



# CCSC Eastern's 28th Annual Regional Conference

---

Richard Stockton College

November 2 & 3, 2012



**The 28<sup>th</sup> Annual Eastern Conference of the CCSC**

**Host: Computer Science and Information Systems  
School of Business, Richard Stockton College  
Galloway, NJ 08205**

**CCSC Eastern's 28th Annual Regional Conference**  
**Richard Stockton College of New Jersey**  
*November 2-3, 2012*

**Program**

**Friday, November 2, 2012**

<b>Time</b>	<b>Event</b>	<b>Location</b>
12:00 pm – 1:00 pm	<b>Registration</b>	Campus Center Grand Hall
1:00 pm – 1:15 pm	<b>Welcome</b> <ul style="list-style-type: none"> <li>• Welcome to CCSCE 2012, Vincent Cicirello, Conference Chair</li> <li>• Welcome to Richard Stockton College, Harvey Kesselman, Provost &amp; Executive Vice President</li> </ul>	Campus Center Theatre
1:15 pm – 2:15 pm	<b>Keynote</b> <ul style="list-style-type: none"> <li>• Trends in Cybercrime (<i>FBI</i>)</li> </ul>	Campus Center Theatre
2:15 pm – 2:45 pm	<b>Coffee Break</b> <ul style="list-style-type: none"> <li>• Coffee / snacks provided</li> </ul>	G-Wing Gallery

**Concurrent Session 1**

2:45 pm – 4:00 pm	<b>Session 1A (Tutorial):</b> <ul style="list-style-type: none"> <li>• Getting Started with the Amazon Cloud: Introduction and Discussion of Curriculum Integration (<i>Peter DePasquale, Matthew Jadud, Kevin Coughlin</i>).</li> </ul>	G-108
2:45 pm – 4:00 pm	<b>Session 1B (Papers):</b> <i>Session Chair: Jill Gerhardt</i> <ul style="list-style-type: none"> <li>• Visualization of Student-Implemented OS Algorithms in Java (<i>J. Adam Fischbach</i>)</li> <li>• Two Concrete Examples of Upper_Level Writing Assignments in an Algorithms Course (<i>Dee Weikle</i>)</li> <li>• On beyond Sudoku: Pencil Puzzles across CS (<i>Zack Butler</i>)</li> </ul>	G-137
4:00 pm- 4:30 pm	<b>Coffee Break</b> <ul style="list-style-type: none"> <li>• Coffee / snacks provided</li> </ul>	G-Wing Gallery

**Concurrent Session 2**

4:30 pm – 5:45 pm	<b>Session 2A (Tutorial):</b> <ul style="list-style-type: none"> <li>• Getting Started with the Amazon Cloud: A Hands-On Experience (<i>Peter DePasquale, Matthew Jadud, Kevin Coughlin</i>).</li> </ul>	G-108
4:30 pm – 5:45 pm	<b>Session 2B (Tutorial):</b> <ul style="list-style-type: none"> <li>• HTML5 Jumpstart (<i>Michael Olan</i>)</li> </ul>	D-004

4:30 pm – 5:15 pm	<p><b>Session 2C (Poster Preview – Faculty Posters):</b>  <i>Brief overviews of the posters on display at the poster session</i>  <u>Session Chair: Timothy Highley</u></p> <ul style="list-style-type: none"> <li>• CloudCoder - A Web-Based Programming Exercise System (<i>David Hovemeyer, Jaime Spacco</i>)</li> <li>• An Approach for Evaluating Open Source Projects for Student Participation (<i>Michelle Purcell, Heidi J. C. Ellis, Gregory W. Hislop</i>)</li> <li>• Community-Based Student Learning via Participation in Humanitarian FOSS Projects (<i>Heidi J. C. Ellis, Gregory W. Hislop</i>)</li> <li>• Developing an IT Simulation to Improve IT Curriculum Effectiveness (<i>Bay Arinze</i>)</li> <li>• Using Vertically Integrated Project Teams to Inspire Student Interest in Computing Careers (<i>Gregory W. Hislop, Massood Towhidnejad, Dean Fontenot, Michelle W. Purcell</i>)</li> </ul>	G-137
5:15 pm – 6:30 pm	<p><b>Poster Session</b></p> <ul style="list-style-type: none"> <li>• Poster viewing and discussions with poster presenters</li> </ul>	Campus Center Grand Hall
5:45 pm – 6:30 pm	<p><b>Reception</b></p> <ul style="list-style-type: none"> <li>• Reception</li> </ul>	Campus Center Grand Hall
6:30 pm – 8:00 pm	<p><b>Banquet</b></p> <ul style="list-style-type: none"> <li>• Dinner</li> <li>• Performance by the Stockton Faculty Band</li> </ul>	Campus Center Event Room

