

# TEACHING COMPUTER SCIENCE & ENGINEERING THROUGH ROBOTICS: SCIENCE & ART FORM

**Andrea C. Burrows**

University of Wyoming, Laramie, Wyoming, USA  
E-mail: aburrow1@uwyo.edu

**Mike Borowczak**

University of Cincinnati, Cincinnati, Ohio, USA  
E-mail: mike@mborowczak.com

**Timothy F. Slater, J. Chris Haynes**

University of Wyoming, Laramie, Wyoming, USA  
E-mail: tslater@uwyo.edu, jhaynes4@uwyo.edu

## Abstract

*Is the field of computer science and engineering a strict discipline or an art form? The answer is both. Computer science and engineering concepts are typically found almost exclusively in collegiate engineering and technology programs. Given its importance across the 21<sup>st</sup> Century workforce, arguably these concepts should be taught at all pre-collegiate grade levels. This study, specifically focused on pre-collegiate teachers' increased confidence and subsequent use of robotics, indicates that designed professional development (PD) focused on simple computer programming approaches (e.g. LEGO MindStorm® kits) can propel pre-collegiate teachers to integrate new, challenging computer controlled robotics into their instruction. Surveys, content knowledge quizzes, and artifacts show teachers readily developed sufficient confidence and knowledge in producing lessons embedded with computer programming and robotics. Targeted classroom-ready instruction and modifiable computer programs appear to enhance pre-collegiate teacher knowledge of and confidence in robotics use. Considering overall self-rankings and content, pre-collegiate teacher pre-post scores increased. With these results, the authors argue why approaching computer science as both a strict science and art form is essential in PD.*

**Key words:** *computer science, education, engineering, robotics and teaching.*

## Introduction

Pre-collegiate teacher confidence has been established as a necessary condition to successfully deliver engaging science, technology, engineering, and mathematics (STEM) lessons (Cantrell, Young & Moore, 2003; Enochs & Riggs, 1990). In the scope of this work, a pre-collegiate teacher is defined as someone who instructs students who have not yet entered the university system. With computer science and engineering becoming an ever more common part of contemporary standards, teachers are under significant pressure to increase the amount of integrated STEM taught to students (NRC, 2012). As successful instruction in STEM is an important component in motivating students to pursue collegiate level studies in STEM, there exists an urgent need to rapidly upgrade the knowledge and confidence of teachers helping move students into collegiate STEM career pursuits. Our experiences, supported by research results presented later in this paper, confirm long held community suspicions that in a professional development (PD) workshop environment, PD leaders can highlight what initially appear to be

nuances of computer science and engineering as both a science and a vitally important creative endeavor - an art form - after pre-collegiate teacher confidence increases. Additionally, the authors view the education and learning of pre-collegiate teachers through a social constructivist lens where the pre-collegiate teachers engage in an active process, build meanings with each other, create products, and implement content. The field of computer science and engineering embraces artificial intelligence, computational biology, computer architecture, and graphics among many other domains, which are ubiquitous across 21<sup>st</sup> Century society. Our thesis is that pre-collegiate lessons involving computer science taught through engineering education design could enhance and promote broad-based student success across STEM classes. This is consistent with the long held notion that “engineering is an integrative process and thus engineering education... should be designed toward that end” (Bordogna, Fromm, & Ernst, 1993, p. 3).

### *Problem of Research*

Denning (2005) asserts that computer science “studies information processes both artificial and natural” (p. 28) whereas “programming, design, software/hardware engineering, building and validating models, and building user interfaces are all ‘computing arts’” (p. 29). But whether it is a science or an art form, few pre-collegiate teachers in STEM disciplines have had significant opportunities to learn the computer science and engineering knowledge base needed to create and implement contemporary lessons, which integrate STEM concepts for pre-collegiate students. As much of the current STEM workforce is being trained to be more flexible and proficient in computer programming and computer science, one could reasonably argue that school-based STEM should mirror these new focal points. While innately attractive for pre-collegiate teachers, the typical usage and assignment of the new generation of kit-based lessons (e.g. LEGO MindStorms®) often revolves around computer programming rather than a more traditional definition of computer science. The distinction between the two roughly parallels the usage of calculators to promote arithmetic over mathematics. This is similar to an aerospace engineer designing a new and untested airplane wing. In theory it might fly, but the aerospace designer does not have to fly a partially working plane in order to create the wing. In much the same way, a computer scientist can generate an algorithm or computer code that can solve a problem in theory, but a computer scientist does not have to work on a computer to produce successful code.

