Director/
Chief Operating Officer*

l.clayborn@csteachers.org

Pat Phillips, *Editor*

Phone: 608.436.3050

publications@csteachers.org

VISIT CSTA ONLINE

Your CSTA website is changing!

A new...Look, Feel, & Member Experience

WWW.CSTEACHERS.ORG

CSEDWEEK: JOIN THE BIGGEST LEARNING EVENT IN HISTORY

Cameron Wilson

Editor's Note: CEdWeek is an important event in thousands of schools... but unless you've participated from the very first event in 2009, you might not know the rich history and broad reach of this amazing learning event. In this interview, Cameron Wilson from Code.org, gives us a historical perspective and offers suggestions for a successful event.

CSTA: How did CEdWeek start?

WILSON: The idea for Computer Science Education Week (CEdWeek) came from Congressman Vernon Ehlers (R-MI) after he met with Joel Adams, a constituent of the Congressman and a computer science (CS) professor at Calvin College. Joel raised the alarm that K-12 CS wasn't widely taught in schools, despite the fact that lots of unfilled jobs require computing skills and knowledge.

The Congressman brought the idea to the Association for Computing Machinery (ACM). ACM saw CEdWeek as a platform to elevate awareness of K-12 CS issues. The organization took the lead in pulling a community together around CEdWeek with support from Microsoft, CSTA, and NCWIT, along with a few other key partners.

ACM chose to observe CEdWeek during the week of December 9 to honor the birthday of Grace Hopper, an early computer programmer, a US Navy rear admiral, and a continuing inspiration to computer scientists everywhere. The first CEdWeek ran from December 6-12, 2009.

Computing in the Core, a nonpartisan coalition with the goal of elevating CS to a core academic subject in K-12 education, took over the work of running CEdWeek until 2013, when that organization merged with Code.org.

CSTA: What were the initial goals? How have they changed over time?

WILSON: CEdWeek was initially focused on policy issues and policy

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 customerservice@csteachers.org or call 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.

makers—political influencers who could make CS education a part of foundational learning. Early wins in this regard included a 2012 proclamation by NYC Mayor Bloomberg, as well as celebrations by dedicated CS teachers.

But the appeal of CSEdWeek has since grown to a much broader audience. K–12 teachers (both with and without a CS background) have become the driving supporters of the week’s activities, and other participants now include corporations and nonprofit organizations, district administrators, superintendents, boards of education—and, of course, parents and students all over the world.

CSTA: How has participation grown?

WILSON: By 2012, CSEdWeek had reached a scale of hundreds of events, including ones in a number of countries outside the US (particularly in the UK and Australia). After Code.org introduced the Hour of Code theme for CSEdWeek in 2013, participation grew exponentially. Supported by new activities that students could do either on their own or with teacher lesson plans, and by added operational support and funding, the first “Hour of Code”-themed CSEdWeek reached over 15 million students and over 35,000 events across 167 countries.

Today, CSEdWeek and the *Hour of Code* continue as a grassroots campaign supported by 400 partners and 200,000 educators worldwide, making it the largest education campaign in history. Over 100 million students in more than 180 countries across the globe have participated.

CSTA: What suggestions do you have for planning a successful CSEdWeek?

WILSON: Introducing children to CS through an hour-long activity can help to de-mystify the world of computing, break stereotypes that exist around coding, and spark a lifelong passion. You can also engage an existing CS class to reach out to non-CS students! This broadens the appeal of CS. And working with an entire school or district on an Hour of Code event can help spread access to CS far beyond your classroom. Follow these simple steps to plan an event.

1. Choose an activity—and this year, the options are much broader.

Visit csedweek.org and explore fun activities for students of all ages, created by a variety of partners and for a variety of subjects. In the past, CSEdWeek and the Hour of Code featured mostly one-hour, self-guided tutorials, but starting in 2016, the activities will include multi-hour options, as well as open-canvas coding activities and teacher-led lesson plans. In particular, the 2016 activities will include lesson plans targeting non-CS classrooms so that English, history, or science teachers can find activities that relate to their classrooms.

2. Plan your technology needs—computers are optional.

No downloads or sign-ins are required to use *CSEdWeek.org* activities, and many unplugged tutorials are available. If your school doesn’t have enough computers or Internet access, many of the one-hour activities will also work on smartphones and tablets.

3. Spread the word to students and parents.

Inspire students with videos and posters. Find everything you need at hourofcode.com/resources.

