



Session Descriptions
Monday, July 11 and
Tuesday, July 12, 2016
Town and Country
San Diego, CA

Monday, July 11, 2016

Concurrent Sessions

2:45 – 3:45 pm

Introducing Bootstrap: 2 and Bootstrap: CSP | *Emmanual Schanzer and Eric Allatta*
Description coming soon!

Working Session on Integrating Computational Thinking Across the K-8 Curriculum | *Pat Phillips and Irene Lee*

In this working session, participants will engage in structured, guided discussions of current (or proposed) lessons and practices incorporating CT. Our primary goal is to encourage professional dialog among teachers to gather ideas, learn about what has worked in the classrooms of their peers, and feel encouraged to incorporate new CT activities in their lessons. Our secondary goal is to build upon what we learn and gather from the participants to create a revised CSTA CT pamphlet for teachers and administrators. The “hands on” working part of the session will run for 40 minutes which will include small group discussions, as well as time for the sharing out to the entire group.

SciGirls Code: A National Connected Learning Model to Integrate Computing in STEM Learning with Middle School Girls | *Karen Peterson, Joan Freese, Cassandra Scharber and Rita Karl*

SciGirls Code: A National Connected Learning Model to Integrate Computing in STEM Learning with Middle School Girls is a two-year pilot program lead by Twin Cities Public Television and funded through the NSF’s STEM + Computing Partnerships Program. SciGirls Code uses the principles of connected learning with STEM outreach partners to provide 160 girls and their leaders with computational thinking and coding skills. We are developing curriculum centering on three tracks—e-textiles and wearable tech, robotics, and mobile geospatial technologies; role model training for female technology professionals; professional development for STEM educators; and a research component that investigates the ways computational learning experiences impact the development of computational thinking as well as interest and attitudes toward computer

science. Partners on SciGirls Code include National Girls Collaborative Project (NGCP) and the University of Minnesota Learning Technologies Media Lab, as well as a diverse array of SciGirls CONNECT partner programs.

Linux and the Open Source Classroom | *Neil Plotnick*

Linux is a mature operating system that provides a complete solution for the CS classroom. This session will examine how to deploy a variety of Linux environments for the desktop and server needs of the classroom. Applications that are used for software development will be given particular attention. Options for Windows and Mac dominated environments will also be explored.

ECP+How a French Teacher in Rural Arkansas Learned to Teach CS | *Daniel Moix*

This session will introduce ASMSA's blended approach to Computer Science teacher mentorship and share the successes and areas for improvement experienced by 16 Arkansas classrooms teaching Essentials of Computer Programming "Plus." The technology infrastructure, the professional development model, the student experiences, and the lessons learned will be shared.

Microsoft CS50 Panel (Sponsor Spotlight) | *David Malan, Natasha Chornesky, Angela Yake and Greg Mittleider*

CS50 is Harvard University's introduction to the intellectual enterprises of computer science and the art of programming for majors and non-majors alike, with or without prior experience. Indeed, 72 percent of the course's students have never taken CS before. Among the course's hallmarks are its focus on accessibility, fostering a supportive and exciting student culture, and the rigor of its subject matter. AP CS Principles ("AP CSP") is a new course that has been in development by the College Board, with support from the National Science Foundation, since 2008, with an official rollout for the 2016–17 school year. AP CSP, unlike the existing AP Computer Science A course ("AP CS A"), endeavors to give students a broader literacy about computers and computing, rather than focusing specifically on programming being used as a problem-solving tool.

Still, with programming savvy so beneficial today, and to facilitate an easier transition to AP CS A for those who take the course, we chose to use programming as the primary vehicle via which to introduce students to the important "Big Ideas" of the AP CSP curriculum. In addition to providing lessons, resources, and assignments for students, we made it a priority to focus on providing a support structure for teachers as well. In 2015–16 we piloted, with support from Microsoft, CS50 for AP CSP, an adaptation designed to be more conducive to secondary school settings, with a cohort of over 50 teachers and approximately 1,000 students at middle and high schools around the world. We present in this panel the results of that pilot. What did we do and why did we do it? What would we do differently now that our first year is nearly complete? How did we create and sustain a support network for teachers? How are students responding to the course, and how do we get even more students involved in computer science using this or similar courses?

