Outcomes of Bringing Computational Thinking into STEM Classrooms

David Weintrop, Kai Orton, Michael Horn, Elham Beheshti, Laura Trouille,
Kemi Jona & Uri Wilensky
Northwestern University

Abstract: *Computation is changing the landscape of modern STEM disciplines, but relatively few students take courses that adequately prepare them for the increasingly technological nature of these fields during high school. Further, the students who do study computational topics tend to not reflect the greater student body, with female and minority students being disproportionately underrepresented. To address this issue, we propose embedding computational thinking content into STEM coursework. This paper presents initial findings from the project. Preliminary analysis shows that despite significant gaps in attitudes and confidence between male and female students, no difference in aptitude between genders was found. Additionally, female students who participated in the project showed improved confidence with computational thinking and interest in STEM fields.*

Keywords: Computational Thinking; STEM; Computational Science Education

Introduction

Computer science education in the United States faces two interrelated challenges. First, there is significant concern that our education system is not producing enough people with computational thinking skills to fill both current and projected industry demand (National Governors Association, 2007; National Research Council [NRC], 2011; President's Council of Advisors on Science and Technology, 2010). Second, more than any other major STEM discipline, women and minorities are significantly underrepresented in the educational pipeline and in the workforce (Camp, 1997; Ericson & Guzdial, 2014; Hill, Corbett, & St Rose, 2010; Margolis & Fisher, 2003; Zweben & Bizot, 2012). According to the National Science Foundation's National Center for Science and Engineering Statistics (2013), female undergraduates enrollment has remained below 30% since 1991. Despite numerous ongoing college, university, and national campaigns targeting women, the numbers continue to drop (National Science Board, 2012). Furthermore, the attrition rate among women who major in CS remains high compared to male counterparts (Klawe and Levenson 1995). Among the reasons for these trends, researchers have identified a lack of interest and confidence (American Association of University Women [AAUW], 1994; Margolis, Fisher, & Miller, 2000), limited visibility of positive role models (Scragg & Smith, 1998; Townsend, 2002), and lack of positive experiences with both computer science and in STEM fields more broadly (AAUW, 1994; Kramer & Lehman, 1990; Miliszewska, Barker, Henderson, & Sztendur, 2006; Schenkel 1984).

Fortunately, there are a number of promising efforts underway to address these challenges. These efforts include projects intended to greatly expand student exposure to computer science and programming (code.org, CE21, CS10K, etc.); pending legislation that would allow computer science to count as a core scientific discipline; and efforts to make computer science a requirement for high school graduation. These efforts are primarily focused on developing a new sequence of elective CS courses leading to a redesigned Advanced Placement (AP) CS course.

The success of a separate elective CS course sequence, even if made more appealing, rests on several key assumptions. The first is that stand-alone course content will be appealing enough to attract a diverse range of students. Otherwise there is a danger that new course offerings will perpetuate the existing gap for underrepresented groups. The second assumption is that qualified teachers for these new courses can be found or trained and retained. Currently, only 53% of states support the minimum CS teacher certifications for some level of teaching, with only 13 of these requiring endorsements for teaching at the secondary level (Khoury, 2007). Retention of teachers is of particular concern due to a surplus of high-paying jobs in the tech industry. Finally, there is an assumption that students will have room in their already-full high school schedules to take elective CS courses.

There is an equally pressing need to train future scientists, engineers, and mathematicians who understand and make use of computational thinking (CT) to address complex and increasingly data-driven challenges. In addition, we take seriously the goal of preparing an educated citizenry capable of participating and contributing in an increasingly computational world.

In this paper we present preliminary findings from our three-year study supporting math, chemistry, biology, and physics teachers in embedding computational thinking enhanced lesson plans into their curriculum. This method directly addresses pursing an embedded strategy that brings CT into classrooms that addresses the challenges mentioned above. Instead of introducing stand-alone computing courses to schools, we bring CT into existing math and science classrooms. Through developing CT enhanced activities that fit within existing STEM classes, and training in-services STEM teachers to teach these lessons, we can draw on the existing teacher workforce and provide computing instruction to all students through the courses they are already taking. Below we briefly present the theoretical framework that motivates this work, the study design that accompanies this project, and the preliminary findings from our three-year project, highlighting our successes and the challenges we still face moving forward.