Host a school-wide assembly. Show a video or invite a speaker to do an “unplugged” activity with students in front of the entire student body. Showcase student projects from your CS classroom. Reward participants. Go big and create prizes and awards for teachers and students!

Invite the community. Parents, grandparents, volunteers, and business leaders can all learn to code. In high schools, CSEdWeek provides a fantastic opportunity to recruit new students for future CS courses. Encourage parents to participate, too! Students with engaged parents are more likely to pursue CS. See a sample email to parents at hourofcode.com/promote/resources.

4. Celebrate!

Share your experience and photos on social media with [#csedweek](https://twitter.com/csedweek) [#HourOfCode](https://twitter.com/HourOfCode).

Let us know if your contact information changes.
customerservice@csteachers.org

THE CSTA VOICE WELCOMES YOUR COMMENTS

publications@csteachers.org 1-608-436-3050
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.

**CSEDWEEK
2016**

**DECEMBER 5–11
WWW.CSEDWEEK.ORG**

IDEAS

Local promotions

Student showcases

Hour of Code

Parent events

Newspaper articles

Letters to the editor

Field trips

Guest speakers

Cross-curricular
projects

School announcements

Each one, teach one

Mini-sessions

Administrator
bootcamps

**CSEDWEEK
2016**

THE 2016 CSTA CS STANDARDS

Deborah Seehorn

The past year has been a banner year for K–12 computer science (CS) education. While work was underway on the revisions to the 2011 CSTA K–12 CS Standards, and work on the newly developed K–12 CS Framework (*k12cs.org*) was also underway, we all heard a wonderful announcement from President Obama about the “CS for All” initiative (*www.csforall.org*). That was music to the ears of any CS educator who had advocated for inclusive K–12 CS education.

These last two years will definitely go down in the record books as a great era for CS education. Throughout the US, states, cities, and local school systems have begun new, or strengthened existing, K–12 CS education programs. Curriculum developers are busy addressing the needs of K–12 CS students and teachers. And CSTA has been working on a significant revision to the CSTA K–12 CS Standards (now in interim status on the CSTA website at: *www.csteachers.org/?page=CSTA_Standards*).

You may be wondering why the standards were released in interim status before the Framework was released. Teachers and curriculum providers were interested in having a revised version of the CSTA K–12 CS Standards to start the 2016–2017 school year, so the decision was made to release the standards in interim form, and make any needed revisions after the Framework was released.

CSTA began the revision process last fall by forming the CSTA K–12 CS Standards Task Force comprised of 14 outstanding CS educators—with CS educational expertise ranging from kindergarten through grade 12, community college, and university.

The Task Force made a conscious

decision early in the process to address the K–12 CS Framework in writing the CSTA standards. The Framework process began just before the Task Force met in person for the first time. When the Task Force met, the members completed a gap analysis to determine where there were needs for additional standards to address the Framework. Additionally, several of the Task Force members also participated in the development of the Framework—as advisors or writers of the Framework concepts and practices.

After the in-person meeting, the Task Force members began collaborating virtually to revise, clarify, add, and delete standards from the 2011 version of the CSTA K–12 CS Standards as they addressed the core concepts and practices in the Framework. The goal of the Task Force members was to write clear, measurable, rigorous learning standards that every student should master in grades K–12. Each standard addresses one core concept and one core practice in the K–12 CS Framework. Not every concept and not every practice is addressed at every grade level, but the majority of the concepts are.

Public feedback on the CSTA Standards was solicited at two different points during the year and the Task Force members responded to the feedback and adjusted standards as necessary. The Interim CSTA K–12 CS Standards were released at the 2016 CSTA Annual Conference. More feedback and input was obtained from participants during a conference session on the revised standards.

The 2016–2017 Task Force members have been eagerly awaiting the release of the Framework, which was just released on October 17. The Task Force members will now take a deeper look at

the Standards and make needed adjustments. The Task Force will also be analyzing related educator needs for the implementation of the Standards. Our goal is to have the finished version of the Standards released next spring—or at the 2017 CSTA Annual Conference, at the latest.

You will find a downloadable PDF file of the Interim CSTA K–12 CS Standards on the CSTA website (www.csteachers.org/?page=CSTA_Standards). You will also find information about the Task Force members and the revision process.

