NSF Project Showcase

SIGCSE 2012

March 1 - 3

Raleigh, North Carolina

National Science
Foundation

Program at a Glance

Thursday, 10:00 a.m.—11:30 a.m.

Thursday, 3:00 p.m.—4:30 p.m.

Friday 10:00 a.m.—11:30 a.m.

Program at a Glance (cont.)

Friday 10:00 a.m.—11:30 a.m. cont.

How to Run a Successful REU Site Program (CNS-REU) 15
Susan D. Urban, Michael Shin, and Mohan Sridharan

Friday 3:00 p.m.—4:30 p.m.

FRABJOUS CS—Framing a Rigorous Approach to Beauty and Joy for Outreach to Underrepresented Students in Computing at Scale (CNS-CE21) 16
Dan Garcia, Tiffany Barnes, and Brian Harvey

Exploring Computer Science— An Equitable Learning Model for Democratizing in K-12 Computer Science Education (CNS-BPC & DUE-MSP) 17
Joanna Goode and Gail Chapman

How to Broaden Participation and Scaling Up Computational Thinking by Bringing Game Design into Middle Schools (CNS-BPC & DUE-MSP) 18
Alexander Repenning, Kris Gutierrez, Jeffrey Kidder, and David Webb

Cybersecurity Laboratory: Enhancing the Hands-on Experience in Cybersecurity Education (DUE-SFS) 19
Susanne Wetzel, Werner Backes, and Ruth Schwartz

Saturday 10:00 a.m.—11:30 a.m.

Personal Robots for CS1: Next Steps for an Engaging Pedagogical Framework (DUE-TUES) 20
Douglas Blank and Deepak Kumar

Cyberlearning Environments with Emphasis on Social Approach and Collaboration (DUE-TUES) 21
Jafar Saniie, Erdal Oruklu, and Tricha Anjali

REU Site for Smart UAVs (CNS-REU) 22
Saad Biaz and Wei-Shinn Ku

An Interactive Undergraduate Data Mining Course with Industrial-Strength Projects and Hypertextbook (DUE-TUES) 23
Rakesh Verma, Peng Chen, and Irene Chen

Tools and Best Practices for Improving the Quality of Students' Commenting Skills

Thursday, 10:00 a.m.—11:30 a.m.

Peter DePasquale (College of New Jersey), **Michael Locasto** (University of Calgary), and **Lisa Kaczmarczyk** (UC San Diego)

For students, the process of documenting software is unlike almost any other activity in the traditional Computer Science undergraduate experience. The act of writing quality source code comments requires developing skills in clear, concise technical writing and applying these skills in concert with an understanding of both the practical and theoretical underpinnings of a section of code.

From a learning perspective, students can find it difficult to create quality source code documentation because they believe that the code is ultimately more important than the prose explaining it, and they can struggle to gain an understanding of how such a construct might be evaluated by their professor, peers, or employers.

Instructors can find it equally hard to discuss such practices in both introductory and advanced undergraduate courses. Furthermore, comment *grading* is an imprecise, labor-intensive procedure at best. But just what practices should we be encouraging students to embrace?

Our project centers on the development and classroom use of our COMTOR tool. COMTOR provides a platform for helping assess source code documentation in an objective, structured fashion. We are also engaging in a community development effort that combines undergraduate research, discovery of best practices, and educator outreach to help nurture a generation of students possessing a renewed appreciation of the value of good source documentation.

Student Participation in Humanitarian FOSS

Thursday, 10:00 a.m.—11:30 a.m.

Heidi Ellis (Western New England University), **Gregory Hislop** (Drexel University), and **Ralph Morelli** (Trinity College)

Free and Open Source Software (FOSS) is software that is developed collaboratively in an open environment and is freely available to everyone. Humanitarian FOSS (HFOSS) is software that meets humanitarian needs in areas such as health care, economic development, and education. HFOSS projects can provide a rich learning environment for students due to the transparency of the development process, community support, and availability of project artifacts. In addition, the humanitarian nature of HFOSS can attract students to the major by demonstrating the social power of FOSS and allowing students to improve the human condition. The HumIT and SoftHum projects both utilize HFOSS as a base for student learning.