CSTA K-12 CS Standards 2.0 | *Deborah Seehorn and Tammy Pirmann*

On December 1, 2011, the CSTA K-12 Computer Science Standards were released. These standards have been instrumental in assisting many schools, school systems, and states in implementing and promoting computer science education in the K-12 classroom. This past fall, CSTA enlisted the assistance of several computer science educators to revise those standards. This session will provide a brief overview of the new and

improved standards and provide an opportunity for discussion among the participants. Come and get a head start on planning for your fall CS classes and programs!

GenCyber Camps – Paving the way to a Cybersecure future | *Tina Ladabouche and TBD*

The GenCyber grant program provides funding for summer cybersecurity camp experiences for students and teachers in grades K-12. Funded by the National Security Agency and National Science Foundation, GenCyber grants have already been used by academic institutions in 36 States and the District of Columbia. Meet the NSA program management team, hear about some of the wildly successful camps from the Camp Program Directors themselves, and consider running a summer camp in *your* district!

The GenCyber program is designed to help students understand correct and safe on-line behavior, increase interest in cybersecurity careers, promote diversity in the future cybersecurity workforce, and improve teaching methods for delivering cybersecurity content. Grantees take the lead in figuring out how to make it all happen. For more information, see www.gen-cyber.com.

Exploring Computer Science Computes for the Social Good | *Nancy Se and John Landa*

In this session we will discuss how Exploring Computer Science teachers can integrate student projects focused on computing for the social good into the ECS year-long course. The two presenters, an ECS teacher and an ECS classroom coach, accompanied by several students (or videos of students talking about their projects), will discuss which topics in the curriculum (including CS content and practices) can link to this theme, and how to assist students with their projects. Presenters will also discuss the student outcomes and display examples of student work. Questions about how to integrate inquiry, equity, and CS content into student CS projects will be addressed.

Concurrent Sessions

4:00 – 5:00 pm

Equity, Diversity and Inclusion in Computer Science in High School: What Works? What Doesn't? | *Susan Yonezawa, Nan Renner, Roman del Rosario, Kellie Fleming and Gail Lake* In this hour-long session, presenters from UC San Diego, and three San Diego area districts -- Sweetwater Union HS District, San Diego Unified, and Vista Unified -- will share insights as to how they approach student recruitment into new Computer Science courses at the middle and high school levels. Presenters will focus on issues of equity, inclusion and diversity and will focus on strategies -- at the district, school, and student level -- that help encourage low-income, minority and female students to enroll and persist in CS courses, sometimes for the first time. University, district, school (administrators and teachers) as well as students will provide insights as to how to improve recruitment practices.

CS Pathways in Gr 5-8: Making Apps for Social Good | *Fred Martin, Molly Laden, Akira Kamiya and Lijun Ni*

CS Pathways is an NSF-supported partnership among the University of Massachusetts Lowell, the Tri-City Technology Education Collaborative, and the urban school districts of Everett and Medford, MA (near Boston). The districts' technology teachers developed

a 20-hour introductory computer science curriculum designed for all middle school students. We introduce students to computing by engaging them in creating their own mobile apps for socially beneficial purposes. Our approach is designed to be broadly welcoming and inclusive to all students; to appeal equally to both girls and boys, and students from diverse backgrounds. The project began with a cohort of five teachers for the 2014-15 academic year, and presently is bringing in a second cohort of five teachers. The session will include details on the teachers' professional development process, the curriculum they developed, connections to state learning standards, reflections on classroom practices, and student learning outcomes.

Bringing CS to Students with Learning Differences | *Sarah Wille, Erica Roberts and Steve Svetlik*

Expanding learning opportunities in computer science is an issue of educational equity for all students, including those with learning disabilities and/or an ADHD. The goal of this session is to arm educators with research-informed approaches for including this student population in the new AP CS Principles course, and CS courses more generally. Presenters will introduce an evidence-based guide that provides strategies for working with students with specific learning disabilities or an ADHD in computer science classrooms. The session will describe the NSF-funded research behind development of these “preliminary guidelines” and their substance, share modifications to a CS Principles lesson informed by the guidelines, and solicit feedback on our project work. Educators will leave with a copy of the preliminary guidelines, an adapted CS Principles lesson, and practical recommendations to increase successful participation of students with learning disabilities or an ADHD in high school computer science classes.