**Saturday, November 3, 2012**

<b>Time</b>	<b>Event</b>	<b>Location</b>
7:30 am – 8:30 am	<b>Continental Breakfast</b>	C/D Atrium
7:30 am – 8:30 am	<b>Registration</b>	G-Wing Gallery
8:00 am – 12:00 pm	<b>Programming Competition</b>	D-004 / D-027

**Concurrent Session 3**

8:30 am – 9:45 am	<b>Session 3A (Workshop):</b> <ul style="list-style-type: none"> <li>Intro 3 Ways: An Introduction to Three Environments for Teaching Introductory Programming (<i>Adrienne Decker and Frances Trees</i>). This is part 1 of a 2 part workshop. Part 2 follows in next session.</li> </ul>	G-108
8:30 am – 9:45 am	<b>Session 3B (Papers):</b> <u>Session Chair: Peter DePasquale</u> <ul style="list-style-type: none"> <li>Teaching Introductory Programming through Reusable Learning Objects: A Pilot Study (<i>Giovanni Vincenti, James Braman, J. Scott Hilberg</i>)</li> <li>A Tutoring System for Debugging: Status Report (<i>Elizabeth Carter and Glenn D. Blank</i>)</li> <li>Arduino as a platform for a Computer Organization Course (<i>Nathan Sprague</i>)</li> </ul>	G-137
8:30 am – 9:45 am	<b>Session 3C (Papers):</b> <u>Session Chair: Duo (Helen) Wei</u> <ul style="list-style-type: none"> <li>Exploration into Reproduction Literacy, An aspect of Digital Literacy, and its Correlation with Noel-Levitz Measures (<i>Leighann S. Forbes, Ryan C. Leonard, Theresa M. Vitolo</i>)</li> <li>An evaluation of a Cooperative Learning Method in Programming and Problem Solving I (<i>Duo (Helen) Wei</i>)</li> </ul>	G-138
9:45 am – 10:15 am	<b>Coffee Break</b> <ul style="list-style-type: none"> <li>Coffee / snacks provided</li> </ul>	G-Wing Gallery

**Concurrent Session 4**

10:15 am – 11:30 am	<b>Session 4A (Workshop):</b> <ul style="list-style-type: none"> <li>Intro 3 Ways: An Introduction to Three Environments for Teaching Introductory Programming (<i>Adrienne Decker and Frances Trees</i>). This is part 2 of a 2 part workshop. Part 1 was in previous session.</li> </ul>	G-108
10:15 am – 11:30 am	<b>Session 4B (Papers):</b> <u>Session Chair: Giovanni Vincenti</u> <ul style="list-style-type: none"> <li>Categorizing the School Experience of entering Computing Students (<i>Thérèse Smith and Robert McCartney</i>)</li> <li>Applications of Computing Technology–Combining Tangible Skills and Theory in CS0 (<i>Nadine Hanebutte</i>)</li> <li>Scaling Office Hours: Managing Live Q&amp;A in Large Courses (<i>Tommy MacWilliam and David J. Malan</i>)</li> </ul>	G-137