SoftHum: The SoftHum project has investigated a variety of ways that students can contribute to and participate in HFOSS projects including contributing code, fixing bugs, documenting, design, and more. The SoftHum effort has developed a set of processes for various class levels and an array of teaching materials to aid instructors in involving students in HFOSS projects. In addition, a set of guidelines to help instructors get started and an approach for selecting appropriate HFOSS projects for a class have also been developed.

HumIT: The HumIT project began by exploring ways in which students could be engaged in providing support for IT issues of large-scale HFOSS applications. The idea was that students could provide virtual IT support for a variety of HFOSS application installations, thereby reducing the effort required to install and maintain the applications for the non-profit groups that use such applications. However, it was quickly discovered that many HFOSS deployments prefer a local person for IT support and that the details of individual deployments of HFOSS applications were too complicated for students to be able to easily help in IT support. Therefore, attention shifted to ways in which students could provide IT support for the actual project (e.g., writing and testing installation instructions).

Computing in the Arts: A Model Curriculum

Thursday, 10:00 a.m.—11:30 a.m.

Bill Manaris and **Renee McCauley** (College of Charleston)

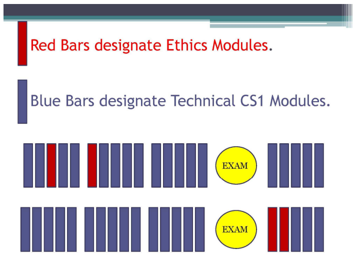
Computing in the Arts (CITA) is an innovative, interdisciplinary Bachelor of Arts degree program, which integrates computer science and information technology with traditional art theory, practice, and performance. CITA combines creativity, problem solving, and computational thinking through an interdisciplinary curriculum of courses offered by Computer Science and Arts departments (such as Music, Art History, Studio Art, and Theatre). It is easy to implement at traditional universities as it reuses and interweaves existing CS and Arts courses. Knowledge synthesis is achieved mainly through four synthesis courses - interspersed throughout the curriculum - where students learn how to integrate CS and Arts knowledge.

This project addresses a recognized need of the 21st century technological society: broadening participation and excellence in computing education. It develops the CITA model and synthesis courses for easy adaptation by other institutions.

Integrating Ethics into Computer Science Courses: Looking for Statistically Significant Effects

Thursday, 10:00 a.m.—11:30 a.m.

Keith W. Miller, Mary Tracy, (Univ. of Illinois Springfield),
Michael Loui (University of Illinois Urbana Champaign) and
Ken Urban (Parkland Community College)


Red Bars designate Ethics Modules.
Blue Bars designate Technical CS1 Modules.
EXAM
EXAM

Curriculum development is often thought of as a zero-sum game: if you put something in, then you must take something out. This argument arises when discussing integrating computer ethics into a computer science course: if you put ethics in, what are you going to take out? But is the zero-sum assumption always correct? Perhaps adding ethics into a computer science course will motivate students to learn both ethics and technical skills in a complementary manner. We propose to study exactly this notion: we hypothesize that if instructors replace some teaching time for technical skills (in this case online teaching modules) with modules that emphasize professional ethics, students will score better on assessments of their technical skills. We anticipate that students' performances will improve because they will be motivated to study harder.

We propose to test this notion using online teaching modules for the introductory programming course for computer science majors, CS1. The "treatment group" will start the ethics modules earlier in the course, prior to the midterm assessment. The "control group" will complete the ethics modules after the midterm. We hypothesize that the treatment group will outperform the control group at midterm in both aspects. There will be a second assessment of both groups at the end of the course, when both groups will have completed all the same modules, though in a different order.

If this study demonstrates that computer science instructors have little to lose and much to gain by including ethics in CS1 and other undergraduate courses, we anticipate many positive consequences. One important consequence is that we might see higher retention rates for female students in computer science; some research suggests that women are more interested in STEM subjects, and particularly in computer science, when technical details are put into context.

Engaging African Americans in Computing through the Collaborative Creation of Musical Remixes

Thursday, 3:00 p.m.—4:30 p.m.