Integrated Computing and Robotics in a Project-Based Math Classroom | *Adam Ko*

The University of California, Davis Center for Integrated Computing and STEM Education (C-STEM) aims to transform K-12 education through the integration of computer programming and robotics into the classroom. The C-STEM Center has developed a coding environment using Ch, a superset of C, to implement project-based computing and robotics activities aligned with Common Core State Standards. The session will cover three areas: (1) using computing as a unique access point for Algebra, Statistical and Geometric concepts for Integrated Math I; (2) integrating robotics to foster problem-solving skills and to apply math concepts; and (3) overcoming challenges in providing a computing and robotic experience that is engaging for every student. The presenter will share his experience implementing the C-STEM curriculum in his 9th grade Math classroom located at High Tech High, a project-based charter school located in San Diego, California.

Unplugged? Or Plugged Into an Idea That Works? | *Stacey Kizer*

Like many teachers, I teach my CS courses during 45-minute periods – and in my Programming courses I’ve chosen to spend the first 10 minutes of that brief period every day on “unplugged activities”. This session will explore the impact that decision has made on my students’ motivation and ability to approach problems analytically, write algorithms, and ultimately: code. We’ll look at what classroom data shows – from two high schools in my own experience and the research from others –regarding whether improving their computational thinking ability improved students’ efficiency and motivation while in a CS class. You’ll see actual resources, ideas, strategies, along with samples of student work, and develop your own. Links to the puzzles, challenges,

creative mathematical problems, mazes I've created that work in high school classrooms will be provided for you to use and modify.

How to Ramp Up a CS Program With Blended Learning Tools (Sponsor Spotlight) |

Saber Khan and Jeremy Joven

At the Browning School, an independent K-12 school, and Success Academy Charter Schools, both in New York City, we have built a comprehensive technology programs that combines computer science and design. We have identified the resources and trained our teachers. Our program is possible through the extensive use of online blended learning tools. Join us for a session to hear about our programs and get ideas for your program.

Applied Math With Computational Thinking | *Pierre Bierre, Gregory Duran, Lee Applebaum and Joe Pistone*

Math learning can take on greater engagement and 21st century relevance as students apply theory to challenges where they author software combining numerics and graphics. This panel session will explore 3 case-studies in Math taught from a hands-on CT perspective: • AP Calculus BC • Advanced 3D Geometry • Introductory Number Theory Panelists will discuss their experiences as course designers and practitioners, and relate students' perceptions of the impact of Math with programming. Ideas on how to get started teaching Math + CT will be offered.

NGCP session: TBD

Description coming soon!

Birds of a Feather

5:15 – 6:00 pm

Expanding Participation in Coding and Computer Science in Arkansas | *Anthony Owen and Carl Frank*

This presentation would focus on the history, current status, and future vision of computer science in Arkansas. During the section on the history, Governor Hutchinson's computer science campaign promise and subsequent support will be discussed along with information on how Arkansas has moved the needle so rapidly, including best practices and lessons learned. Current status discussions will revolve around the Arkansas Department of Education's K-12 curriculum development work and how cultivating partnerships with local industry and national entities such as CSTA has been of great mutual benefit. The discussion regarding the future vision of computer science in Arkansas will focus on the state's short-term and long-term plans to remain a national leader in this area.

Building and Revitalizing a Local CSTA Chapter | *Pierre Bierre and Fran Trees*

This Birds-of-a-Feather will bring together experienced Local CSTA Chapter Leaders with new leaders and those considering starting a Chapter. The discussion will include strengthening (rebooting) an existing chapter as well. Topics will include: process for starting a chapter strategies for recruiting members meeting location & time choices (commute factors) planning engaging topics and activities online web presence and communications putting on PD workshops local advocacy & outreach obtaining sponsors finance & legal concerns leadership development / smooth leadership succession.

Computer Science Learning Centers in Elementary Classes | *Melissa Sanchez*

How do you effectively integrate computer coding opportunities, problem solving activities, technology applications, STEAM challenges, MakerSpace projects and curriculum connections into the elementary classroom? Through Computer Science learning centers, of course! Design and implement standards-based centers in your elementary classroom and provide meaningful technology integration and computational thinking experiences for students at all levels. From computer programming with Scratch, Turtle Art and LEGO Robotics to block-based languages on iPads and unplugged activities, you CAN integrate Computer Science into the elementary curriculum through learning centers in the classroom. In this session attendees will receive information about Computer Science based learning center materials, management and implementation along with various methods for documenting and tracking student learning.