10:15 am – 11:30 am	<p><b>Session 4C (Nifty Ideas):</b></p> <p><i>Nifty Ideas and Lightning Talks:</i></p> <p>We invite instructors to come and share "nifty" course assignments and/or interesting uses of technology they believe conference attendees will find useful and can incorporate in their courses. All topics will be considered. A few prepared topics include:</p> <ul style="list-style-type: none"> <li>○ A Coloring Contest and Other Tools (<i>John Beidler</i>)</li> <li>○ Incorporating 2D Game Development into CS1 (<i>Val Kolesnikov</i>)</li> <li>○ Heather Amthauer &amp; Michael Flinn</li> </ul>	G-138
------------------------	--	-------

11:45 am – 1:15 pm	<p><b>Luncheon and Awards</b></p> <ul style="list-style-type: none"> <li>• Luncheon</li> <li>• Awards Ceremony for Best Paper and Best Poster Competitions</li> <li>• Awards for Programming Competition</li> </ul>	Campus Center Event Room
-----------------------	---	-----------------------------

**Concurrent Session 5**

1:15 pm – 2:30 pm	<p><b>Session 5A (Tutorial):</b></p> <ul style="list-style-type: none"> <li>• Engaging and Motivating Computing Activities with Computational Thinking Concepts You Can Use (<i>Heidi Webb</i>)</li> </ul>	G-108
1:15 pm – 2:30 pm	<p><b>Session 5B (Panel):</b></p> <ul style="list-style-type: none"> <li>• Green Computing</li> </ul>	G-137
1:15 pm – 2:00 pm	<p><b>Session 5C (Student Research Showcase):</b></p> <p>Student presentations of summer research projects:</p> <ul style="list-style-type: none"> <li>• User Interfaces for generating SPARQL queries for end-users (<i>Akshaye Dhawan</i>)</li> <li>• Classifying Operators for Permutation-based Genetic Algorithms (<i>Robert Certera</i>)</li> </ul>	G-138

2:30 pm – 3:00 pm	<p><b>Steering Committee Meeting</b></p> <ul style="list-style-type: none"> <li>• Planning meeting for 2013</li> <li>• All who are interested in organizing next year's conference are invited</li> </ul>	G-103 conference room
----------------------	---	--------------------------

**Post-Conference Workshops**

3:00 pm – 6:00 pm	<p><b>Workshop:</b></p> <ul style="list-style-type: none"> <li>• The Finch, a Robot for the CS Classroom (<i>Tom Lauwers</i>)</li> </ul>	D-004
3:00 pm – 6:00 pm	<p><b>Workshop:</b></p> <ul style="list-style-type: none"> <li>• Teaching 2D Game Development in CS curriculum (<i>Val Kolesnikov</i>)</li> </ul>	D-027
3:00 pm – 6:00 pm	<p><b>Workshop:</b></p> <ul style="list-style-type: none"> <li>• Geographic Information Systems (GIS) Applications (<i>Weihong Fan</i>)</li> </ul>	AS-209

## **Nifty Ideas Presentation Abstracts**

### **A Coloring Contest and Other Tools** (*John Beidler, University of Scranton*)

I have always been a fan of Cliff Schafer's Algoviz website. Every time I use one of the visualizations that I download from Algoviz, I wonder about the possibility of having a visualization toolkit that made it easy for students to obtain visualization of what is going on inside their software without spending an enormous amount of time learning about Java's AWT and Swing graphics.

Specifically, my favorite assignments, which I give to students in my Data Structures and Algorithms course, revolves around the data structure representation of the map of the 48 contiguous United States and solving various graph and digraph problems. I always felt that these assignments would be more meaningful to the students if they could see a visualization of the map of the US change as their software attempts to solve the problem. Every once in while I would spend some time looking for a map of the US, or some other part of the globe that I could use with Java, I could not find any map that I could use that could be controlled by a Java program.

I also teach a web development course and on my last sabbatical I decided to focus on ways of improving the presentation of SVG graphics in my web course. As I investigated ways of incorporating SVG graphics in Javascript I ran across an open source SVG map of the US, as well as SVG maps of other parts of the world. After completing the focus of my sabbatical I quickly went back and built a script to extract the essence of the svg US map and make it available to a Swing Jpanel, which made it easy to incorporate a US map into the Java assignments in my sophomore level Data Structure and Algorithms course. One result is an assignment called The Coloring Contest where the students employ various methods for four coloring a map and watching the results as their program colors and recolors states.