Brian Magerko and **Jason Freeman** (Georgia Tech)

The EarSketch project at Georgia Tech is a multi-year effort to leverage culturally situated education with learner creativity to create a learning environment that enables learners to remix hip hop music and other genres by learning how to write computer code. EarSketch has three main components: a software environment for remixing music via code, a curriculum for using the software in a formal learning environment, and a social media website for sharing projects. In the EarSketch software environment, learners will control loops, beats, effects, and filters by writing small Python code snippets that make use of commands provided by our custom Python API for Reaper, a low cost commercial digital audio workstation (DAW). The EarSketch curriculum is a methodology being developed for using the software within a formal CS education setting. We are iteratively testing across two years, with summer workshops being held each year to provide in-depth case studies of the software and curriculum before a final evaluation in Year 3 in a computer science class at Lanier High School, an Atlanta-area high school. The social media sharing site, which is currently under development, is intended to be integrated with the curriculum and software to provide a seamless way for students to upload their projects, be exposed to other students' work, and download and remix using other projects' code and audio assets (with proper attribution). The end result of this work will be a rigorously evaluated approach to CS education that aligns strongly with the key concepts and learning goals in the CS: Princples curriculum framework.

Building a K-12 Computing Pipeline in Alabama to Address Participation Diversity

Thursday, 3:00 p.m.—4:30 p.m.

Jeff Gray (Univ. of Alabama), **Mike Wyss**, **Shelia Cotton** (Alabama-Birmingham), and **Shaundra Daily** (Clemson)

Through support from three NSF programs, a pipeline of computing awareness has been ongoing in Birmingham, Alabama since 2008. An ITEST award initiated the effort in Birmingham by defining a group of student cohorts that continued over three years through a series of activities: Alice and game programming (summer before 10th grade), linear algebra focused on examples from the Alice experience (10th grade in-class), computer visualization in the context of Mechanical Engineering (summer before 11th grade), and introduction to robotics and Java (during 11th and 12th grade). This project also trained teachers in both Alice and linear algebra. As a result of the City of Birmingham purchasing XO laptops for each child in elementary school, a DR K-12 award is bringing training to teachers and students in elementary schools. With a focus on "Integrating Computing Across the Curriculum," the DR K-12 project offers summer camps to teachers and students in Scratch, and provides yearlong professional development training to 4th and 5th grade teachers to assist in raising the awareness of computing across all subjects in the elementary school context. A new BPC award explores the potential for a multi-tiered mentoring model across several age groups. In this project, students from the computer science department at UAB team with secondary education math students to learn about teaching computer science (Fall semester). The team of university students teaches a class focused on Alice and CS Unplugged in the spring to local high schools each week. Three different weeklong summer camps are offered to middle school students, where the high school students from the spring assist collegiate facilitators. The concept provides a peer group of mentors from middle school up to college.

Computational Thinking in IT—A Scenario-based Approach

Thursday, 3:00 p.m.—4:30 p.m.

Deborah Boisvert (Univ. Massachusetts Boston), **Paula Velluto** (Bunker Hill), **Irene Bruno** (George Mason), and **Charles Winer**

Advancing the Successful IT Student through Enhanced Computational Thinking (ASSECT) is a consortium of community colleges and associated 4-year universities spanning five states – Massachusetts, represented by Bunker Hill Community College and University of Massachusetts Boston; Virginia, represented by Northern Virginia Community College and George Mason University; Indiana, represented by Ivy Tech Community College and Purdue University Calumet; Colorado, represented by Cameron University and California, represented by City College of San Francisco. ASSECT is focused on designing industry-relevant scenarios that provide opportunities to learn, demonstrate and assess computational thinking (CT) skills as well as providing professional development for faculty in effective pedagogical approaches and assessment techniques. The project leadership has developed a research-based computational thinking framework that correlates to the ACM Computing Curricula Information Technology Volume (IT 2008) and a rubric that reflects important foundational CT capabilities for Information Technology professionals.

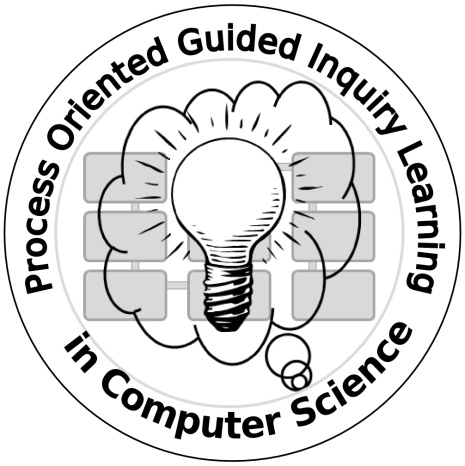
To date, eleven scenarios have been developed, validated by industry partners and piloted in classrooms at the high school, community college and university levels. The scenarios are designed to illustrate typical problems that are presented to students to solve and cover such topic areas as database, web design, Green IT, security, mobile app development, and programming in Alice. Students are immersed in problem solving methodologies and techniques that are measured at the end of the projects via a computational thinking survey that measures through self reporting whether students are engaged in computational thinking while they are solving a particular problem.