The impediment to creating and implementing novel, interesting, and applicable computer science (e.g. robotics) and engineering lessons for students is not the teachers use of computers, but rather their lack of confidence in tackling the broad knowledge base that currently exists in the field of computer science and engineering. To expose and examine this problem in the context of a teacher professional development workshop, the authors gave pre and post STEM confidence surveys, pre and post content quizzes, and collected artifacts to highlight a way for teachers to focus on computer programming and computer science to better accomplish integrated STEM lesson development. The answer to this research problem can directly inform how PD programs can be most efficient in upgrading teachers’ knowledge, skills, and confidence in this rapidly growing domain.

It can be argued that the field of computer science and engineering has a growing identity issue among STEM disciplines. In many circles it is considered a pure subject with its own disciplinary structure and hierarchy, like traditional biology, chemistry and physics – in others computer science is considered applied subject matter in lines with other engineering disciplines. Regardless, a “programming-first” approach is used most often when teaching students computer science across North America (Cooper, Dann, & Pausch, 2003). In contrast, the authors in this study used a primary hands-on, trial and error method, also known as bricolage (Ben-Ari, 2001), to expose beginners to computer programming using off the shelf robotics kits.

Recommendations by Bordogna, Fromm, and Ernst (1993) to integrate engineering include: (a) focusing on the broad educational experience where individual concepts are connected and integrated, and (b) viewing students as professionals to make engineering more attractive, exciting, and fulfilling so they connect with the materials. Furthermore, to enhance the preparation of students to engage in engineering, there is a trend towards increasing the design component in the curricula (Dutson, Todd, Magleby, & Sorensen, 1997). However, “design thinking is complex” (Dym, Agogino, Ozgur, Frey, & Leifer, 2005, p. 103). Yet there is hope for the students of today and tomorrow. Pre-collegiate teachers can increase the numbers of students interested in STEM and steer students towards STEM careers. Pre-collegiate teachers now have access to the background and research to bring STEM and thus engineering design with computer science programming into their classes. As a generalization, “above all, recognize that more effort needs to be expended on strategies to promote the adoption and implementation of STEM reforms...” (Fairweather, 2008, p. 28). The authors propose that lesson plans with strong engineering and computer science content will promote pre-collegiate teacher use of computer science/engineering lesson development as well as increase confidence in using computer science and engineering.

The authors used a mixed method methodology, containing quantitative surveys (pre and post for confidence and content during an intensive, two-week, summer PD intervention) and qualitative artifacts, and investigated if pre-collegiate teacher confidence in STEM, in particular computer science and engineering, could be increased through instruction in specific content where pre-collegiate teachers utilized robotics kits as a connection to engineering design problems. The authors administered pre-collegiate teacher pre and posttest content quizzes on a daily basis to track the increase in STEM knowledge as well as collected lesson plan artifacts to examine for the content presented. With a focus on pre-collegiate teacher confidence, the authors pursued the following research question: *Does confidence level of pre-collegiate teachers impact their ability to create and implement lessons that use computer programming and robotics kits as a mechanism to introduce computer science and engineering in classroom instruction?*

## **Methodology of Research**

### *General Background of Research*

In the 2012 summer, 13 pre-collegiate teachers (five elementary, five middle, and three high) met in a rural western city of the United States for two weeks of intensive PD. The daily eight-hour PD intervention included pre-collegiate teacher content refresher sessions as well as laboratory time (Figure 1). First, the content sessions included the following general topics: astronomy/data encoding, technology/reactions, mathematics/web software, science/GPS, agriculture/pH, engineering/water quality, engineering/structures (e.g. aviation), engineering/biomedicine. The PD facilitator experts (the authors of this paper) highlighted STEM integration, real-world applications, career connections, and societal impacts for computer science and engineering during each of the content sessions. Second, the laboratory sessions involved the pre-collegiate teachers building off-the-shelf robotics kits (e.g. LEGO MindStorms®) and then programming the robots on teacher self-determined novice, intermediate, and/or expert levels. Examples of tasks included making the robot perform the following functions: move, loop, sense light/sound, track a line, turn around, navigate a maze, and problem solving (e.g. hitting a wall). The PD facilitator experts were available during laboratory time to assist with programming.