The student reaction to the visualization was much more positive than I expected. Their basic response included the observation that if they would have seen visualizations of some of their CS 1 and CS 2 assignments. After listening to their suggestions I added a visualization component to a jar file of resources we use in the CS 2 and Data Structures and Algorithms Course. The resources are available from a link on my home page, at <http://www.cs.scranton.edu/~beidler>.

### **Incorporating 2D Game Development into CS1** (*Val Kolesnikov, Baker University*)

CS1 courses can be a vehicle to attract more students into the field of Computer Science. They can also be either positive or negative influences for retaining CS students in the discipline. CS1 courses provide fundamentals for the rest of CS curriculum and therefore have to be rigorous. Rigor does not mean dry content. It is possible to make the course fun and exciting by incorporating elements of gaming. This talk will center on how gaming aspects have been introduced into several CS courses in our curriculum. The most important part of the conversation will be on combining computer graphics and elements of gaming into CS1 type and service type courses with CS content. We found that this approach increases retention and brings higher number of students into CS field. We will present which aspects of gaming can be introduced with which general topics and then discuss possible assignments and projects.

## **Faculty Poster Presentation Abstracts**

### **Community-Based Student Learning via Participation in Humanitarian FOSS Projects** (*Heidi J. C. Ellis, Western New England University; and Gregory W. Hislop, Drexel University*)

Free and Open Source Software (FOSS) projects offer a rich learning environment for computing students due to the transparent nature of the process and artifacts used to develop the product. Student participation in such a project allows students to learn collaboratively within a professional community while working on a real-world, frequently international project. This learning differs from a traditional classroom environment because students learn from the community itself and the instructor becomes a guide rather than the main source of knowledge. Humanitarian FOSS (HFOSS) projects have the additional benefit of attracting students due to their altruistic nature and the possibility for benefiting the human condition.

This poster presents results of a multi-year study into student opinion of learning via participation in HFOSS projects. The study used a 5-point Likert survey to elicit student opinion of perceived learning. The study involved nine different courses and summer internships offered at four different academic institutions between summer 2008 and fall 2011 resulting in 151 student survey responses. The survey covered three main aspects of the impact of student participation in HFOSS projects:

1. The impact of participation in an HFOSS project on student attitude towards computing
2. The degree of perceived learning related to software engineering knowledge and skills
3. The impact of participation in an HFOSS project on major selection and career plans

This poster will report on student survey results which appear to indicate that students are positively impacted by participating in HFOSS projects. Results of upper-tailed hypothesis tests for the mean provide sufficient evidence to conclude that student participation in HFOSS has a significantly positive impact on the three study aspects of attitude, software engineering knowledge, and career planning.

**An Approach for Evaluating Open Source Projects for Student Participation** (Michelle Purcell, Drexel University; Heidi J. C. Ellis, Western New England University; and Gregory W. Hislop, Drexel University)

Student participation in free and open source software (FOSS) has potential to improve student learning in computing majors. Experiences contributing bug fixes, testing, writing documentation and developing new features can enable students to learn in a more authentic environment developing technical skills as well as teamwork and communication skills. However learning curves related to a large complex code base, use of development tools, understanding of community interactions, and scheduling of class deliverables with the FOSS calendar are only a few of the hurdles to education inherent in learning in a FOSS environment. One major roadblock to instructors desiring to support student involvement in FOSS projects is the difficulty in identifying FOSS projects suitable for student participation. This poster presents one attempt to address this problem by providing a framework of evaluation criteria that can be used to determine approachable FOSS projects for student involvement.