Computer Science as Part of Universal STEM in Middle School | *Elizabeth Bacon, Meaghan McLoughlin and Ryan Grady*

Universal computer science at the elementary and secondary level has become increasingly popular. This presentation will discuss the challenges and benefits of implementing universal computer science in middle school. We will discuss the impetus, planning, implementation, and impact of one school's middle school computer science program while opening the discussion to those who are considering, planning, or implementing their own programs.

Culturally-Responsive CS Teaching Practices | *Sarah Judd*

How can people from over-represented populations create a classroom environment that encourages students from under-represented backgrounds to see themselves as programmers? Sarah, the presenter, has dealt with this problem directly as a white teacher in urban school districts, and through Girls Who Code, which hires teachers of any gender to encourage girls to study Computer Science. Using what she's seen work in her own classrooms as well as those at Girls Who Code as a starting point, this session will discuss how you can build an identity-affirming classroom for students with different identities than your own. You'll leave with resources and ideas for concrete techniques you can use in your own classrooms to be a better ally for your students.

Tuesday, July 12, 2016

Opening Keynote

8:30–9:45 am

Computer Science for All: How Business Must Be Engaged | *Moderator, Gary Beach (Publisher Emeritus, CIO Magazine) and CIO panelists: Cora Carmody, former CIO Jacobs Engineering and founder of Technology Goddesses; James Nanton, CIO, UST Global; and Sabrina Lubinsky, CIO, Oakwood Worldwide*

If computer science for all is ever to become a reality in the United States, business leaders must be fully engaged. But what is the business value of computer science proficiency in the workplace? Is it a “nice to have” or a “must have” skill for firms to effectively compete? A panel of chief information officers will share their opinions on

those topics and others; like, does the term computer science need a brand makeover, what must be done to make the discipline of computer science more diverse and should the United States follow the example of the United Kingdom and mandate that computer science be taught to every young person from age five to 16? If that happened, would business leaders willingly lend skilled employees help teach code, programming and computer science in our nation's classrooms?

Concurrent Sessions

10:00 – 11:00 am

CS Matters in Maryland: A Collaborative CS Principles Course | *Marie desJardins, Jan Plane, Megeab Garvin, Dianne O'Grady-Cunniff, Joe Greenwalt and Christina Morris*

The NSF-funded "CS Matters in Maryland" project used a collaborative master teacher approach to develop a complete curriculum for AP™ CS Principles that emphasizes student collaboration, inquiry-directed learning, active student engagement, and differentiation to meet diverse students' needs. The theme of the course is data: how we collect, analyze, visualize, and use data in our networked world to solve problems and make informed decisions. In this session, we will describe the CS Principles framework, the CS Matters curriculum, our collaborative development process, and our professional development approach for new teachers. The session will include a hands-on period to demonstrate a typical lesson in the course. We will also leave time for a question-and-answer session about our approach to CS Principles, how we have been working to deploy the curriculum to all high schools in Maryland, and our statewide efforts to improve K12 CS education.

Teaching Strategies for K-8 Computer Science | Sheena Vaidyanathan

Learn teaching strategies and best practices to meet the learning needs of every student in your K-8 computer science classroom. The session will use sample lesson plans and student samples from a district wide computer science class for 6th graders to illustrate key 'how to teach' issues in a computer science classroom: balancing structured and unstructured class time, using both original and remix projects, giving student choice to encourage creativity, managing collaboration and pair programming and most importantly differentiation to reach all types of students including girls, english language learners, struggling students, and advanced students. Strategies are based on 5+ years of experience teaching Scratch and other languages to over 500 students each week in a public district in California.

Making K-12 Computer Science Accessible to All Students | *Andreas Stefik and Richard Ladner*

This session will focus on efforts to make K-12 computer science accessible to students with disabilities. There will be a short introduction to the NSF-funded AccessCS10K project that has the goal of AccessCS10K is to increase the successful participation of students with disabilities in K-12 computing education. The main focus of this session will be an introduction to the Quorum programming language that is an easy to learn text-based language that is fully accessible. Topics that will be discussed are the Quorum Hour of Code, Quorum-based curricular alternatives for block-based languages, the Lego Robotics support in Quorum, and the Quorum game engine. Participants in the session will have an opportunity to write short programs in Quorum. They will learn about

curricula and strategies to include students with disabilities in their classrooms. They will have an opportunity to write short programs in Quorum.