The facilitators encouraged program development without providing the answers to the pre-collegiate teacher programmers. Consequently, the pre-collegiate teachers engaged in content and programming explanation, discovery, analysis, invention, synthesis and construction which fall into both science and art form categories according to Denning (2005).

<b>Day: Topic</b>	<b>Content Based Activity</b>	<b>Laboratory Computing Connections</b>	
		Strict Science	"Art form"
<b>W1D1: Data Encoding</b>	Astronomy & Morse Code	Communication & Cryptography	Toy Sonar
<b>W1D2: Reactions</b>	Batteries	Data Storage, Logic	Feedback Loops
<b>W1D3: Web Software</b>	GeoGebra: Interactive Graph Paper	Parallel Computing	Autonomous Agents
<b>W1D4: GPS</b>	Geocaching	Precision, Round off Errors	SWARMS Multi-agent System
<b>W1D5: pH</b>	Senor Principles and Usage	Fundamentals of Integrated Circuits	Sensor Development
<b>W2D1: Water Quality</b>	Water Filtration	Hierarchy - Top Down Design	Remote Exploration
<b>W2D2: Structures</b>	Bio-inspired Physical Structures	System Architecture	Genetic Algorithms & Biomimicry
<b>W2D3: Biomedicine</b>	Heart Blockage Removal Tool Design	Imaging, Large Data	Cross-disciplinary Development
<b>W2D4-5: Showcase</b>	Presentation of Self-Developed Lessons	Communication to Novice Audiences	

**Figure 1: Daily professional development content and laboratory sessions.**

### *Instrument and Procedures*

In order to assess the confidence level of pre-collegiate teachers and their ability to create and implement lessons that use computer programming and robotics kits, the authors collected three data sets. First, a pre and post STEM confidence survey was administered at the beginning and ending of the two-week PD intervention. Second, a daily pretest and posttest was given to the pre-collegiate teachers to determine their STEM content knowledge. Finally, the authors collected artifacts that included lesson plans and completed robotic programs during the two-week session. These artifacts are most easily characterized as teacher created material that fell into three themes, which along with the other two instruments are detailed in subsequent sections.

### *Data Analysis*

Pre PD intervention and post PD intervention STEM survey data (seven questions) contained 13 individual responses. Of the seven questions, one focused on self-reported confidence, four measured perceived self-confidence, and two focused on perceived impact to student ability. For the pre and post STEM confidence survey data a mean was tabulated with standard deviations. A set of t-test comparisons determined strong statistical significance (99% confidence) in self-reported pre-collegiate teacher confidence, while the remaining six pre-post data sets showed slightly weaker statistical significance (85%-90% confidence).

Pre and post STEM content knowledge quizzes (approximately five content and three pedagogy questions per quiz) contained 13 individual responses on a daily basis. A mean was tabulated with standard deviations. A t-test shows with greater than 99% confidence that the two pre and posttest content knowledge quiz scores data sets are statistically different. Although the number of participants was small, the number of data points compared was large and the apparent relative normal distribution among the responses suggests that a t-test is a reasonable measure, particularly when used as part of a triangulation with other artifacts. Artifacts contained at least 13 individual lesson plans and final LEGO MindStorm® robotic programs. Some of the pre-collegiate teachers wrote several lessons and several final robotic programs. In relation to the artifacts (lesson plans and robotic programs) themes were inductively generated based on repetition of words for the lesson plans and success of the program for the robots. This approach, used in grounded theory, is consistent with the analysis methods advocated in the seminal work by Glaser and Strauss (1967). Emerging major lesson plan themes included: STEM integration, engineering and design, and encoding.