Additionally, you can view a video about the CSTA K–12 CS Standards and CSTA on the CSTA homepage. Please take the time to become familiar with the CSTA K–12 CS Standards and plan to make CS education available to all K–12 students in your school, district, or state.

org/?page=CSTA_Standards). You will also find information about the Task Force members and the revision process.

UPDATED UNPLUGGED RESOURCES FROM NEW ZEALAND

Tim Bell

The *CS Unplugged* project and the associated *CS Field Guide* are undergoing major improvements after receiving some impetus from changes in the New Zealand curriculum, and with continued funding support from industry.

The New Zealand Ministry of Education announced in July 2016 that the subject of Digital Technologies (which includes programming and topics from computer science (CS)) will become a part of the national curriculum beginning in 2018 (www.edgazette.govt.nz/Articles/Article.aspx?ArticleId=9323). The curriculum will cover all years of schooling (years 1 to 13) and incorporates six areas: algorithms, data representation, digital applications, digital devices and infrastructure, humans and computers, and programming.

This builds on six years of experience with such topics being offered at the senior secondary school level. To support these classes, the CS education group at the University of Canterbury (Christchurch, NZ) developed the *CS Field Guide*. The *Field Guide* is an online teaching resource that tackles advanced CS topics (such as algorithms, HCI, graphics, vision, and software engineering) at a

level appropriate to the current NZ assessments. The *Field Guide* was inspired by a project initiated by Peter Denning and has grown into an interactive online resource that is approachable for people with little background in CS. The current version is still in beta but is widely used (and like many online projects, is likely to be in “beta” pretty much forever!)

Most recently it has been restructured so that it is delivered through GitHub, to make it fully open source. Already, a number of helpful contributions have been received, and this makes it easy for others to host and adapt their own versions.

The new structure allows for translations (with some projects underway to produce these), and the entire *Field Guide* can be downloaded as a printable PDF for those who like to have paper versions. It also allows for connecting it to other curricula, and a guide to using it for the Advanced Placement CS curriculum (in the US) has been developed by James Atlas at the University of Delaware (available at: www.csfieldguide.org.nz/en/curriculum-guides/apcsp).

Many of the “interactives” (small,

MEET THE AUTHORS

Tim Bell

University of Canterbury
Tim is a professor of CS in Christchurch, NZ. He has been involved in the design and deployment of the new CS curriculum in NZ.

Toby Bradley

Pascagoula-Gautier School District
Toby works as the Network Supervisor with the Pascagoula-Gautier Schools in Mississippi.

Abby Bouchon Daniels

Google
Abby works on Google’s K12 Education Outreach team, which focuses on increasing access and exposure to CS education for students traditionally underrepresented in the field.

Judith Gal-Ezer

Open University of Israel
Judith is a professor of CS. She served as head of the Mathematics and CS Department, and now as Advisor to the President on Gender Issues.

Jeremy Kubica

Google
Jeremy is an engineer working on machine learning. He is the author of *Computational Fairy Tales* and several introductory CS books.

Dave Reed

Chair, CSTA Board of Directors
David is the Director of CS and Informatics at Creighton University in Omaha, NE.

Deborah Seehorn

CSTA
Deborah is Co-Chair of the CSTA K-12 CS Standards Revision Task Force and is currently serving as chair of the CSTA Curriculum and Certification Committee.

Scott Sigman

Drury University, MO
Scott is a professor of CS and the Past-president of CCSC.

Cameron Wilson

Code.org
Cameron was previously Director of Public Policy for the ACM. At Code.org he works in government affairs and policy work.

online, interactive demonstrations) have been updated, and new ones added, along with new videos introducing the chapters. The list of topics covered has been expanded, although a project over the next few months will bring in new sections and chapters covering more areas of CS.

At the same time, the CS Unplugged material (csunplugged.org) will be revamped to be more suitable for classroom use. It was originally written for outreach and classroom extension, at a time when we only dreamed of having these topics in the curriculum.

The material will be re-written and extended to match classroom use. In particular, it will be presented as lesson plans that are ready to use for teachers and have more supporting video and summary material to help teachers quickly understand how to use it in their classroom. It will also link the activities to programming exercises that can allow “plugged-in” follow up.

The sites will be updated with new material as it becomes available. Each site has a mailing list that you can join to receive announcements as major updates occur.

COMPUTATIONAL THINKING VIDEO SERIES

- **What is computational thinking?**
- **How is computational thinking distinct from other thinking skills?**
- **How can teachers assess computational thinking skills?**

Have you ever wanted to ask an expert these questions?