Why Bring Computational Thinking to STEM Classrooms

A primary motivation for introducing CT practices into science and mathematics classrooms is in response to the increasingly computational nature of STEM disciplines as they are practiced in the professional world (Foster, 2006). Computation is now an indispensable component of STEM disciplines (Henderson, Cortina, & Wing, 2007, p. 195). This rise in importance of computation with respect to the STEM fields has been recognized both by those within the STEM education communities and computer science education organizations (ACM/IEEE-CS Joint Task Force on Computing Curricula, 2013). Bringing computational tools and practices into STEM classrooms gives learners a more realistic view of what STEM fields are and better prepares students for STEM careers (Augustine, 2005; Gardner, 1983).

Preparing students for the modern STEM landscape is not the only reason to bring CT into STEM classrooms. From a pedagogical perspective, the thoughtful use of computational tools and skillsets can deepen learning of STEM content (Guzdial, 1994; National Research Council, 2011; Repenning, Webb, & Ioannidou, 2010; Sengupta, Kinnebrew, Basu, Biswas, & Clark, 2013; Wilensky & Reisman, 2006). The reverse is also true – namely, that science and mathematics provides a meaningful context (and set of problems) within which CT can be applied. This differs markedly from teaching CT as part of a standalone course where the assignments tend to be divorced from real-world problems and applications. This sense of real-world applicability has been deemed important in the effort to motivate diverse participation in computational professions (Margolis & Fisher, 2002). This reciprocal relationship—using computation to enrich

STEM learning and using STEM to enrich computational learning—is at the heart of our motivation to bring CT and STEM together.

A final motivation for bringing CT into STEM classrooms is to reach the widest possible audience and address longstanding issues of the underrepresentation of women and minorities in computational fields discussed above. Currently, only a fraction of high school students have the opportunity to take a computer science course due to a lack of qualified teachers, inadequate facilities, or a lack of student interest. Embedding CT activities in STEM coursework directly addresses the issue of students self-selecting into (or out of) computer science classes, which has been a challenge long plaguing the effort to reach underserved youth (Margolis & Fisher, 2003; Margolis, 2008). It also avoids practical issues of fitting new classes into overcrowded schedules and finding teachers to teach them.

Methods and Data Sources

Our work bringing CT into STEM classes involved three main efforts: 1) defining CT in STEM; 2) developing CT enhanced STEM lessons; and 3) creating online assessments to evaluate the success of our interventions. In the first year of our grant we developed a CT in STEM practices taxonomy that outlines a set of 22 skills across four categories that serves as a structured definition of CT in STEM (Weintrop et al., in preparation). Over the first two years of the study we developed a portfolio of 22 classroom-tested CT activities for high school biology, chemistry, physics, and mathematics classrooms. All of these activities are aligned with our CT-STEM practices taxonomy and mapped onto relevant Common Core, CS Curricula and the Next Generation Science Standards. Each CT-STEM lesson plan includes classroom materials and a teacher guide and requires 2-3 class periods[1]. Finally, to assess student learning, we created five CT in STEM assessment sets (Weintrop et al., 2014) that were administered by participating teachers immediately after teaching a CT-STEM lesson plan.

The data we present in this paper were collected during the 2013-2014 school year at 11 participating schools in a Midwestern city. Over the course of the project we had 58 teachers attend our professional development workshops from 38 schools. The data presented below are from teachers who participated in the third year of the project. The student attitudes and perceptions data we present are drawn from attitudinal surveys that were administered to students in participating classrooms at the beginning and the end of the school year. A total of 704 attitudinal surveys were completed (475 pre and 229 post) with 49.7% of the surveys being filled out by female students. These surveys included a total of 162 questions focusing on students' attitudes towards CT and STEM, their confidence with the subjects, and their interest in fields related to computation. The survey primarily used a 5-point Likert scale and asked students to respond to statements

---

[1] Visit the project website for more information: http://ct-stem.northwestern.edu

such as “I feel comfortable working with computers” and “I am interested in pursuing a career in engineering.” We also report a preliminary analysis of the student response to our CT-STEM assessment sets. A total of 1,022 assessment sets were completed by 549 students during the 2013-2014 school year.