### Results of Research

Pre STEM confidence survey data (Figure 2) shows that nine of the 13 (69%) pre-collegiate teachers ranked themselves as a novice and four of the 13 (31%) ranked themselves as an intermediate in STEM and computer programming. None of the teachers ranked themselves as an expert. After the two-week PD, eight of the 13 (62%) pre-collegiate teachers ranked themselves as an intermediate and five of the 13 (38%) ranked themselves as experts in STEM and computer programming within the LEGO MindStorm® platform. It is worth pointing out that, at the start of the PD, the elementary and high school teachers (data points E1-E5 and data points H1-H3, respectively) tended to rank themselves as novices, whereas the middle grades teachers (data points M1-M5) ranked themselves between novice and intermediate, but after the two weeks there were more self-reported experts in the elementary and high school teacher groups. The t-test performed between the data sets shows them as statically different ( $p < 0.001$ ).

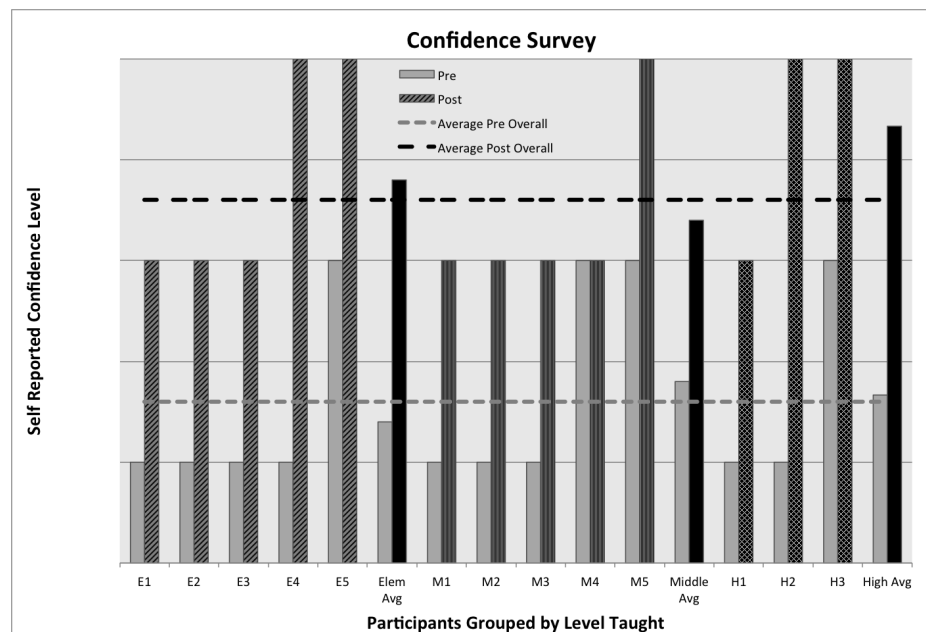
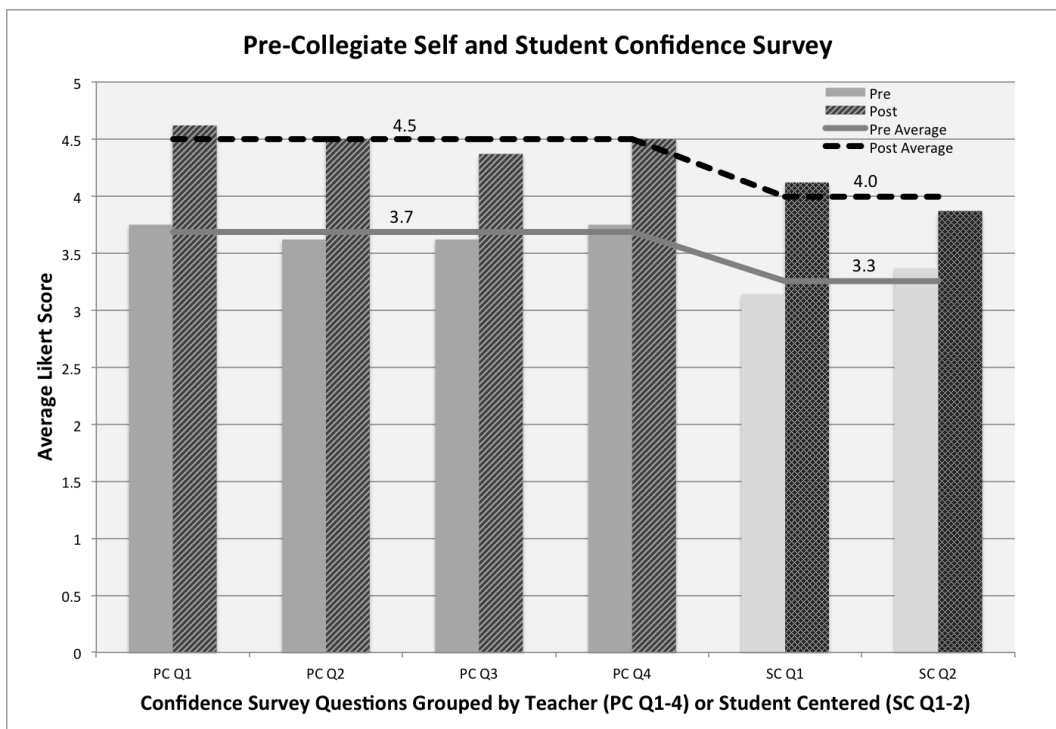