Building upon several years of experience involving students in humanitarian open source software projects, an approach was developed to assist instructors in identifying appropriate FOSS projects. This poster presents this framework including a set of criteria that impact student involvement along with approximate metrics for assessing the criteria. This set of criteria includes Mission Critical criteria that must be fulfilled in order for a project to be deemed suitable. The Mifos microfinance and the Sigmah humanitarian project management software are used as examples to demonstrate the utility of the approach. This model and approach are sufficiently flexible to support differing types of student participation including fixing bugs, writing code, writing documentation, testing, and more. In this poster, we present the framework and findings from initial steps to evaluate its validity.

**CloudCoder - A Web-Based Programming Exercise System** (David Hovemeyer, York College of Pennsylvania; Jaime Spacco, Knox College)

CloudCoder is a web-based programming exercise system designed for introductory programming courses. Using CloudCoder, instructors can assign practice problems to reinforce concepts and assess mastery of skills. Students access their assigned problems using a web browser. A typical problem asks the student to write a function or complete program to perform a simple task such as classifying input or performing a computation on input data. When the student submits her solution to a problem, the server tests her code against a series of test cases designed by the instructor and reports which tests executed correctly. CloudCoder is inspired by existing systems such as Codingbat, but is designed to be installed and used widely. As such, CloudCoder is open source (<https://github.com/daveho/CloudCoder>) and supports programming exercises in multiple languages (currently C, Java, and Python).

Our poster will report our initial experiences using CloudCoder in introductory programming courses in Spring 2012. We will report on the extent to which students have used CloudCoder, their impressions, and (hopefully) evidence that CloudCoder is effective at improving student learning. We will also present on future plans, including social/collaborative features to make it easy to share useful exercises between courses and institutions. We will also invite CCSC-E participants to join us in using CloudCoder for assigning and sharing programming exercises.

### **Developing an IT Simulation to Improve IT Curriculum Effectiveness** (*Bay Arinze, Drexel University*)

Business simulation games are effective for developing business acumen; and for business training and development. They are extremely popular in Fortune 500 companies and MBA programs alike as a replacement for "dry" lecture-based classes on business strategy and functional area analysis. Business schools use them to foster strategic and critical thinking, training in the integration of various business functions, and in marketing analysis, financial and accounting strategy, operations, teamwork and leadership.

There are virtually no specialized simulation tools that exist to enable the Information Systems student to develop the same critical thinking, strategic and tactical planning abilities and teamwork and leadership that general management and business students have benefited so greatly from. An IT Strategy Simulation will enable MIS and IT students to gain a knowledge and mastery of critical skills in strategic and tactical IS/IT decision making. In addition, they will understand, through simulations, the range of alternatives that are available to the IT manager and what the consequences of these are on the business.

Key dimensions of the simulation will include:

1. Functional or Total enterprise
2. Competitive or Non-competitive
3. Interactive or Noninteractive
4. Industry specific or Generic
5. Individuals or by Teams
6. Deterministic or Stochastic
7. The time period simulated

An IT Strategy Simulation will enable MIS and IT students to gain a knowledge and mastery of critical skills in strategic and tactical IS/IT decision making. In addition, they will understand, through simulations, the range of alternatives that are available to the IT manager and their consequences on the business.

### **Using Vertically Integrated Project Teams to Inspire Student Interest in Computing Careers** (*Gregory W. Hislop, Drexel University; Massood Towhidnejad, Embry-Riddle Aeronautical University; Dean Fontenot, Texas Tech University; Michelle W. Purcell, Drexel University*)

There are too few students majoring in computing disciplines. In 2008, the most recent year for which data are available, less than 58,000 computing degrees were granted. However, the Bureau of Labor Statistics reports that new jobs for software engineers alone will increase by 295,000 between 2008-2018. Compounding this issue is the lack of interest in computing fields by women and underrepresented groups and the need for developing computational thinking capacity as a "fundamental skill for everyone, not just computer scientists."

Developing computational thinking capacity is a pressing national need, not just for computing professionals. Meeting this need will require enthusiastic participation of many more students than those currently choosing computing education. The Inspire-CT project, a National Science Foundation (NSF) funded project, was created to explore increasing student interest in computing by engaging students in meaningful computing projects much earlier in their education, even as early as pre-college. Unlike the current instructional approach, which begins with extensive skill building focused on primarily learning to program, Inspire-CT is designed to show students the potential of computational thinking in action by earlier exposure to projects that have a direct impact on people and society.