CSTA Source: CS Teaching Repository REBORN! | *Debbie Carter and Joe Kmoch*

There have been many excellent instructional developed by CS educators, but finding them can be difficult, especially for teachers who haven't yet formed connections in the CS education community. Many K-12 computer science educators teach in isolation, and changing technology demands that our skills and resources be updated frequently. The CSTA Source: A Web Repository of K-12 Computer Science Teaching and Learning Materials, was developed to house resources that support and are explicitly mapped to the CSTA's K-12 CS Standards (2011). The 200+ resources include projects, lesson plans, presentations, and program strategies, contributed by educators and organizations. Recent enhancements include porting to a Drupal-based platform and multiple browsing hierarchies, in addition to a full-catalog search by keyword. This session will introduce the repository to educators, or acquaint them with its new features, and encourage both the use and the submission of resources.

Your Foundation – Resources to Develop Java Skills | *Robert Martin*

Over the last four years, the number of students taking the AP Computer Science A (CSA) Exam has increased 120%. To support teachers as schools expand their AP CSA course offerings, the AP CSA Development Committee is providing this session for new AP CSA teachers. AP CSA students are expected to fluently write object-oriented solutions using the Java programming language, utilizing standard Java library classes and interfaces from the AP Java subset. To help students prepare for success in the AP CSA Exam, AP CSA teachers must know how to use Java to design, implement, and analyze solutions to problems.

TEALS – Teaching CS Through Tech Industry Partnerships | *Kasey Champion and Paul Mulvaney*

TEALS (Technology Education and Literacy in Schools) helps high schools with little to no computer science presence build sustainable programs by pairing trained computer science professionals with classroom teachers to team-teach. By bringing in professionals from across the tech industry TEALS volunteers and partner teachers create a ripple effect, impacting the students they teach, and the many students who will study CS in the future. TEALS works in 20 states, in 300 schools and impacts over 10,000 students. Come and learn from a TEALS classroom teacher how you can bring CS to your schools with a TEALS.

Teach a Cybersecurity Course at Your School | *Melanie Wiscount*

NOTHING is secure! Hackers, malware, advanced attacks, security breaches are common words in the news, and not only are countries, governments, militaries, commercial businesses, and non-profits the victims but each of us are prey if we spend any time online. Prepare your students to be cybersecurity specialists. Teach lessons about password security, cryptography, cloud security, privacy, risk management, data loss prevention, and cyber espionage, as well as security measures against malware, advanced attacks, and social engineering. Help students develop practices of responsible, ethical, civil, and accountable behavior online. At the end of this course, they will be a great resource for their families, their teachers, their employers, and their peers.

Administrator Session

Description coming soon!

A Different Approach to Assessment – AP CSP Performance Tasks | *Lien Diaz and Andrew Kuemmel*

Description coming soon!

Concurrent Sessions

11:15 am – 12:15 pm

Teaching CSP Online for the First Time | *Ruth Mass*

Ever thought of teaching an online class? Learn what worked, and what didn't work from a novice about how to successfully teach Computer Science Principles online.

Teaching and Learning Through Creating Games in ScratchJr | *Aye Thuzar and Aung Nay*

This session is based on our Blocks and Beyond Lessons and Directions for First Programming Environments A VL/HCC 2015 paper, Teaching and Learning through Creating Games in ScratchJr: Who needs variables anyway! [1], in which we developed lessons for games in ScratchJr beyond what is covered in the ScratchJr curricula from scratchjr.org. The idea that we developed for ScratchJr is a sprite moving towards a goal as a visual progress tracker/indicator of a player's progress without the use of variables, which are not available in ScratchJr. This concept used in game creation will allow teachers to move beyond the interactive storytelling phase and maintain excitement with students when ScratchJr is used in extended sets of lessons. This session will introduce attendees the concept of making games in ScratchJr without the use of variables. It will be followed by a discussion. (BYOD iPad/Android Tablet).