Figure 2: STEM self-reported confidence pre and post-survey data.

While the self-reported confidence data (the first survey question) shows statistically significant results, the Likert-based STEM confidence results (Figure 3) are not as clear ( $p < 0.15$ ). Of the other six Likert-based questions answered by the teachers, four focused on their own confidence with particular STEM concepts while two focused on their perception of their students' abilities. A consistent half point bias exists between the teachers' confidence level and their perception of their students' abilities in both the pre and post-survey data.



**Figure 3: STEM pre-collegiate self (PC) and student centered (SC) confidence.**

Pre and post STEM content knowledge quiz data (Figure 4) show that there was significant improvement each day as based on t-test values where  $p < 0.001$  (with one exception: data encoding having a  $p < 0.015$ ). The content with the greatest percent increase (117%) was pH, followed by structures 91%, and then water quality at 83%. The least percent increase (17%) dealt with biomedicine content. The two-week average pretest scores were 57% whereas the average posttest scores were 88%.

Artifacts including lesson plans (Table 1) show that three major themes emerged including: STEM integration, engineering and design, and encoding. Robotic programs implemented on the LEGO MindStorm® platform by the pre-collegiate teachers accomplished the goals of each level (novice, intermediate, expert). The artifacts consistently reveal that the pre-collegiate teachers first learned by using facilitator guidance through trial and error (also called bricolage) and then moving into the logic used by intermediates and experts, although trial and error was still a part of their learning process.