The approach uses guided active learning through project work leveraging the excitement generated by student teams working on capstone design projects. Through vertical teaming of advanced undergraduates, pre-college and early college students the capstone experience allows much less experienced students to comprehend and share in that excitement based on their skill levels. In addition, vertical teaming mitigates issues related to lack of technical capability of pre-college and early college students and in some cases lack of adequate instructional support. The poster presents project goals and objectives, activities and results, and future plans.



## **Student Summer Research Showcase Presentation Abstracts**

### **User Interfaces for generating SPARQL queries for end-users** (*Kiriakos Kontostathis, Michelle Tanco, Akshaye Dhawan (Faculty Advisor)*)

This work examines the problem of generating effective web-interfaces for searching and browsing data represented in Resource Description Framework (RDF) and Web Ontology Language (OWL). The goal of this work is to design interfaces that are reusable across different ontologies without the need for extensive customizations to the source code. As a proof of concept, we present the design of Dynamic Interface for Navigating Ontologies (DINO) - a web interface to query and browse semantic data. We have tested our interface against about a dozen ontologies and it has worked without any revisions in the underlying source code.

### **Classifying Operators for Permutation-based Genetic Algorithms** (*Robert Cernera, Vincent Cicirello (Faculty Advisor)*)

Genetic algorithms model the genetic processes of living organisms at the cellular level. They are used to mimic the process of natural selection in order to find an optimized solution. In simple GA, we start with a population of “chromosomes,” usually represented as bit strings. Each chromosome is evaluated for its usefulness as a solution by a fitness function. The solutions with the most potential will be chosen to breed new offspring, which in turn eliminates a less optimal solution to maintain a relative population size. The altered population is then reevaluated and the process repeats until a sufficient solution is determined by the fitness function.

An alternate representation for GAs is that of the permutation, where individuals are represented by an ordering over a set of elements. Permutations often are a better fit to a problem, such as scheduling problems. A large variety of specialized mutation operators have been developed for permutations. Our work aims at providing guidance in choosing from among the many available mutation operators.

Fitness-distance Correlation is a measure of the relationship between the fitness of a GA individual and its “distance” to the optimal solution of the problem. It can be used to characterize the expected problem solving performance of GA. Additionally, mutation is meant to be a “small” change to an individual; which consequently depends on which distance measure is most relevant to the problem. There are many choices for distance measures, but are generally dictated by the problem trying to be solved. We have profiled the most popular mutators for a variety of distance measures. As a result, there are particular mutator choices that better map to the distance characteristics of a specific problem.

## **Student Poster Presentation Abstracts**

### **Biometrics based Security Features on WME 2.0: A Web based Collaborative E-Learning Environment for Mathematics** (*Chad Vanorsdale, Jessica Novak, Jordan Cannin, W. Liao (Faculty Advisor), O. Guzide (Faculty Advisor)*)

In this poster we present the biometrics based security enhancements to WME 2.0, a Web based Math Learning and Education Environment that allows students, teachers and mathematicians make contribution to form a mathematical learning community, with the intention to bring more excitement and joy to all participants. The security features based on biometrics would form a foundation on trustful online testing system, which is an essential component for online education and distance learning.

### **A Survey of Mobile Operating Systems** (*Chad Vanorsdale, Jordan Cannin, Jessica Novak, W. Liao (Faculty Advisor), O. Guzide (Faculty Advisor)*)

Mobile devices nowadays are capable of supporting a variety of applications, ranging from client side applications such as emailing and texting, to server applications such as file sharing, RPC calls and Web services. However, those capabilities would not be possible without the underlying operating systems. In this poster, we first describe the features and

requirements that distinguish mobile operating systems from traditional operating systems. Then we survey a list of mainstream mobile operating systems, including the client based mobile operating systems and server operating systems.