Preliminary Findings

One of our motivations for embedding CT in STEM is to address issues of students self-selecting into or out of elective computer science courses. As a result of our approach, all students enrolled in conventional science and math classes are exposed to CT, thus addressing issues of low numbers of female and minority students taking computer science. Of the 549 students who took one of our assessment sets during class, 49% (271) self-identified as Hispanic, 37% (203) as African American, 15% (83) as white, and 10% (53) as Asian[2]. Of this same sample, 52% were male while 48% were female. The diversity of students taking our assessments and the equality with respect to the gender of students suggests our approach of bringing CT into STEM classes is an effective way to introduce a broad and diverse set of students to CT.

Comparing the responses given on the pre survey between male and female students, we see disparities that match those reported in other studies on gender and STEM and computer science fields. Female students were significantly less interested in the STEM fields, felt CT was less important, and reported being less comfortable with computers than their male counterparts. When asked about interest in possible future professions, female students were significantly less interested in careers in computational sciences, engineering, mathematics, and computer science (see Appendix 1). Finally, female students were less confident in all 20 questions pertaining to CT in STEM contexts (see Appendix 2).

As part of this study, students were given CT in STEM assessments after working through a lesson plan designed as part of the project. A preliminary analysis of student responses shows no significant difference in performance between students based on gender. Looking at the subset of responses to our General CT in STEM assessment set that can be automatically scored, we see that the 161 female had an average score of 2.21 out of 5, while the 192 male students had an average score of 2.27 out of 5, a difference that is not statistically significant $t(352) = .377$, $p < .706$. The fact that female students at the start of the year saw themselves as less confident in computational fields as well as less interested in pursuing careers in computational fields, paired with findings that show female students have the same level of aptitude, highlights the importance and potential of integrating CT with mandatory course work. If the content were reserved for an

---

[2] Please note, the ethnicity field was presented as check-boxes to the students, so it was possible to select multiple ethnicities, thus these numbers sum to greater than 100%.

elective course, these students would likely self-select out of computational learning opportunities if given the chance.

At the end of the school year, we administered our attitudinal survey a second time to see if students being exposed to our CT in STEM activities improved their perceptions of and attitudes towards CT. Responses in the post-test show significant gains on questions relating to interest in pursing careers in science $t(349) = 2.018$, $p < .044$, enjoyment related to using computational tools for schoolwork $t(439) = 2.905$, $p < .004$ and the learning benefits of doing so $t(349)=2.531$, $p < .012$. Most importantly, girls showed positive gains on 19 of the 20 questions pertaining to confidence in CT in STEM questions (though not always significant – see Appendix 2). This shift highlights the effectiveness of CT learning experiences situated within STEM for female students.

Conclusion

The finding of no difference in CT aptitude between girls and boys, but large differences in attitudes towards and perceptions of CT suggests that reserving CT instruction to opt-in, elective courses will perpetuate the issue of under-representation of girls in CT. Our approach of brining CT into STEM classrooms is designed to provide hands-on, empowering CT experiences for all students. In this short paper we have shown preliminary results showing the effectiveness of this approach. In the full version of this paper, we will present a more detailed analysis of the data presented in order to more fully defend the claims we make about the positive effects of bringing CT in STEM classrooms and how the approach broadens participation and changes attitudes of students historically underrepresented in both STEM and computational fields.

References

ACM/IEEE-CS Joint Task Force on Computing Curricula. (2013). *Computer Science Curricula 2013*. ACM Press and IEEE Computer Society Press. Retrieved from http://dx.doi.org/10.1145/2534860

American Association of University Women. (1994). *Shortchanging Girls, Shortchanging America*. Washington, DC: AAUW Educational Foundation.

Augustine, N. R. (chair). Rising above the gathering storm: Energizing and employing America for a brighter economic future (2005). Washington D.C.: National Academies Press.

Camp, T. (1997). The incredible shrinking pipeline. *Communications of the ACM*, *40*(10), 103–110.