An Education and Industry Partnership to Grow Computer Science in New Zealand

| Margot Phillipps and Ewen Bell

A program in New Zealand which worked outside of the normal bounds of the “state-managed” education system to offer students the chance to gain credit in the national qualifications system in Computer Science and at the same time train and empower teachers to continue offering courses. It was in response to the slow uptake of formal CS courses by high schools once the standards became available. The intended audience is High school teachers with a passion for growing Computer Science and sympathetic industry staff. The session will present the program and its results, including video of interviews with teachers trained by the program and an open discussion time.

Interdisciplinary CS in Middle School | *Karine Laidley and Angelica Gunderson*

Interested in bringing computer science into your middle school but unsure how to accommodate it? Attend this one-hour session led by Project Lead The Way to take a closer look at ways to include computer science in an engaging, problem-based approach, which enables students to recognize the connection between computer science and other disciplines. Gain new insight into how computational thinking concepts can be used to solve authentic problems and enhance creativity and innovation. Attendees of this session will be introduced to various interdisciplinary methods to teach computer science in middle school along with other subjects, including but not limited to science, math, and

engineering. Alignment to middle school standards and learning objectives will be provided, and sample projects will be demonstrated to illustrate how lessons can be structured for problem-based learning. Time will be reserved for Q&A at the end of the session.

Microsoft Sponsor Spotlight

Details coming soon!

Pure Data: Visual Programming for Performance | *Jenks Whittenburg*

Using visual programming for audio and video manipulation can bring excitement and creativity to core programming concepts.

Overcoming the Obstacles for CS in Education | *Andrew Svehaug and David Miyashiro*

Looking for ways to convince your school and district administrators to expand computer science opportunities for your students? Come hear from one of the leading Superintendents for CS Education speak on how the Cajon Valley USD was able to scale CS district-wide and overcome the systematic challenges present in every district. We will discuss Rios Elementary CS Magnet School (America's first K-5 CS Magnet) as well as other projects the district is pioneering to provide examples and inspiration for expanding your CS program.

Mini sessions

Making Lemonade When You Have No Lemons | *Eve Sarra*

If you are faced with teaching Computer Science with little or no budget, this mini-session will give you some solid ideas of where to start, what to do, and will provide you with some valuable resources that will get your Computer Science program off the ground and running smoothly. Eve Sarra, an inner-city high school teacher, will outline what she did when her school was without a Computer Science program. Firmly believing that Computer Science classes could give many of "her" kids the skills they needed to pursue a successful career, she fought to make Computer Science a part of the curriculum and pledged to do whatever it took to get courses up-and-running. You will be inspired by how she leveraged free resources such as Code.org, Khan Academy, Scratch and CodeHS into a successful Computer Science program!

Elementary STEM Problem-Based Learning + Computing | *Lisa Milenkovic*

This session will present a transdisciplinary model for integration of STEM with computing in the elementary day. The presented model introduces design units that encompass a scaffolded series of performance tasks integrating science and computer science with literacy. By contextualizing the content first, we engage the students in solving authentic STEM problems in the elementary classroom.

Title TBD | *Pat Youngpradit*

Description coming soon!

Concurrent Sessions

1:15–2:15 pm

Advanced Placement Computer Science Principles (APCSP): A Report from Teachers | *R Brook Osborne and Owen Astrachan*

As we approach the launch of the APCSP exam in the 2016-17 school year, many teachers are deciding which curriculum they will use once they start teaching the course. Come to this session to hear from teachers delivering courses written by Project Lead the Way (PLTW), Code.org, Beauty and Joy of Computing (BJC), Mobile CSP, and Thriving in our Digital World. Learn more about what each of these programs has to offer, from professional development to curricular resources and tools directly from classroom teachers who are working with these materials. The session will provide an overview of what each program offers, insights on the experiences of practitioners of each program, and time for open discussion and questions with the community.

Hummingbird Robotics: Programming Options for Ages 10 to 110 | *Bambi Brewer*

Students with different levels of coding experience can use the Hummingbird robotics kit to create and program their own robot. Beginners can use drag-and-drop programming to control graphics and robotic elements. More advanced students can use Ardublock and Arduino language to transition from drag-and-drop programming to a text-based language. Finally, students who are ready for a more powerful language can use Python or Java to program the Hummingbird. This presentation will explore the basics of each platform, as well as examples of cross-curricular projects.