Process-Oriented Guided Inquiry Learning in Computer Science

Thursday, 3:00 p.m.—4:30 p.m.

Clifton Kussmaul, R. Libby, and **Carl Salter**
(Muhlenberg College)

This project will develop, refine, validate, and disseminate two sets of process-oriented guided-inquiry learning (POGIL) materials in computer science (CS), specifically for software engineering (SE) and data structures & algorithms (DS&A). These materials will be used to raise the awareness of POGIL and related techniques among CS educators. POGIL is based on the biology of learning and has been developed and validated over the last 15 years. In POGIL, teams of learners (typically 3-5) work on scripted inquiry activities and investigations designed to help them construct their own knowledge, often by modeling the original processes of discovery and research. The teams follow processes with specific roles, steps, and reports that encourage individual responsibility and meta-cognition. The instructor serves as a facilitator, not a lecturer. Studies generally find that POGIL significantly improves student outcomes. The project leverages and extends the successful POGIL approach to another area (CS), and explores some new directions that could also benefit POGIL in general. The project produces materials for adoption by other CS faculty to improve the quality of CS education, and particularly to improve the performance and retention of average and below average students. The project also seeks to identify and foster a community of CS and SE educators to develop and refine POGIL materials in the future.



Databases for Many Majors: A Student-Centered Approach

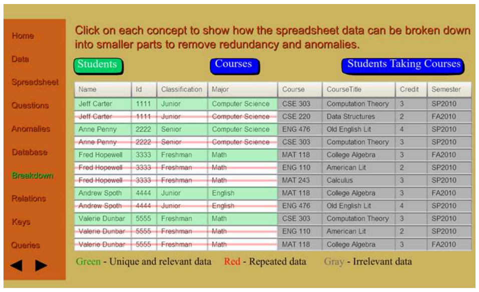
Friday, 10:00 a.m.—11:30 a.m.

Suzanne W. Dietrich (Arizona State University) and
Don Goelman (Villanova University)

With the pervasive role played by databases in our information-centric society and the increasing demand for students with interdisciplinary skills, the next challenge on the horizon for database educators, and the context of this project, is a database curriculum for many majors. This goal is consistent with the enhanced consciousness of computational thinking, which introduces computational methods and models to non-computing majors. The objective of this project is to develop and test animations as pedagogical tools to enable a diverse audience of students with the knowledge of relational databases.

A significant contribution of this research is the creation of two animations for improving learning. One introduces relational databases and how this technology avoids the redundancy and anomalies of other approaches. The second focuses on querying relational databases, including fundamental SQL queries. The animations have already been incorporated into multiple settings, including several non-majors courses offered by computer scientists as well as courses in geography and computational molecular biology. Techniques for customizing the animations to specific domains are currently under development. A website is available as a resource for interested educators: http://databasesmanymajors.faculty.asu.edu

At SIGCSE 2012, stop by the NSF showcase for a demonstration of the animations, a sample cooperative exercise, and a discussion of approaches for integrating the animations into your curriculum.



Click on each concept to show how the spreadsheet data can be broken down into smaller parts to remove redundancy and anomalies.

Name	Id	Classification	Major	Course	CourseTitle	Credit	Semester
Jeff Carter	1111	Junior	Computer Science	CSE 303	Computation Theory	3	SP2010
~~Jeff Carter~~	~~1111~~	~~Junior~~	~~Computer Science~~	CSE 220	Data Structures	2	FA2010
Anne Penny	2222	Senior	Computer Science	ENG 476	Old English Lit	4	SP2010
~~Anne Penny~~	~~2222~~	~~Senior~~	~~Computer Science~~	CSE 303	Computation Theory	3	SP2010
Fred Hopewell	3333	Freshman	Math	MAT 118	College Algebra	3	FA2010
~~Fred Hopewell~~	~~3333~~	~~Freshman~~	~~Math~~	ENG 110	American Lit	2	SP2010
~~Fred Hopewell~~	~~3333~~	~~Freshman~~	~~Math~~	MAT 243	Calculus	3	SP2010
Andrew Spoth	4444	Junior	English	MAT 118	College Algebra	3	FA2010
~~Andrew Spoth~~	~~4444~~	~~Junior~~	~~English~~	ENG 476	Old English Lit	4	SP2010
Valerie Dunbar	5555	Freshman	Math	CSE 303	Computation Theory	3	SP2010
~~Valerie Dunbar~~	~~5555~~	~~Freshman~~	~~Math~~	ENG 110	American Lit	2	SP2010
~~Valerie Dunbar~~	~~5555~~	~~Freshman~~	~~Math~~	MAT 118	College Algebra	3	FA2010