The CSTA Computational Thinking Task Force is building a series of video interviews in which we do just that!

Listen in on our conversation with Chris Stephenson, Director of Computer Science Education Programs at Google, as she answers our questions and describes cross-curricular computational thinking applications in the task of preserving native languages (youtu.be/FuN6g8NmuHc).

Listen to our conversation with Eric Snow, Education Researcher in the Center for Technology in Learning at SRI International as he answers our

questions and describes his research in assessing computational thinking (youtu.be/92pv8dPltjE).

Don't miss our discussion with Jeannette Wing, Corporate Vice President at Microsoft Research, and early computational thinking visionaries, about how research into computational thinking will change everything in education (youtu.be/fSoknljUI4Q).

We visited with John Woollard, Senior Teaching Fellow at Southampton Education School at the University of Southampton, UK, and instrumental in creating Computing At School (CAS), as he described successful computational thinking teacher training (www.youtube.com/watch?v=o_q38yCQUEs).

And recently, we discussed computational thinking with Carolyn Sykora and learned about the revised ISTE standards that include CT learning (www.youtube.com/watch?v=PfUjsa5BlvM).

We have planned several more interviews with experts in the field for later this fall. See the list at: www.csteachers.org/page/CompThinkInterviews

CONGRATULATIONS, CSTA MEMBER

Gene Luen Yang

received a MacArthur Fellowship for his work as a graphic novelist and cartoonist. He is the author of a series of comics, “Secret Coders,” designed to engage kids in computer science through mystery stories.

www.macfound.org/fellows/975/

PROFESSIONAL DEVELOPMENT

WHOIS CCSC?

Scott Sigman

Like the Internet command “whois,” my task in this article is to give you an introduction to the Consortium for Computing Sciences in Colleges (CCSC). Unlike the whois command, which returns a list of information, my goal is to give you more than just facts; I want to give you a feel for the spirit of CCSC and its members.

Let me start with a story. In 1983, having completed a master’s degree in mathematics, I was hired by a small university. I thought my job was going to be to teach math courses and an occasional programming course. However, much like what is happening today with a rapid growth in teaching computer science (CS) topics across K–12 classes, the mid-1980s was a time of rapid growth of CS programs in small universities.

You can see where I am going with this story. Like many of you today, in 1983 I found myself teaching CS classes that I was ill-prepared to teach. I needed help knowing what to teach, how to teach, and what to expect of my students. In the mid-1980s, while I was taking classes and studying on my own (you know the routine), I discovered the conferences sponsored by CCSC. At these conferences I found a community of folks who, like me, were interested in doing a better job teaching and who were willing to share what they were doing and learning.

In a nutshell, my story encapsulates the mission of CCSC. We are a professional organization of educators from the computing sciences who teach at the collegiate level. Our mission is to support those teaching the computing sciences at colleges and universities across the US. We share a passion for improving the quality of our classes and of improving education in the computing sciences.

To achieve our mission, we are organized into 10

regions: Central Plains, Eastern, Midsouth, Midwest, Northeastern, Northwestern, Rocky Mountain, South Central, Southeastern, and Southwestern. Each region sponsors a regional conference. In addition, we publish six issues of the Journal of Computing Sciences in Colleges each year. The Journal provides a means of disseminating the work of our members and is freely available through the ACM Digital Library (dl.acm.org).

The CCSC conferences are organized regionally and each has its own regional flavor. They do, however, share common characteristics. Conferences are scheduled on a Friday afternoon and Saturday morning. Many offer a workshop on Friday morning and sponsor student activities on Saturday afternoon. A highlight of each conference is the opportunity for attendees to network and share with other computing teachers. Keynote and banquet speakers from local and national businesses, government agencies, and academia are featured. Many conferences sponsor activities for K–12 teachers. The dates and locations for upcoming conferences are listed in the “Save the Date” calendar in this issue of the Voice. More information on this year’s conferences can be found on the CCSC web portal at www.ccsc.org/regions by following the link to the region’s website.

I view CCSC and CSTA as sister organizations. While we address different audiences, we share the mission of supporting our members in their professional development and improving education in the computing sciences. My invitation for CSTA members is to look for K–12 activities at a CCSC conference near you. If the region nearest to you does not offer K–12 activities, contact the Regional Representative or Conference Chair and volunteer to help develop some. Let’s find new ways to cooperate.