Ericson, B., & Guzdial, M. (2014). Measuring Demographics and Performance in Computer Science Education at a Nationwide Scale Using AP CS Data. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (pp. 217–222). New York, NY, USA: ACM. doi:10.1145/2538862.2538918

Foster, I. (2006). 2020 Computing: A two-way street to science's future. *Nature*, *440*(7083), 419–419.

Gardner, D. P. (Chair). (1983). *A nation at risk: The imperative for educational reform*. Washington, DC: U.S. Department of Education.

Guzdial, M. (1994). Software-realized scaffolding to facilitate programming for science learning. *Interactive Learning Environments*, *4*(1), 001–044. doi:10.1080/1049482940040101

Henderson, P. B., Cortina, T. J., & Wing, J. M. (2007). Computational thinking. In *ACM SIGCSE Bulletin* (Vol. 39, pp. 195–196). ACM. Retrieved from http://dl.acm.org/citation.cfm?id=1227378

Hill, C., Corbett, C., & St Rose, A. (2010). *Why So Few? Women in Science, Technology, Engineering, and Mathematics.* Washington, D.C.: American Association of University Women.

Khoury, Ghada. (2007) Computer Science State Certificaiton Requirements CSTA Cerification Committee Report. Compuster Science Teachers Association. Retrieved from https://csta.acm.org/ComputerScienceTeacherCertification/sub/TeachCertRept07New.pdf

Klawe, M., & Leveson, N. (1995, January). Women in computing: where are we now? *Communications of the ACM*, *38*(1), 29+.

Kramer, P. E., & Lehman, S. (1990). Mismeasuring Women: A Critique of Research on Computer Ability and Avoidance. *Signs*, *16*(1), 158–172.

Margolis, J. (2008). *Stuck in the shallow end: Education, race, and computing*. The MIT Press.

Margolis, J., & Fisher, A. (2003). *Unlocking the clubhouse: Women in computing*. The MIT Press.

Margolis, J., Fisher, A., & Miller, F. (2000). The anatomy of interest: Women in undergraduate computer science. *Women's Studies Quarterly*, 104–127.

Miliszewska, I., Barker, G., Henderson, F., & Sztendur, E. (2006). The Issue of Gender Equity in Computer Science–What Students Say. *Journal of Information Technology Education*, *5*.

National Governors Association [NGA]. (2007). *Innovation America: A final report.* Washington, DC. Available at: http://www.nga.org/Files/pdf/0707innovationfinal.pdf.

National Research Council [NRC]. (2011). *Successful K-12 STEM Education: Identifying Effective Approaches in Science, Technology, Engineering, and Mathematics*. Washington, DC: The National Academies Press.

National Science Board [NSB]. 2012. *Science and Engineering Indicators 2012*. Arlington VA: National Science Foundation (NSB 12-01).

President's Council of Advisors on Science and Technology. (2010). *Prepare and inspire: K-12 education in science, technology, engineering, and math (STEM) for America's future.* Washington, DC: Author. Available at: http:// www.whitehouse.gov/sites/default/files/microsites/ ostp/pcast-stem-ed-final.pdf.

Repenning, A., Webb, D., & Ioannidou, A. (2010). Scalable game design and the development of a checklist for getting computational thinking into public schools. In *Proceedings of the 41st ACM technical symposium on Computer science education* (pp. 265–269).

Schenkel, Susan. 1984. Giving away success: Why women "get stuck" and what to do about it. New York: McGraw-Hill.

Scragg, G., & Smith, J. (1998). A study of barriers to women in undergraduate computer science. In *ACM SIGCSE Bulletin* (Vol. 30, pp. 82–86). ACM. Retrieved from http://dl.acm.org/citation.cfm?id=273167

Sengupta, P., Kinnebrew, J. S., Basu, S., Biswas, G., & Clark, D. (2013). Integrating computational thinking with K-12 science education using agent-based computation: A theoretical framework. *Education and Information Technologies*, 1–30.

Townsend, G. C. (2002). People who make a difference: mentors and role models. *ACM SIGCSE Bulletin*, *34*(2), 57–61.