**A CUDA Implementation of the Harris Corner Detection Algorithm** (*Takumi Bolte, Matthew Lang (Faculty Advisor)*)

General purpose GPU computing (GPGPU) has quickly grown since NVIDIA released the GeForce 256 in 1999. Today, over ten percent of supercomputers in the Top 500 list utilize massively parallel NVIDIA GPUs. The NVIDIA CUDA programming model is an extension of C that allows developers to write GPU-parallel C programs. In this presentation, I will describe my use of CUDA to parallelize the canonical corner detection algorithm: the Harris Corner Detector. I have implemented several CUDA versions of the Harris filter, beginning with a naive implementation and culminating in an implementation optimized for the specific architecture features of the NVIDIA GPUs. The CUDA programming model provides a simple method of parallelizing data parallel algorithms with annotations that extend C--much like OpenMP. This allows a programmer to trivially parallelize existing code. However, naive parallel implementations rarely achieve the speedup possible using massively parallel GPUs. More sophisticated implementations take advantage of the memory hierarchies on the GPU which decrease the latency of read and write operations such as shared or constant memory. The limiting factor to using these memory architectures, however, is the limited amount of shared and constant memory available on each GPU. These implementations need to take into consideration how much memory is available and how it will be distributed among the workload. I will present implementations and speedup comparisons of the Harris filter for the CPU, naive GPU, and several optimized versions that utilize both shared and constant memory. These implementations will be made available as a plugin for the widely used open source image editing tool GIMP (GNU Image Manipulation Program).

**Framework for modeling crowd behavior in evacuation scenarios** (*Frank Capobianco, Matthew Lang (Faculty Advisor)*)

We address the challenges of creating a framework for accurately modeling crowd behavior in threat-based evacuation scenarios using autonomous agents. Many evacuation simulations depict agents fleeing using the nearest exit. Our framework is designed such that agents are free-roaming entities that occupy a given space and when provided with a threat flee away from and evacuate the space.

These agents are autonomous in that they rely solely on sensory information to guide them through the space. Path planning algorithms are implemented to dynamically alter an agent's path out of a space based on changes in the environment. In addition, agents are equipped with collision avoidance algorithms to ensure agents move through the space avoiding obstacles, moving around walls, and avoiding agent-agent collisions.

Using this framework we will be able to model realistic agent behavior and garner practical analysis on the flow of agents through the space, which will shed light on the space in terms of crowd density, the flow of agents through exits, and the ease at which agents evacuated and fled from the threat.

**Teaching fundamental CS concepts with unplugged activities** (*Allison Samson, Matthew Lang (Faculty Advisor)*)

In this presentation, we will discuss the design of a collection of unplugged activities as well as the result of piloting them with various age levels of Girl Scout troops.

The gender gap in Computer Science is a long-known and oft-studied problem. According to studies, many women do not feel that they are welcome in the field. Another problem is they feel the career is stigmatized by society. In order to combat these ideas, Moravian College Undergraduate Women have created a collection of outreach activities designed to encourage young students to consider computer science as a potential career and dismantle stereotypes about computer science. These activities, which are simple and fun games, require very little investment or set up and are appropriate for varying ages. Each illustrates a core concept in the discipline (e.g., problem solving, abstraction, optimization, etc.) and most require nothing more than pocket change.

Students have reported that they find the activities enjoyable and interesting, while troop leaders say that the girls have expressed excitement over the topics explored. So, not only have the activities affected students positively by exposing them to analytical thinking at a young age, but organizers have reported a resulting increased enthusiasm for computer science.

This presentation aims to encourage others to adopt similar programs. These activities are fun to organize, require little to no setup or equipment, create a huge impact, and are very rewarding.

**Enhancing the Small Team Development Process** (*Michael Waxmonsky, Alek Szilagyi, Ben Coleman (Faculty Advisor)*)

The jAmaseis seismology education software development project went through its fourth summer of development this year. Being behind schedule, the goal was to rework the development process to enhance our productivity and quality of output. We adopted several real world project tools such as scrum and agile to enhance time management and communication within the team. Through the application of these practices, we were able to work more directed and focused on specific, manageable goals, as well as more cohesive as a team of developers.