IMPORTANT: To receive CSTA announcements via email, you must opt in.

If you’re not receiving announcements from CSTA and don’t want to miss out on news about member benefits, visit:

visitor.r20.constantcontact.com/manage/optin?v=001-LUVny3yzzWvmrqvipeuG3ARDVtRef2z

CLASSROOM TOOLS

COMPUTATIONAL FAIRY TALES

Jeremy Kubica

Editor's Note: This interview with Jeremy Kubica, author of over 70 Computational Fairy Tales, explores the tales and how to use them in the classroom. Most of the tales focus on the computer science (CS) concepts and their application in a non-computer domain. Topics cover a range of concepts from introductory programming to advanced CS concepts, from data structures and algorithms to computational complexity, as well as programming tips.

CSTA: What are *Computational Fairy Tales*?

KUBICA: *Computation Fairy Tales* are humorous, and often absurd, short stories that help explain computational concepts through analogy. Each story centers around a single concept with the characters often applying that concept to the physical world. Arrays and binary search trees become physical objects to be traversed. Sorting algorithms appear in everyday situations, such as running an efficient tailoring shop or waiting for tickets to a sporting event.

CSTA: What motivated you to write these Fairy Tales?

KUBICA: I wanted to try to explain computational concepts in a way that captured the same excitement that I feel when I really understand an algorithm. There are many great educational resources out there that formally explain the concepts, but so many of them dive right into detailed math, technical descriptions, and dry examples. You often have to wade through pages of formalities and fully understand the algorithm before you see why it is interesting. With *Computational Fairy Tales*, I wanted to jump to the end. I wanted to start by saying “This is why this concept is so amazing!” The stories themselves are often comical, bordering on slightly absurd, to both provide a light, amusing introduction and highlight how these concepts can be viewed in different ways.

CSTA: Who is the intended audience?

KUBICA: The stories were written for people just starting to learn CS; specifically I wrote them for students in high school or early college courses. However, I have also received positive feedback from parents of younger

students, all the way down to late elementary school.

CSTA: Do you have a favorite?

KUBICA: My favorite story is the first one that I wrote—Caching and the Librarian of Alexandria. The story follows the first day of work for an apprentice librarian who has spent his life hoping to learn the magic behind the librarian’s ability to find the canonical answer to most patrons’ questions with surprising speed. When he finally asks the head librarian about this amazing power, he learns that the librarians rely on a simple, but highly effective caching scheme for the scrolls. While the apprentice is crushed to learn that there isn’t any real magic involved, the story illustrates how simple computational concepts can lead to dramatic real-world improvements.

CSTA: What have you heard from readers?

KUBICA: I’ve received some really nice feedback from teachers and students who have told me that the stories have helped illustrate the concepts. Classes have included links to the stories as part of pre-class reading assignments or to accompany homework assignments.

CSTA: Have there been any surprises?

KUBICA: I was pleasantly surprised by how well the stories have been received in Great Britain. They seem to have resonated well with a lot of educators there.

CSTA: Tell us about your books.

KUBICA: I have three books that set out to explain a different aspect of CS.

The first book, *Computational Fairy Tales*, is a collection of interlinked short stories that provides a broad overview of CS concepts—from variables and loops to advanced algorithms and run-time analysis. The majority of the stories follow Princess Ann as she travels on a quest to stop an unknown computational darkness that threatens the kingdom.

The second book, *Best Practices of Spell Design*, is a single story in which each chapter illustrates a different best

practice of software development. The story follows a senior wizard, Marcus, and his apprentice, Shelly, as they attempt to undo the disastrous consequences of sloppy spellcasting. The book covers many of the practical lessons that I wish I had learned in class, but actually learned through painful (and often repeated) mistakes in the real world.

The third book, *CS Detective: An Algorithmic Tale of Crime, Conspiracy, and Computation*, was published this last summer. The story follows Frank Runtime, private investigator, disgraced former detective, and search expert, as he

investigates the theft of documents from the Capital police station. The book covers a range of search algorithms, data structures, and their real-world applications. This book also offers a deeper technical explanation of concepts than is found in the other stories. It's intended for people who are interested in learning about the algorithms.

LEARN MORE:

computationaltales.blogspot.com

MEMBER RECOGNITION THE PAEMST RECIPIENTS

The most recent recipients of the National Science Foundation (NSF) Presidential Awards for Excellence in Mathematics and Science Teaching (PAEMST) have been announced and there are five CSTA members among the illustrious group of educators! We had a chance to ask them a couple questions and we want to share their responses. Learn more about these exceptional teachers at: bit.ly/2bBGPcq.