Green - Unique and relevant data   Red - Repeated data   Gray - Irrelevant data

Establishing the Exploring Computer Science Course in Silicon Valley High Schools

Friday, 10:00 a.m.—11:30 a.m.

Dan Lewis, **Ruth Davis**, **Pedro Hernandez-Ramos**, and
Craig Blackburn (Santa Clara University)

This project is a partnership between SCU and twelve San Jose, California high schools with a high percentage of underrepresented and economically disadvantaged students, supported by two NSF grants. The first provides training, equipment and support to teachers willing to offer the "Exploring Computer Science" course (http://csta.acm.org/Curriculum/sub/CurrFiles/ECS_v4.pdf), while the second provides teachers with graduate assistants who are trained in the curriculum, appropriate pedagogy and the challenges of the high school classroom. Since the summer of 2010, we have trained 14 teachers and 5 graduate students in the ECS curriculum. Ten high schools are offering the course in the current academic year. Each graduate fellow is assigned to two teachers and spends about 15 hours a week assisting two teachers both inside and outside the classroom.

The ECS course (originally developed for LAUSD) is a one-year course covering Human Computer Interaction, Problem Solving, Web Design, Introduction to Programming, Computing and Data Analysis, and Robotics. Through ECS, we aim to introduce high school students to the breadth of computing, to strengthen high school teachers' expertise and comfort with computing, to increase the number and diversity of students entering the field of computing and graduates entering the IT workforce, and to improve the communication, instruction, and teamwork skills of graduate student fellows. In addition, this partnership is informing the creation of a set of materials to ease adoption in additional schools. These materials include additional notes for students and for teachers, recruitment materials, and promotional materials aimed at administrators, teachers, students, and parents.

Integrating Security in the Computing Curriculum: Security Injections @ Towson

Friday, 10:00 a.m.—11:30 a.m.

Blair Taylor, Siddharth Kaza, Shiva Azadegan, Michael O'Leary (Towson Univ.), and **Claude F. Turner** (Bowie State Univ.)

Computer security is an increasing international concern. The first response from many colleges and universities has been to add security courses and security tracks to the computer science curriculum, effectively increasing the number of security-skilled graduates. The next step is to integrate security throughout the curriculum, starting with the core programming courses, to ensure that all undergraduate computing students learn fundamental security principles early and often. The Security Injections @ Towson project (www.towson.edu/securityinjections) provides a suite of security injection modules to be used by instructors in the core programming courses, CS0, CS1, and CS2; as well as Computer Literacy, Database, and Web Programming.

Security injections are minimally-invasive security-related modules for use in existing undergraduate computer science classes. Each security injection module targets a particular security issue; for example, modules for the core classes focus on the "big three" security vulnerabilities: integer overflow/errors, buffer overflow, and input validation. Each module has been developed through an intense collaborative process with five diverse partnering institutions ranging from community colleges to comprehensive universities using a Develop-Train-Pilot-Revise-Deploy model. Each module has been tested using formal quantitative and qualitative assessment, including student pre and posttests, faculty surveys, code checks, checklists, and scorecards, to evaluate awareness, applicability, and interest. Results from over 1500 students have shown significant increases in security awareness and the ability to apply secure coding principles.

How to Run a Successful REU Site Program

Friday, 10:00 a.m.—11:30 a.m.