Weintrop, D., Beheshti, E., Horn, M. S., Orton, K., Trouille, L., Jona, K., & Wilensky, U. (In Preparation). Defining Computational Thinking for Science, Technology, Engineering, and Math

Weintrop, D., Beheshti, E., Horn, M. S., Orton, K., Trouille, L., Jona, K., & Wilensky, U. (2014). Interactive Assessment Tools for Computational Thinking in High School STEM Classrooms. In D. Reidsma, I. Choi, & R. Bargar (Eds.), *Proceedings of*

Intelligent Technologies for Interactive Entertainment: 6th International Conference, INTETAIN 2014, Chicago, IL, USA (pp. 22–25). Springer International Publishing.

Wilensky, U., & Reisman, K. (2006). Thinking like a wolf, a sheep, or a firefly: Learning biology through constructing and testing computational theories— an embodied modeling approach. *Cognition and Instruction*, *24*(2), 171–209.

Zweben, S., & Bizot, B. (2012). 2012 Taulbee Survey Strong Increases in Undergraduate CS Enrollment and Degree Production; Record Degree Production at Doctoral Level. *Computing Research News*, *25*(5), 11–60.

Appendix 1 – Male/Female Pretest Differences

Below we report on the exact questions, response numbers, and significance of differences between genders for responses received on the attitudinal survey that was administered at the start of the school year. This survey includes 475 total responses (229 Female/246 Male) and includes students from all four years of high school.

Statement:	Avg. Female Response	Avg. Male Response	T-Statistic
I think being a scientist is a possible career for me.	2.760	3.102	$t(474) = 3.099$, $p < .002$
I think being a mathematician is a possible career for me.	2.502	2.911	$t(474) = 3.717$, $p < .000$
I am interested in a career in computational science	1.725	2.045	$t(474) = 4.025$, $p < .000$
I am interested in a career in engineering	1.747	2.711	$t(474) = 10.96$, $p < .000$
I am interested in a career in mathematics	1.755	2.077	$t(474) = 3.617$, $p < .000$
I am interested in a career in computer science	1.581	2.301	$t(474) = 8.326$, $p < .000$
Generally, I feel comfortable using computational tools.	3.297	3.610	$t(474) = 3.850$, $p < .000$
Generally, I feel comfortable working with computers.	3.799	4.130	$t(474) = 3.976$, $p < .000$
I am relaxed when I work with a computer.	3.537	3.902	$t(474) = 3.947$, $p < .000$
I am used to using computational tools.	3.079	3.463	$t(474) = 4.298$, $p < .000$
Science or math class is more interesting when it involves computational thinking tools or approaches.	2.939	3.252	$t(474) = 4.137$, $p < .000$
Science or math class is more fun when we use CT tools or approaches.	2.969	3.244	$t(474) = 3.622$, $p < .000$
I learn more in class when I get to use CT tools or approaches.	3.031	3.179	$t(474) = 2.096$, $p < .037$
I am more comfortable than my teachers with computational tools.	2.830	3.073	$t(474) = 3.060$, $p < .002$
I am interested in learning more about computers.	3.188	3.715	$t(474) = 5.675$, $p < .000$

Statement:	Avg. Female Response	Avg. Male Response	T-Statistic
I like using computers in my schoolwork.	3.415	3.752	$t(474) = 3.504$, $p < .001$
Computational thinking comes naturally to me.	2.913	3.260	$t(474) = 4.739$, $p < .000$
What is your background using computational tools?	2.009	2.402	$t(474) = 3.881$, $p < .000$
What is your background in computer programming?	1.668	2.187	$t(474) = 5.517$, $p < .000$

Appendix 2 – Female Attitudes towards Computational Thinking in STEM

Below we report on the exact questions, response numbers, and significance of differences on the questions the specifically pertain to confidence in our CT in STEM course. The Male/Female Pretest Difference columns compare responses between male and female students on the pretest. The Pretest/Posttest Female Difference columns show attitudinal changes among female students over the course of the year.