How has CSTA added value to your CS teaching career?

CSTA has been a tremendous support in my journey as a computer science (CS) educator. The CSTA resources and community have provided me with ideas, direction, and research-based findings from which to build curriculum that engages my students and inspires them to question, observe, analyze, and reflect.

Janice Mak, Fireside Elementary School, Phoenix, AZ

CSTA has been key to my teaching career! The close network of friends and colleagues I've met through CSTA keeps me going through good times and bad. My career would never have been as rich without the people of CSTA!

Daniel Moix, Bryant High School, Bryant, AR

What inspired you to teach CS?

I had a 15-year career as an Information Technology specialist in software development and publishing businesses. My job was always interesting and challenging. I try to bring my diverse experiences to the classroom to inspire my

students for the infinite possibilities that are available in CS. *Neil Plotnick, Everett High School, Everett, MA*

I didn't set out to be a CS teacher. I got a degree in CS and found that the tutoring and collaborative parts of my college experience were more fulfilling than time spent working in a "cube." When there was a need for a CS teacher at my high school, I stepped in as a long-term substitute and never looked back.

Daniel Moix, Bryant High School, Bryant, AR

How do you stay motivated in the classroom?

My motivation goes into creating the kind of lessons my students will emotionally and academically respond to in my lessons. After the lesson is planned, I don't have to do anything to motivate myself. My students do all the motivating by asking the right questions, discussing possible solutions, debugging their code, and expressing excitement when their algorithm works.

Melanie Wiscount, Ron Brown College Preparatory High School, DC

Every day is full of surprises with my fourth graders. My STEM projects and their daily creativity are what keep me motivated. I love to transport the classroom to real-world challenges.

Esther Alvarez-Meléndez, Academia San Ignacio de Loyola, San Juan, PR

REFLECTING ON CS EDUCATION

TEN TIMELESS PRINCIPLES OF CS EDUCATION

Judith Gal-Ezer

In recent years, numerous publications in the computer science (CS) education professional literature began with the sentence: “CS is a young discipline and CS education is even younger.” Well, time has passed and CS education grew up.

It might not be fully mature, but knowledge and experience have been accumulated, the community has grown, and the world realizes that CS deserves special attention. In many countries, policy makers understand that time and money should be devoted in order to implement goals, such as “computing for all.”

Israel has a long history of formalized CS education. A high school CS curriculum has been taught for more than 20 years. During the past four or five years, Israel began implementing a computing curriculum in middle school (grades 7–9), and beginning next school

and looking forward to the future, these are my beliefs about their relevancy for CS education today.

- **CS is a full-fledged scientific discipline.**
The 2020 *SCIENCE* report published by Microsoft formalizes this: “Indeed, we believe CS is poised to become as fundamental to biology as mathematics has become to physics. [...] However, what this report uncovers, for the first time, is a fundamentally important shift from computers supporting scientists to ‘do’ traditional science to CS becoming embedded into the very fabric of science and how science is done, creating what we are prepared to go so far as to call ‘new kinds’ of science.”

Computing or CS, (or informatics, as titled in Europe), should be on the same par as the other sciences, such as physics, biology, or chemistry. In

Indeed, we believe CS is poised to become as fundamental to biology as mathematics has become to physics.

year, we will pilot a program in grades 4–7 of elementary school.

The common understanding now is that pupils should be exposed to computing in a relatively early stage of their studies, but the question of “how young is too young?” (which was the title of WIPSC2016) is still in the air. In spite of much progress, there is not enough research to answer questions about when and what should be taught, and who should teach it. While it is natural that in high school CS should be an autonomous discipline, it is not trivial to ask if this should be the situation in elementary schools where often one teacher teaches a wide variety of subjects.

The following principles are based upon the guidelines for Israeli curriculum designers in the 1995 document, *A High-School Program in Computer Science* (www.openu.ac.il/personal_sites/download/galezer/high-school-program.pdf). In light of the past twenty years

this sense, if these sciences are accredited as autonomous disciplines, so should CS be.