Mastering the Brain Break: Using Improvisation to Create a Culture of Trust and Collaboration in the CS Classroom | *Daniel Fenjves*

Learning Computer Science can be a challenging and oft-frustrating endeavor for students in the K-12 classroom. Recent research shows that taking short breaks every 30-40 minutes improves behavioral and neurocognitive performance – resulting in increased productivity and student focus. In this experimental, hands-on session participants will learn a variety of improv-based ‘brain break’ activities to get students out of their seats while developing a culture of trust and collaboration with each other. The activities to be presented have been used by over 40 teachers and 800 students at the Flatiron School’s Pre-College software development courses.

Beat Senioritis with a Real-Time Networked Space Sim | *Brett Wortzman and Michael Hawker*

Is engaging students at the end of the school year a challenge for you? If so, come learn about our real-time space simulation framework, which we have successfully used to engage the majority of both our male and female students for the past 5 years. We will brief you on how we introduced students to this new environment and the challenges our students encountered. Then, we will explain how the platform works and how you can use this project in your own classroom.

Access, Barriers, and Perceptions in K-12 Computer Science Education: Problems and Solutions |

Description Coming Soon!

Teachers' Voices In State-Level CS Education Reform | *Renee Fall, Art Lopez, John Owen, Dianne O'GradyCunniff, David Petty, Chinma Uche and Carol Yarbrough*

Many states are now making educational policy decisions that will impact how CS is taught in the future. What standards are used? Which teachers are allowed to teach CS? What credentials must they have? How is professional development provided? Teachers must play a role in influencing state-level CS education reform and making CS available to all students. But where are their voices heard? How do they get a seat at the table? This session brings together six teachers from different states involved in the Expanding Computing Education Pathways Alliance (ECEP) to share stories of their roles beyond their own classrooms and schools. Panelists will inspire educators to contribute their own voices to state-level reforms. Discussion will focus on lessons learned and tips for balancing advocacy with educators' other roles.

Administrator session: TBA

Description Coming Soon!

Mini sessions

Google Drive Facilities Web Development Teaching | *Gayle DiCarlantonio*

Google Drive provides tools for coding and showcasing finished student websites (www.palmdigital.org). Students create folders for their websites within their own publicly shared website folders. For the editing of HTML5, CSS3, and JavaScript files, they create workspaces using the Editey IDE, which can display multiple files in a simple text editor with live preview. Separate panes allow html, css, and javascript files to be visible and simultaneously edited. Finished websites are also published via Editey. Editey provides basic text editing in an attractive, uncluttered interface that is ideal for beginning coders. A notable feature is its seamless integration with Google Drive, delivering the same level of access and collaboration that users currently enjoy with Google docs.

A Visual Debugger for Java in Eclipse | *Victor Milenkovic and Joseph Masterjohn*

The Visual Debugger is a plugin for the Eclipse development environment that provides a visual representation of data structures in debug mode. This representation closely matches the one used for instruction and helps the student connect classroom learning with the programming assignments. The instructor can also use the tool to generate handouts and to animate algorithms. The tool generates a record of the students' use, which the instructor can use to identify areas for more intensive instruction.

Concurrent Sessions

2:30–3:30 pm

Advanced Placement Computer Science: Two Courses, All Students | *Owen*

Astrachan, Lien Diaz, Rich Kick and Sandy Czajka

Engagement, student interest, rigor, and effective pedagogy are the cornerstones two AP Computer Science courses that can facilitate differentiated instruction, allow for problem-centric and other approaches, and engage a diverse audience of students. Participants will learn about different approaches and courses for both the APCSA and APCSP courses -- including complete curricula that are online and freely available. Participants will learn about the courses and exams, so that they will have the knowledge to introduce these courses in their schools and about professional development opportunities offered by College Board and others as part of College Board approved instruction. At the center of

the discussion are experiences from those having successfully offered high school APCSP and APCSA courses and the challenges for new teachers in offering such courses.

Inspiring Passion and Creativity in the Computer Science A Curriculum with Genius Hour | *Brandon Milonovich*

Google has become well known for their "20% time" or "genius hour", which has great implications in the classroom. Teaching AP courses often means guiding students through a curriculum with relatively little flexibility with regard to course content and student choice. A growing body of research supports the notion of student choice in the classroom, often through a project based learning approach. In this session, participants will have the opportunity to see the model used in a Computer Science A course to engage students throughout a year long project investigating projects related to Java programming that they are passionate about.