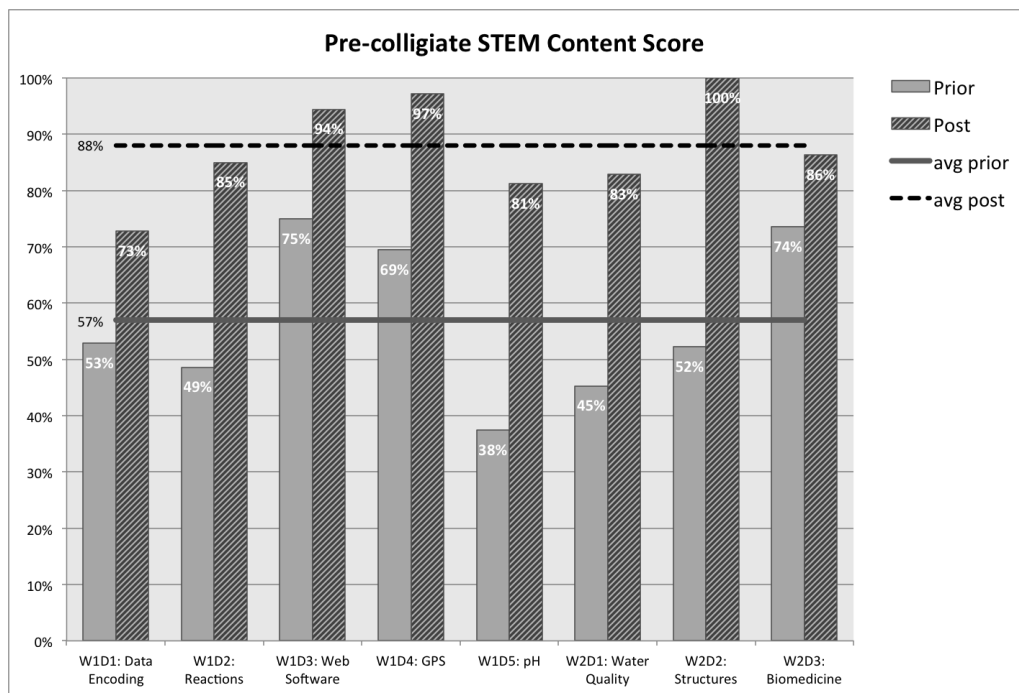


Figure 4: STEM content knowledge quiz data showing week (W) and day (D).

Table 1. Lesson plan topics divided into themes (artifact data).

Participant(s)	Grade Level Taught	STEM Content Integration	Engineering Design	Encoding
Female-Female	Elem.	GPS Insect Hunt	Balance and Motion Structure	Arithmetic and Secret Messages
Female-Female	Elem.	Treasure Hunt, Plant Variables	Creating a Filter	English Literacy through Encoding
Female	Middle	Making Tracks	All Shook Up	Can You Hear Me?
Female	Middle	Creek Bed Sample Collection	Senior Citizen Solutions	-
Female	Middle	Patterns of Change in Clouds	-	Crack the Code to Solve the Mystery
Male-Female	Middle	Which Direction?	Filters for Backpackers	Codes and Digital Literacy
Male	Middle	CO2 Bottle Cars	Bridge Building, Model Rockets	-
Male	Middle	Weather Patterns	What's the Soil Quality Like Here?	Animal Communication
Female	High	Geology with Interactive Maps	Salt Water to Fresh Water	-
Female	High	-	Water Quality	-
Male	High	-	-	Flame Test through Decoding
Lesson Count		10	10	7

Brief definitions and examples highlight the three main uncovered themes of STEM content integration, engineering design, and encoding. To the authors of this paper, STEM content integration implies the linkage of two to four STEM components in a created lesson. For example, *Making Tracks*, a middle level lesson employs mathematics through spatial reasoning, investigates science through ecology and geography, and showcases technology through GPS and web based tools. To meet the criteria for the second theme of engineering design, a lesson must include a complete design process from planning, through prototype building, and iterations of testing. One lesson for high school students, *Salt Water to Fresh Water*, includes science as inquiry where students identify a water problem, conceptualize a solution, design a prototype, build a prototype, and then iteratively test and refine a solution for creating potable drinking water from salt water. Lastly for the authors, encoding is representing information in another mode. *Arithmetic and Secret Messages*, an exemplar lesson from the encoding theme, captivates elementary students by having them solve basic operations by encoding a message

for another student and then decoding one that they are given. The authors categorized all of the lessons that the pre-collegiate teachers created using the constructed definitions above. These lessons are available at <http://www.uwrobotics.com>.

## Discussion

Taken together, these results illustrate that teacher confidence and knowledge can be enhanced, even when they have little to no formal collegiate experience in these domains. *Does confidence level of pre-collegiate teachers impact their ability to create and implement lessons that use computer programming and robotics kits as a mechanism to introduce computer science and engineering in classroom instruction?* Simply put, yes it does. As such, this data lends weight to the ongoing plea to policy makers to provide continued opportunities for high quality PD for pre-collegiate teachers, particularly in areas that will be new in the *Next Generation Science Standards*, which are based on the *Framework for K-12 Science Education* (NRC, 2012). This research motivates us to pursue more deeply, questions surrounding supporting teachers' understanding and implementation of computer science concepts. The recurring core computer science concepts found in this study are STEM content integration, engineering design, and encoding.