- **The program should concentrate on the key concepts and foundations of the field.**
This principle is still relevant and a similar sentence should appear in every curriculum, program, or standards document.
- **Each program should have required and elective chapters.**
I still think it is crucial that teachers teach topics in which they are most confident.
- **Conceptual and experimental issues should be interwoven throughout the program.**
This is a very important guideline. Some popular initiatives tend to concentrate on coding rather than exposing students to the more conceptual or abstract facets of the discipline. But, concentrating only on algorithmic or computational thinking

- without some “hands on” work is also wrong.
- **Ideally, two quite different programming paradigms should be taught.**
Two paradigms can help students gain more insight into the discipline. Different paradigms might include different ways of thinking, different ways of solving problems, or different programming languages, such as procedural and logical (Pascal and Prolog) or object-oriented (Java).
 - **A well-equipped and well-maintained computer laboratory is mandatory.**
Of course, it is important to have computers so that “hands-on” activities are possible. However, the computers do not have to be in a dedicated room or lab. In some countries and regions, students have personal computers to use throughout the school. “Hands-on” activities are a must, even if the program starts with an “unplugged” program.
 - **Course material are essential for a program.**
It is very important that there be both quality teaching and learning materials.
 - **Teachers certified to teach the subject must have adequate formal CS education.**
Teachers are the key to the success of implementing any CS program. It is necessary that CS teachers have formal education in the discipline, along with teacher preparation in the pedagogy and methodology for teaching CS. In-service training and professional development are also important elements in ensuring teachers are adequately prepared.
- Even though the above principles were written more than a decade ago, I believe they are still relevant and even crucial for ensuring quality CS education at all levels.

TEACHING ABOUT CS CAREERS

CAREERS WITH CODE: A NEW CS CAREERS MAGAZINE

Abby Bouchon Daniels

Computer science (CS) education provides a way of thinking that focuses on creative problem solving and teamwork (important skills for any career) and is a powerful way to express one’s self. And from the programmers behind Pokemon Go to the creators of chatbots, the impact of CS is ubiquitous in society.

With a projected 1 million jobs going unfulfilled in computing-related roles by 2020, we need computer scientists from all backgrounds to bring their unique perspectives to solve real-world problems. CSTA plays a critical role in preparing students for these careers of tomorrow by helping students connect the dots between their skills and passions.

To illustrate the many careers that intersect with CS, Google and Refraction Media have partnered to create a free “CS + X” career magazine called *Careers with Code* (careerswithcode.com). *Careers with Code* shows students how to combine their interests (their “X”) with CS and how these intersections are creating game-changing careers. Readers can get to know people, such as Jonathan Graham, who use CS in their daily work in sometimes unexpected ways.

A lifelong music fan, Jonathan learned to code to mix live music on stage. One summer while visiting family in Pennsylvania, he was struck by the number of coal mines closing down in the region. Jonathan decided to put his CS skills to work by providing skill-based learning for laid-off coal miners, helping them explore new technical career opportunities. He is now the co-founder of the nonprofit Mined Minds Foundation (www.minedminds.org), which aims to spur economic development by seeding technology hubs in the coal towns in Pennsylvania and West Virginia.

If you’d like to incorporate Jonathan’s story and other *Careers with Code* articles into your student outreach but don’t know where to start, check out *Careers with Code*’s free user manual (careerswithcode.com/classrooms). CSTA members and students are also encouraged to also share feedback (careerswithcode.com/feedback). Additional free magazine resources include:

- Articles, including “9 Ways Computer Science Is Making the World A Better Place” (issuu.com/refractionmedia/docs/careers_with_code_us/27).

- A directory of CS resources (issuu.com/refractionmedia/docs/careers_with_code_us/42).
- A step-by-step checklist for mapping next steps in CS (issuu.com/refractionmedia/docs/careers_with_code_us/40).
- Option to order hard copies (careerswithcode.com/magazine-order).

For CSEdWeek 2016, *Careers with Code* will have

a guided “unplugged” CS career exploration activity available. Please visit the *Careers with Code* website (www.careerswithcode.com) to access this activity starting in December!

We hope that *Careers with Code* will inspire students to use technology to create meaningful things for their communities and equip CSTA educators to celebrate and support them along the way.

SPONSOR NEWS

MEETING NETWORK NEEDS OF THE PASCAGOULA DISTRICT

Toby Bradley

Editor’s Note: iboss (www.iboss.com) was a sponsor of the 2016 CSTA Annual Conference in San Diego.