Susan D. Urban, Michael Shin, and **Mohan Sridharan**
(Texas Tech University)

This presentation will outline the fundamentals of organizing and running a successful NSF Research Experiences for Undergraduates Site Program based on our own experience with a program entitled Software Engineering and Service Composition at Texas Tech University (NSF Grant No. CNS-1005212). In particular, we will address eight different categories for consideration. These categories include pre-program administration and planning, student orientation and group bonding, defining appropriate and challenging research projects, the importance of faculty and graduate student mentoring, creating professional development opportunities, making time for social activities, showcasing and publishing research results, and conducting student feedback and program evaluation. The presentation will provide examples of student projects and results that have led to publications and other forms of student recognition. We will also provide a checklist for administrative activities as well as the first week-orientation schedule and the 10-week summer program schedule with professional development activities. The process of preparing for the end-of-program research showcase and continuing post-program communication for research publications will be addressed. The advice outlined will be beneficial for existing programs as well as for those that are considering the submission of NSF REU Site proposals.

FRABJOUS CS—Framing a Rigorous Approach to Beauty and Joy for Outreach to Underrepresented Students in Computing at Scale

Friday, 3:00 p.m.—4:30 p.m.

Dan Garcia (Berkeley)**, Tiffany Barnes** (UNC Charlotte)**,** and **Brian Harvey** (Berkeley)

We propose to reach out to students from underrepresented groups by disseminating, researching, and improving our Beauty and Joy of Computing (BJC) curriculum. We build on our work as two of the five initial pilots for the Advanced Placement Computer Science: Principles course to support the CS10K initiative through in-service teacher professional development and cours e adoption. We use the Snap programming language, an extension of Scratch formerly known as "Build Your Own Blocks," that combines technical sophistication with an attractive drag-and-drop interface. BJC invokes passion, beauty, joy, and awe through engaging students in a rigorous computing curriculum that promotes creativity and collaboration using Snap's visually rich programming environment, while also provoking thought around current events and how computing relates to people's lives.

Our primary objectives are to support the CS10K and AP CS Principles efforts to prepare both high school teachers and students to be creators in computing through this project to: (1) conduct, evaluate and scale team-based professional development for 100 in-service teachers supported by regional collaboration with university faculty and students, (2) empirically investigate the effectiveness of our curriculum, with particular emphasis on adoptability in high schools and understanding what works best for underrepresented minorities, and (3) enhance the Snap software with debugging support, and the ability to run in the browser. We will collaborate with the STARS Alliance to inform our efforts to broaden participation and the Computer Science Teachers Association (CSTA) to provide a vibrant community of support for high school teachers and students engaging the new BJC course.

Exploring Computer Science— An Equitable Learning Model for Democratizing in K-12 Computer Science Education

Friday, 3:00 p.m.—4:30 p.m.

Joanna Goode and **Gail Chapman** (UCLA)

Exploring Computer Science (ECS) is a computer science curriculum designed in response to research findings about the severe limitations of Advanced Placement Computer Science in engaging more than a narrow band of students and the historical injustice of denying a quality computer science education to underrepresented students.

ECS is a year-long college-preparatory course, consisting of six units: Human Computer Interaction, Problem Solving, Web Design, Introduction to Programming, Computing and Data Analysis and Robotics. Students are introduced to the foundational, creative, collaborative, interdisciplinary, and problem-solving nature of computer science through an inquiry-based, equity-focused pedagogy. Teachers are at the core of our work to increase access to computer science knowledge. ECS Teachers are supported with a professional learning community, in-classroom coaching, and on-going professional development.

ECS is being offered in 27 schools in the Los Angeles Unified School District, the second largest district in the country. It is also being offered in a number of other districts, including San Jose, Oakland, and Chicago. ECS has been a catalyst and foundation for Mobilize, a NSF MSP project. At the heart of Mobilize is "participatory sensing"—students use mobile phones and web services to systematically collect and interpret data about issues important to them and their communities. Participatory sensing has been incorporated into the activities of the Computing and Data Analysis unit of ECS.

For more information please visit: www.exploringcs.org; www.mobilizingcs.org

These projects are supported by NSF Broadening Participation in Computing Grant CNS-1042302: Into the Loop and NSF MSP Grant 0962919: Mobilize.

How to Broaden Participation and Scaling Up Computational Thinking by Bringing Game Design into Middle Schools

Friday, 3:00 p.m.—4:30 p.m.