Starting a High School or Middle School Cyber Competition Team | *Chris Simpson*

Cyber competitions for high school and middle school students are growing around the country. Examples of these include local competitions like the San Diego Mayors' Cyber Cup and national competitions like Cyber Patriot. Cyber competitions support STEM education by getting students interested in cyber security and other STEM fields like computer science and engineering. As they learn more about cyber security they learn more about computers and how they work. In this talk I will dispel that fear provide guidance on how to start a Cyber Competition team whether or not you have any experience with cyber security. You will learn: 1. The basics for forming a team. 2. Where to find learning material. 3. How to get community support. 4. Where to find competitions. 5. Benefits of cyber competitions 6. I will also discuss working with your IT department.

Internet Underpinnings for Computing Security | *Liz Fraumann and Craig Hardin*

You wouldn't hand a 16-21 year old a driver's license without some training/education first, yet that's what many schools and parents are doing with computers, tablets, smart phones & other devices we give our kids. Typically youth are not given any foundational awareness, let alone understanding of potentially serious and lasting consequences from what's happening every time they push a key, button or icon. This interactive session will provide a series of short high school student produced videos that explain in simple terms what some of the current technology is and how it works (i.e. networks, IP addresses, routing, OSI 7 layer model, etc.) We will demonstrate interactive games (with audience participation) that cover relevant cyber/cybersecurity topics with content that any teacher in any school, can incorporate in any class K-12, to energize, or even as a reward for the students.

Gutsy Teachers in Rural Kentucky Tackle Coding For the First-Time Ever | *Stephanie Younger*

The Kentucky Valley Educational Cooperative (KVEC) is a public educational agency in Eastern Kentucky, a geographic area about which the The New York Times commented "[It] just might be the hardest place to live in the United States. Statistically speaking." These rural Appalachian public schools collaborate through the Cooperative to provide programs and services for their schools aimed at empowering students to change those statistics. *Creative Coding Through Games and Apps* is designed to provide students, whose teachers have no programming or computer science experience, with an

opportunity to learn how code. Meet a teacher who was gutsy enough to dive right in and teach programming, using this curriculum and listen to the stories of students' lives have changed, forever.

Before We Code: Renewing Emphasis on Teaching Problem Solving and Lessons to Learn from Design Thinking | *Joseph Dioguardi*

Computer Science education has long been focused on the teaching of the skills of problem solving. Design Thinking is a systematic approach to design. With renewed emphasis on coding in K-12 education, computer science teaching must maintain its focus on passing on the skills of problem analysis and computer science theory. What lessons can we learn from the process of design thinking and how can we apply it in the teaching of computer science?

Administrator session: TBA

Description Coming Soon!

Mini Sessions

Blended Learning in Introductory CS: Lessons From A Happy Rollout | *Joe Thompson*

At Gunston, we introduced CS by using blended platforms, most notably CodeHS. This helped us by allowing students who wished to push ahead to do so, while others who needed more time could move at their pace. In addition, a number of students were able to take advantage of the platform despite not being part of a class at all. CS is a very broad field of endeavour, and blending/flipping helps embrace all of it.

3D Printing For Everyone | *Carol Sweeney*

3-D printing has become popular in schools. We buy the printers but then what? I started a program in my High School last year with the goal that every freshman in Computer Resources class would be able to 3D print. We were delighted by printing nametags, then designing useful items like boxes, and phone holders. The Environmental Science students designed attachments for reuse of water bottles. One student studied the prosthetic hand and made a successful model, which was donated. Working with other teachers for 3D curriculum integration, we developed a planning document such that printing steps were documented and final results captured. We have a book of all of those project pages and samples to show.

Problem-Centered Programming Instruction Problem List | *Stephen Hughes and J. Philip East*

In 2013 we began discussing what programming instruction might look like if it focused on the problems being programmed rather than the language features being used. We presented these ideas at CSTA 2015 and were asked to share to the problem list we might show students on the first day. This session will briefly (re)introduce this teaching style; share a categorized list of problems and a discuss problem selection considerations and rationale.

Closing Keynote: TBA

4:00–5:00 pm