The Pascagoula School District (www.pgsd.ms), located in the Gulf Coast Region of Mississippi, is home to 19 K–12 schools that serve 9,000 students and staff at 21 sites across the district region. As Network Supervisor, I’m responsible for managing the network and ensuring security, while supporting a diverse mixture of Windows and Mac computers and iOS and Chromebook tablets. Like all districts, we need to maintain CIPA compliance by safeguarding access to the Internet so their schools could qualify for valuable E-Rate funds.

We had been using a web-filtering solution but I didn’t feel we were getting the security and functionality I felt we needed. I think there are other districts that may feel the same way. Years ago, when CIPA was first passed, it was enough to have a web filter that managed Internet access and was reasonably efficient at keeping inappropriate content off the desktops in the student computer labs. All CIPA required was to implement technology that could safeguard Internet access—it didn’t specify which technology. Like a lot of schools, we chose a pretty good web filter that was priced right and had the features we needed at the time.

But things have really changed. We are keeping more personal data on the school network. One-to-one initiatives mean that we have to enforce use

policies and CIPA whether devices are at school or at home. And with the growth of BYOD, online testing, and the many learning resources on Google and YouTube that teachers want to access, the security picture has changed dramatically and the security approaches we were using are no longer enough. That’s where we found ourselves at Pascagoula and why we decided that, rather than renew our existing security solutions, we would explore what else was available. We’re very glad we did.

The standard solution we were using was showing evidence of not being up to this task. We were disappointed that, although the company had recently pushed out an upgrade, it actually deprecated some of the features we liked best.

The solution we chose is the iboss Secure Web Gateway Platform. It has the advanced threat defense features we need and is also able to address other areas that are most critical for us:

- We were able to get visibility across all 131K ports on the network, giving us more insight into our network traffic so we can better handle the multiple platforms and mobile devices we are supporting. The responsive user interface and dashboards have been a valuable tool for knowing exactly what’s happening across the network.
- iboss BYOD management allowed us to give teachers and select classes network access while making certain all users are encompassed in their

security policies. By using iboss in the MAC Gateway mode, we are able to track all non-Active Directory devices on the network by their MAC address. This means we can tag allowed devices such as computer workstations, while non-allowed devices receive only guest privileges.

- We found multiple benefits in the reporting capabilities, including being able to schedule management reports, easily drill down from district-wide views to individual users, and track events and threats in real-time with live dashboards, all accessed from a central management console.
- iboss bandwidth optimization and controls have been invaluable, giving us a very important capability because when a major OS update is pushed out during state testing, which has happened, we've been able to count on the iboss solution to ensure

critical processes continue unimpeded.

- The responsive Technical Support team has proven far superior to our experiences with our previous vendor. From the beginning, we had concierge access to an experienced security engineer, which was instrumental in streamlining the switch from our previous vendor. iboss has stayed responsive—when issues arise, they are promptly addressed.

What I would tell other schools districts who are assessing or reassessing their network and data security strategies, is to make a list of the features that would comprise the ideal cybersecurity solution and compare it with what you have now. You may be surprised to discover that you can get more functionality than you thought, without breaking your budget.

MARK YOUR CALENDAR

CCSC (Eastern)

K-12 Special Registration

October 28–29, 2016, Frostburg, Maryland

www.ccsc-eastern.org

2016 Infosys Foundation USA/ACM/CSTA Awards Application

November 1, 2016, Deadline

www.csteachers.org/default.asp?page=ApplicationProcess

CSTA Annual Conference Call for Proposals

November 4, 2016, Deadline

www.xcdsystem.com/csta/member/index.cfm

CCSC (Southeastern)

November 4–5, 2016, Asheville, North Carolina

www.ccscse.org

Aspirations in Computing High School and Educator Awards

November 7, 2016, Deadline

www.aspirations.org

Cyber Security Awareness Week Conference

November 10–12, 2016, Brooklyn, New York

csaw.engineering.nyu.edu

CSEd Week

December 5–11, 2016, in YOUR community

csedweek.org

SIGCSE

March 8–11, 2017, Seattle, Washington

sigcse2017.sigcse.org

CCSC (Central Plains)

K-12 Events

March 31–April 1, 2017, Lincoln, Nebraska

www.ccsc.org/centralplains

2017 CSTA Annual Conference

July 9–11, 2017, Baltimore, Maryland

cstaconference.org

Check the most recent CSTA events on the CSTA website

www.csteachers.org/events/event_list.asp

List your CSTA event by contacting c.cruceetti@csta-hq.org