**Classifying Operators for Permutation-based Genetic Algorithms** (*Robert Cernera, Vincent Cicirello (Faculty Advisor)*)

Genetic algorithms model the genetic processes of living organisms at the cellular level. They are used to mimic the process of natural selection in order to find an optimized solution. In simple GA, we start with a population of “chromosomes,” usually represented as bit strings. Each chromosome is evaluated for its usefulness as a solution by a fitness function. The solutions with the most potential will be chosen to breed new offspring, which in turn eliminates a less optimal solution to maintain a relative population size. The altered population is then reevaluated and the process repeats until a sufficient solution is determined by the fitness function.

An alternate representation for GAs is that of the permutation, where individuals are represented by an ordering over a set of elements. Permutations often are a better fit to a problem, such as scheduling problems. A large variety of specialized mutation operators have been developed for permutations. Our work aims at providing guidance in choosing from among the many available mutation operators.

Fitness-distance Correlation is a measure of the relationship between the fitness of a GA individual and its “distance” to the optimal solution of the problem. It can be used to characterize the expected problem solving performance of GA. Additionally, mutation is meant to be a “small” change to an individual; which consequently depends on which distance measure is most relevant to the problem. There are many choices for distance measures, but are generally dictated by the problem trying to be solved. We have profiled the most popular mutators for a variety of distance measures. As a result, there are particular mutator choices that better map to the distance characteristics of a specific problem.

**An Agent-Based Approach to a Strongly Dynamic, Sequential Competition Knapsack Problem** (*Candice Schumann, Timothy Highley (Faculty Advisor)*)

We have developed a formal mathematical description of a card game called Fairy Tale. During the game, players draft cards, then play cards, and then score points based on the combinations of the cards they have played. Not all cards that are drafted are played, and the rules for the draft incorporate passing undrafted cards to other players. Together, these rules form the game that we characterize as a Strongly Dynamic, Sequential Competition Knapsack Problem (SDSCKP). This is an NP-hard problem, so a practical a solution to the SDSCKP could not be found. We created an implementation of the Fairy Tale Game and several algorithms for playing the game. Because this is the first presentation of algorithms for this Knapsack Problem variant, we do not have any other SDSCKP algorithms to use for comparison. However, comparisons among our own algorithms have yielded interesting insights. For example, in some cases a player’s particular algorithm choice is less important than simply having an algorithm that is different from the algorithms the other players are using. Additionally, we have demonstrated that even during the first decision of the game there is no static ranking for the value of the cards in the game. The presence or absence of cards in the opening draft pool affects the anticipated values of the other cards in the draft pool.

## **2012 Conference Committee**

### **Chair**

Vincent Cicirello, Richard Stockton College

### **Papers**

Aakash Taneja, Richard Stockton College

### **Panels, Workshops, and Tutorials**

Michelle Liu, Marymount University

Dave Hovemeyer, York College of Pennsylvania

### **Nifty Ideas & Lightning Talks**

Heather Amthauer, Frostburg State University

Michael Flinn, Frostburg State University

### **Student & Faculty Posters**

Timothy Highley, La Salle University

### **Undergraduate Programming Contest**

Michael Black, American University

### **Speakers**

Donna Schaeffer, Marymount University

Russ Manson, Richard Stockton College

### **Local Arrangements**

Duo Wei, Richard Stockton College

Demetrios Roubos, Richard Stockton College

### **Web Site & Social Networking**

John Wright, Juniata College



- Campus Center:** Welcome, Keynote, Poster Session, Reception, Banquet, Luncheon
- D-Wing:** Programming Competition, Saturday Breakfast, some Post-Conference Workshops
- G-Wing:** Most conference sessions
- Arts & Sciences Building:** 1 post-conference workshop (GIS Applications)
- Parking:** Use any of lots 1, 2, 3, 4, 5.