	Male/Female Pretest Difference			Pretest/Posttest Female Difference		
	Avg. Female Response	Avg. Male Response	T-Statistic	Avg. Female Post Response	Pre/Post Difference	T-Statistic
You will be successful in this course?	3.148	3.610	$t(474) = 5.064$, $p < 0.000$	3.281	0.133	$t(349) = 1.170$, $p < 0.243$
You will do well on the computational thinking components of this course?	2.808	3.374	$t(474) = 5.543$, $p < 0.000$	3.033	0.225	$t(349) = 1.885$, $p < 0.060$
You would be successful in future science courses?	3.105	3.553	$t(474) = 4.697$, $p < 0.000$	3.355	0.251	$t(349) = 2.103$, $p < 0.036$
You would be successful in future math courses?	3.402	3.654	$t(474) = 2.777$, $p < 0.006$	3.372	-0.030	$t(349) = 0.268$, $p < 0.789$
You would be successful in future computer science courses?	2.646	3.382	$t(474) = 7.148$, $p < 0.000$	2.868	0.221	$t(349) = 1.798$, $p < 0.073$
You could tutor another student for this course?	2.646	3.037	$t(474) = 3.862$, $p < 0.000$	2.851	0.205	$t(349) = 1.760$, $p < 0.079$
You could analyze a set of data (i.e., look at the relationships between variables)?	3.201	3.528	$t(474) = 3.429$, $p < 0.001$	3.281	0.080	$t(349) = 0.694$, $p < 0.488$
You could ask a meaningful question that could be answered experimentally?	2.983	3.451	$t(474) = 4.791$, $p < 0.000$	3.107	0.125	$t(349) = 1.024$, $p < 0.306$
You could ask a meaningful question that could be answered using computational models?	2.598	3.175	$t(474) = 5.638$, $p < 0.000$	2.777	0.179	$t(349) = 1.443$, $p < 0.150$
You could use a scientific approach to solve a problem outside of class?	3.057	3.455	$t(474) = 4.087$, $p < 0.000$	3.248	0.191	$t(349) = 1.626$, $p < 0.105$
You could solve an unstructured problem (that is, one for which no single 'right' answer exists)?	2.777	3.321	$t(474) = 5.352$, $p < 0.000$	2.893	0.115	$t(349) = 0.953$, $p < 0.341$

	Male/Female Pretest Difference			Pretest/Posttest Female Difference		
	Avg. Female Response	Avg. Male Response	T-Statistic	Avg. Female Post Response	Pre/Post Difference	T-Statistic
You could identify the knowledge, resources, and people needed to solve an unstructured problem?	2.795	3.362	$t(474) = 5.547$, $p <0.000$	3.017	0.222	$t(349) = 1.802$, $p < 0.072$
You could evaluate arguments and evidence so that the strengths and weakness of competing solutions can be judged?	3.057	3.472	$t(474) = 4.005$, $p <0.000$	3.140	0.084	$t(349) = 0.659$, $p < 0.510$
You could weigh the pros and cons of possible solutions to a problem?	3.463	3.756	$t(474) = 3.092$, $p <0.002$	3.529	0.066	$t(349) = 0.580$, $p < 0.562$
You could apply an abstract concept or idea to a real problem or situation in your everyday life?	3.122	3.537	$t(474) = 4.355$, $p <0.000$	3.306	0.184	$t(349) = 1.579$, $p < 0.115$
You could divide complex problems into manageable components?	2.908	3.419	$t(474) = 4.990$, $p <0.000$	3.058	0.150	$t(349) = 1.209$, $p < 0.228$
You could develop several methods that might be used to solve an unstructured problem?	2.786	3.317	$t(474) = 5.232$, $p <0.000$	2.917	0.131	$t(349) = 1.069$, $p < 0.286$
You could extract data from a computational model or tool?	2.703	3.232	$t(474) = 5.139$, $p <0.000$	2.942	0.239	$t(349) = 1.969$, $p < 0.050$
You could convert the data obtained using a computational tool into a result?	2.585	3.134	$t(474) = 5.255$, $p <0.000$	2.769	0.183	$t(349) = 1.487$, $p < 0.138$
You could create a computational model that represents a concept or theory you learned in class?	2.515	2.923	$t(474) = 3.938$, $p <0.000$	2.702	0.187	$t(349) = 1.585$, $p < 0.114$