Alexander Repenning, Kris Gutierrez, Jeffrey Kidder, and **David Webb** (University of Colorado)

For over 15 years we have researched and built creativity tools including AgentSheets (2D) and AgentCubes (3D) with the goal to broaden the participation of students in computer science through game design and computational science modeling. We have developed the Scalable Game Design curriculum covering a large scale of student ages ranging from elementary school to graduate school and a wide rage of game design challenges ranging for making basic 1980 arcade games to contemporary games including sophisticated Artificial Intelligence. The presentation will focus on the results from an ongoing ITEST and a beginning CE21 project exploring the fundamental shift of game design based activities from after school programs to in school courses at the middle school level. The first three years of the Scalable Game Design project have exposed over 8000 students to computational thinking. School sites quickly expanded beyond our original Colorado-based focus to include Alaska, California, Georgia, Ohio, South Dakota, Texas and Wyoming. International replication sites, e.g. in Brazil, are emerging. Some of our middle schools are exposing 400 students each year, making it easy to extrapolate to very large numbers of students being exposed nation wide to computational thinking through this strategy. On average, 45% of our participants are girls and 48% are underrepresented students. Motivational levels, measured by their expressed interest to take similar classes, are very high: 74% of boys, 64% of girls (100% for some schools); 71% of white students, and 69% of minority students would like to continue.

SCALABLE
GAME DESIGN

Cybersecurity Laboratory: Enhancing the Hands-on Experience in Cybersecurity Education

Friday, 3:00 p.m.—4:30 p.m.

Susanne Wetzel, **Werner Backes**, and **Ruth Schwartz**
(Stevens Institute of Technology)

This Scholarship for Service (SFS) capacity building project at Stevens Institute of Technology includes the extending of a laboratory and the developing of related learning materials on multi-core and many-core computing in cybersecurity applications and education.

Specifically, the existing cybersecurity laboratory at Stevens was upgraded to include two 48-core Silicon Mechanics nServ and two Nvidia Tesla S2050 GPU computing servers. Instructional modules on multi-/many-core computing and cybersecurity are being developed and integrated into courses on computer algebra and cryptography. The laboratory facility is also being used for (senior design) projects. Howard University is a first adopter of the developed course materials. The project includes rigorous formative and summative evaluation plans with both qualitative and quantitative components coordinated by an experienced independent evaluator.

Personal Robots for CS1: Next Steps for an Engaging Pedagogical Framework

Saturday, 10:00 a.m.—11:30 a.m.

Douglas Blank and **Deepak Kumar** (Bryn Mawr College)

This project develops an exciting and engaging curriculum for teaching introductory computing within the context of using personal robots. The second goal is the design and development of an affordable personal robot that can be brought to the level of a refined product. The project includes both the development of a textbook (titled "Learning Computing With Robots") for use in the teaching of CS1 courses and a new software framework (called "Calico") that enables the use of a choice of robots, programming languages, and operating systems.

We hope to directly address the ongoing crisis in attracting and retaining students into computing disciplines. It supports faculty enhancement, community building for wider adoption and detailed assessment at a diverse range of institutions. It integrates several facets of research and development including rethinking of introductory curricula; design, development, and manufacturing of innovative robot hardware; design and development of a modern, multi-language, cross-platform software framework; context-driven textbook and other teaching materials; and community building in the educational robotics domain.

The project has a potential to develop resources that can be used and shared by instructors worldwide in teaching CS1 and robotics to undergraduate students. It establishes a non-profit, commercial, open-source distribution channel for robot kits, software, and texts (called the "Institute for Personal Robots in Education"). An extensive dissemination plan includes hands-on faculty training workshops to assist a diverse range of colleges and universities to test, evaluate, and adapt the materials.

Personal Robots for CS1 has been developed by Georgia Tech and Bryn Mawr College. You can find all of our materials at wiki.roboteducation.org. Also, please visit the "Experience It! Playground" at SIGCSE 2012 in the exhibition area, which is sponsored, in part, by this project. There you can get some hands-on experience with our materials.

Cyberlearning Environments with Emphasis on Social Approach and Collaboration

Saturday, 10:00 a.m.—11:30 a.m.

Jafar Saniie, Erdal Oruklu, and Tricha Anjali (Illinois Institute of Technology)

Extending higher education through distance learning platforms has proven to be a very powerful method, however this predominately targets classroom environment (as opposed to a laboratory environment). This is in part due to the additional social and technological challenges that a laboratory environment entails when transitioning to a distance learning platform. Our main objective in this project is to create an accessible and easy-to-implement cyberlearning environment with emphasis on collaboration and teamwork. We investigate social aspects and technological solutions for enabling collaboration in cyberlearning environments. Furthermore, we present policies and guidelines for restructuring existing coursework to be compatible for supporting and encouraging collaboration among students. Multiple senior level laboratory courses are included in the test trials and analysis of how well students perform in the collaborative remote laboratory environment is discussed.