To inform future pre-collegiate PD programs involving computer science concepts, the teacher education community needs to consider systematic analysis of the following: (a) gender in these PD settings, (b) elementary versus high school teacher confidence and content, (c) elementary versus high school teacher lesson implementation, (d) amount of teacher experience with computer science versus confidence and implementation of lessons, and (e) a longitudinal study of these pre-collegiate teachers after a PD like the one described. A clearer distinction between computer science and computer programming in education should also be highlighted. Another interesting topic to explore was exposed earlier in the confidence section; elementary and high school teachers tended to rank themselves as novices (with STEM and computer programming) in the beginning whereas the middle grades teachers ranked themselves as intermediate, but after the two weeks there were more self-reported experts in the elementary and high school teacher groups. Collecting and analyzing data from specific PD experiences that incorporate computer science would allow researchers to build upon the best practices that have already been established such as focusing on content, promoting active learning, fostering coherence with other learning activities, planning time for implementation, and providing technical support (Garet, Porter, Desimone, Birman, & Yoon, 2001; Penuel, Fishman, Yamaguchi, & Gallagher, 2007). Is computer science a strict science or an art form? It is both, and "its ultimate significance has little to do with computers" (Denning, 2005, p. 29). This is consistent with what Ben-Ari (2001) states, "Teaching *how* to do a task can be successful initially, but eventually this knowledge will not be sufficient" (p. 48). The science of computer science must be blended with the art form of creativity.

## Conclusions

The literature clearly suggests that computer science is in the midst of an identity crisis. The subject has two distinct ways to advance. One is through strict instruction of the theory and the other is trial and error innovation. The thesis, related earlier in this paper, stated that pre-collegiate lessons involving computer science taught through engineering education design could enhance and promote broad-based student success throughout STEM classes. The results presented in this paper provide evidence that this can be accomplished. We know that engineering is a process that integrates STEM subjects, and thus instructors should approach teaching computer science, which is based both on technique and inventiveness, as a combination of



both strict science and an art form. One without the other is not sufficient. Instructors using this integrated approach could enhance teachers' knowledge, skills, and confidence in computer science. Incorporating real-world applications, like the ones presented in this paper, treating pre-collegiate teachers as professionals, and offering ways for pre-collegiate teachers to practice both the science and art form of computer science could create fertile ground for students in regards to future computer science exploration. Each piece of the knowledge puzzle that includes content, applicability, perceptions, and practice are essential. The authors asked if confidence level of pre-collegiate teachers impact their ability to create and implement lessons that use computer programming and robotics kits as a mechanism to introduce computer science and engineering in classroom instruction. Yes confidence matters, but practicing computer science as a technical domain and an art form does as well.

Computer science is based on hierarchical principles as much as it is creative trial and error practice. It is possible to lead pre-collegiate teachers through the fundamental processes of the science and art of computer science. With increased confidence and content knowledge the pre-collegiate teachers are more likely to create and implement computer science lessons with their students. Thus, the pre-collegiate teachers should be able to create higher quality lessons with a higher level of computer science complexity to engage their students.

### Acknowledgements

The authors would like to acknowledge that this PD was funded by a federal grant under No Child Left Behind (NCLB) [P.L. 107-110, Title II, Part B] and administered by the Wyoming Department of Education. Additional funding came from the Wyoming's Excellence in Higher Education Endowed Chair Endowment. Special appreciation goes to Richard Sanchez of Johnson County School District #1 in Buffalo, Wyoming and Donna Governor from the University of Georgia in Athens, Georgia for their support of this research.

### References

- Ben-Ari, M. (2001). Constructivism in computer science education. *Journal of Computers in Mathematics and Science Teaching*, 20 (1), 45-73.
- Bordogna, J., Fromm, E., & Ernst, E. (1993). Engineering education: Innovation through integration. *Journal of Engineering Education*, 82 (1), 3-8.
- Cantrell, P., Young, S., & Moore, A., (2003). Factors affecting science teaching efficacy of pre-service elementary teachers. *Journal of Science Teacher Education*, 14, 177-192.
- Cooper, S., Dann, W., & Pausch, R. (2003). Teaching objects-first in introductory computer science. *Proceedings of the 34th SIGCSE technical symposium on Computer Science Education*. New York: New York, 191-195.
- Denning, P. (2005). Is computer science science? *Communications of the ACM*, 48 (4), 27-31.
- Dutson, A., Todd, R., Magleby, S., & Sorensen, C. (1997). A review of literature on teaching engineering design through project-oriented capstone courses. *Journal of Engineering Education*, 86 (1), 17-28.
- Dym, C., Agogino, A., Ozgur, E., Frey, D., & Leifer, L. (2005). Engineering design thinking, teaching, and learning. *Journal of Engineering Education*, 94 (1), 103-120.
- Enochs, L., & Riggs, I. (1990). Further development of an elementary science teaching efficacy belief instrument: A preservice elementary scale. *School Science and Mathematics*, 90, 694-706.
- Fairweather, J. (2008). Linking evidence and promising practices in science, technology, engineering, and mathematics (STEM) undergraduate education. *Status Report for the National Academies National Research Council Board of Science Education*. Retrieved from [http://www7.nationalacademies.org/bose/Fairweather\\_CommissionedPaper.pdf](http://www7.nationalacademies.org/bose/Fairweather_CommissionedPaper.pdf)
- Garet, M., Porter, A., Desimone, L., Birman, B., & Yoon, K. (2001). What makes professional development effective? Results from a national sample of teachers. *American Educational Research Journal*, 38 (4), 915-945.

- Glaser, B., & Strauss, A. (1967). *The discovery of grounded theory: Strategies for qualitative research*. New Brunswick, NJ: Aldine Transaction Press.
- National Research Council. (2012). *A framework for K-12 science education: Practice, crosscutting concepts, and core ideas*. Washington, DC: The National Academies Press.
- Penuel, W., Fishman, B., Yamaguchi, R., & Gallagher, L. (2007). What makes professional development effective? Strategies that foster curriculum implementation. *American Educational Research Journal*, 44 (4), 921-958.

*Advised by Boris Aberšek, University of Maribor, Slovenia*

Received: September 29, 2012

Accepted: November 11, 2012

<b>Andrea C. Burrows</b>	Ed.D. Curriculum and Instruction with Science Specialization, Assistant Professor of Secondary Science Education, University of Wyoming 1000 E. University Ave., Dept. 3374, Laramie, WY, 82071, USA. E-mail: <a href="mailto:aburrow1@uwo.edu">aburrow1@uwo.edu</a> Website: <a href="http://www.uwo.edu/seced/faculty-staff/andrea-burrows.html">http://www.uwo.edu/seced/faculty-staff/andrea-burrows.html</a>
<b>Mike Borowczak</b>	Ph.D. Candidate Computer Science and Engineering, Computer Science and Engineering Graduate Research Assistant, University of Cincinnati, P.O. Box 19233, Cincinnati, OH 45219, USA. E-mail: <a href="mailto:mike@mborowczak.com">mike@mborowczak.com</a> Website: <a href="http://secs.ceas.uc.edu/~borowcm">http://secs.ceas.uc.edu/~borowcm</a>
<b>Timothy F. Slater</b>	Ph.D. GeoSciences, Endowed Chair of Science Education, University of Wyoming, 1000 E. University Ave., Dept. 334, Laramie, WY, 82071, USA. E-mail: <a href="mailto:tslater@uwo.edu">tslater@uwo.edu</a> Website: <a href="http://www.uwo.edu/seced/faculty-staff/timothy-slater.html">http://www.uwo.edu/seced/faculty-staff/timothy-slater.html</a>
<b>J. Chris Haynes</b>	Ph.D. Agricultural Education, Assistant Professor of Agricultural Education, University of Wyoming, 1000 E. University Ave., Dept. 3374, Laramie, WY, 82071, USA. E-mail: <a href="mailto:jhaynes4@uwo.edu">jhaynes4@uwo.edu</a> Website: <a href="http://www.uwo.edu/seced/faculty-staff/chris-haynes.html">http://www.uwo.edu/seced/faculty-staff/chris-haynes.html</a>