REU Site for Smart UAVs

Saturday, 10:00 a.m.—11:30 a.m.

Saad Biaz and **Wei-Shinn Ku** (Auburn)

In 2003, Saad Biaz started at Auburn University an REU Site on Pervasive and Mobile Computing with 8 to 12 participants per summer. His mission is to offer a research experience to underrepresented minorities (Women, African American, Hispanics, and Native Indians) attending 4 year institutions where they are not offered many research opportunities. Overall, 30% of participants fall in this category. In the last two years, this Auburn REU site focused on *Smart UAVs* to address the following challenge: fly autonomously, safely, efficiently, and securely six to twelve UAVs in a limited airspace.

To far we hosted over 80 REU participants who published over 20 papers in regional, national, and international high quality venues. In Summer 2011, the REU participants were organized in three teams. Each team produced high quality technical reports that are currently under review for publication in the *AIAA Journal on Guidance, Control, and Dynamics*. Summer 2012 will our 10th summer.

An Interactive Undergraduate Data Mining Course with Industrial-Strength Projects and Hypertextbook

Saturday, 10:00 a.m.—11:30 a.m.

Rakesh Verma, Peng Chen, and **Irene Chen** (Univ. of Houston)

Due to the explosion of data with the invention of the three key enabling technologies, viz., the computer, the Internet and the World Wide Web, data mining is becoming an increasingly important course in the computer science curriculum. In this presentation, we will discuss the design, implementation, and evaluation results of a new data mining course, for undergraduates, that emphasizes real-world projects and includes a hypertextbook for student support. The course design is flexible and can accommodate a wide variety of student learning styles. This is a joint project with Dr. Ping Chen of University of Houston-Downtown.

Topic Index

Topic Index

Topic	Sessions
Arts in CS	Th 10:00, Th 3:00
Curriculum Development	All Sessions
Diversity	Th 3:00, F 3:00, Sat 10:00
Ethics	Th 10:00
Games	Fr 3:00
Introductory CS	Th 10:00, Th 3:00, Fri 3:00, Sat 10:00
K-12 Education	Th 3:00, Fr 10:00, Fr 3:00
Open Source	Th 10:00
Security	Fr 10:00, Fr 3:00
Skill Improvement	Th 10:00, Fr 10:00, Fr 3:00, Sat 10:00
Teaching Methods	Th 3:00, Sat 10:00
Tools	Th 10:00, Fr 10:00, Sat 10:00
Undergraduate Research	Fr 10:00, Sat 10:00

Meet with Program Officers

This year, NSF Program Officers will be available during most presentation sessions in the meeting area outside the NSF Showcase booth. A specific schedule of times and Program Officers is available at the NSF Showcase booth.

While there will be some open time to meet with Program Officers, some will work by appointment.

Please visit http://www.cs.virginia.edu/~sherriff/nsfshowcase to sign up for an appointment time to meet with a Program Officer or come by the NSF Showcase booth.

About the NSF Showcase

Every year, twenty projects that are currently being sponsored by NSF are asked to present their work in an interactive, personal format during the break sessions and open slots at SIGCSE. The SIGCSE Symposium provides a forum for educators from K-12 through college to discuss issues and new ideas related to the development and implementation of computing curricula, along with other elements of teaching and pedagogy. The goal of the showcase is to share information about programs and research opportunities that attendees might not otherwise hear about.

The NSF Showcase is an annual event at the SIGCSE conference. If you are working on an NSF computer science education related grant, or have recently completed one and would like to present at the showcase, please contact Aaron Bloomfield or Mark Sherriff. The selection process will begin in the fall.

http://www.cs.virginia.edu/~sherriff/nsfshowcase

Aaron Bloomfield - aaron@virginia.edu
Mark Sherriff - sherriff@virginia.edu

For more information, please visit:

http://www.cs.virginia.edu/~sherriff/nsfshowcase

Cover Image:
The Bell Tower at North Carolina State University
Credit: Lucas Layman