Integrating Computer Science in Science Classrooms: Learning Computational Thinking and Expanding Perceptions of Computer Science

Introduction

Computing is no longer the exclusive domain of computer scientists. The growth of computers in all aspects of modern life has drawn increasing attention to the role that computing plays in the professional practice of science. From astrophysicists to zoologists, STEM professionals increasingly investigate phenomena salient to their discipline not simply by using computers, but by designing computational tools (Denning, 2017), and by translating those phenomena in ways that are amenable to such tools. Increasingly, scientists must be fluent computational thinkers, and understand the affordances and constraints of different computational tools, in order to successfully use such tools (Foster, 2006; Grover & Pea, 2013). As such, many scholars have recognized the critical importance of integrating computational thinking (CT) into science and mathematics classrooms, in part, to more authentically align science and math education with the work of STEM professionals and better prepare students for academic and career pathways in STEM (Foster, 2006; Weintrop et al., 2016, Voogt et al., 2013). Responsive to this, recent science education standards include CT alongside mathematics as one of eight core science and engineering practices, and emphasize the importance of CT to support multidimensional science learning (NRC, 2012, NGSS Lead States, 2013). Beyond the ever-growing importance of computation in STEM workforce contexts, CT and science learning are reciprocally supportive from a pedagogical standpoint (Weintrop et al., 2016): science concepts are a natural fit for developing learners' facility with CT, and learning CT helps support students in building, deepening, or consolidating their understanding of standards-aligned science concept(s) (Dickes et al., 2016; Guzdial, 1994; NRC 2011a, b; Peel et al. 2019,; Sengupta et al., 2013; Wilensky & Reisman, 2006). CT also affords facile and authentic connections with other science and engineering practices (SEPs) long-recognized as core epistemic practices in science—and accordingly, in science classrooms—such as scientific modeling and data analysis (Lee et al., 2011, Sengupta et al., 2013; Sengupta et al., 2018; Sneider et al., 2014; Weintrop et al., 2016; Wilkerson & Fenwick, 2017).

Context for Study

Project Goals. The [name anonymized] project is an ongoing effort to create and research simulated internship curriculum units that confront barriers to broader participation in computer science (CS) and position coding as a tool for doing science. The 10-day [name anonymized] units integrate CS concepts and practices, such as computational thinking (CT), in core middle school science classrooms and provide teachers with structured, educative curriculum materials to support implementation. The [name of project] model seeks to support broader participation in CS pathways, with a particular emphasis on females; and enable scalable integration of CT learning experiences in core science courses.

Design of [name anonymized] curriculum units. The units have been designed to accomplish project goals by (1) immersing students (Grades 6-8) in simulated internships that mirror the collaborative and computational work of practicing scientists, offering a more inclusive model of CS work (Buffum et al., 2015; Carter, 2006, Charleston, 2012; Hanks et al., 2011; Liebenberg et al., 2012; Werner et al., 2004); and (2) positioning coding in service of addressing real-world problems to counter negative perceptions of CS endeavors as limited in value (Anderson et al., 2008; Carter, 2006; Grover et al., 2014; Margolis et al., 2012); (3) gathering evidence that can advance and deepen the field's understanding of how students' computer science knowledge and practices develop within the context of science learning experiences; and 4) identifying specific factors, designs, and practices that support CT learning, and improve student dispositions toward STEM and CS-related occupations, and that are likely to improve the capacity of teachers and districts to support CS education.

This paper focuses on student CT knowledge and dispositional outcomes in relation to one [name anonymized] 10-lesson learning sequence designed as a digitally simulated virtual internship. For this simulated internship, students inhabit the role of coding science interns working to implement a restoration project to improve the health of coral reef populations. As they develop computational thinking and core programming practices, students deepen and apply their understanding of ecosystem dynamics and interdependence, human impacts on Earth systems, and global climate change to coral reef ecosystems. Students work in teams to program a scientific simulation so that it better represents an evidence-based, explanatory account of a coral reef ecosystem under threat. The purpose of the simulation is to communicate to stakeholders how various threats affect the health of a coral reef and how those threats may be mitigated. Through this effort, students debug existing code as well as program additional simulation features. Students also write code to program underwater robots to remove threats in variable conditions. Students gain first-hand experience with sequences, loops, and conditionals. Student teams work together to evaluate their code by testing executed code against expected outcomes and real-world data.

Our pedagogical framework grounds Mitch Resnick's coding to learn approach to CS education (Resnick, 2013) in situative learning theories and, in particular, the construct of legitimate peripheral participation (LPP, Lave & Wenger, 1991). Coding to learn, in which learning is contextually embedded in authentic tasks, is a natural fit for situative theories, like LPP, which posit that knowledge is constructed through activity and in relation to others—thus, learning is "situated" in the activity, context, and community in which it occurs (see also Greeno, 1998). LPP's foundational construct of a community of practice (Lave & Wenger, 1991; Wenger, 1998), where individuals share a repertoire of knowledge and practices to address a shared set of problems, informs the intervention's commitment to authentic and collaborative problem solving. Further, with the simulated internship model, in conjunction with task-embedded supports and a gradual release of responsibility, students progress along a trajectory from peripheral to more central participation in the practice of coding scientific simulations. The LPP framework is tightly aligned with the intervention's more distal goal of broadening participation in STEM, offering a clear vehicle for students to explore and come to identify with CS as a "possible self" (Markus & Nurius, 1986). Markus' theory also dovetails with the widely reported distal impacts of students' early perceptions of STEM subjects as a determinant of whether students enter STEM professions (Tai, Liu, Maltese & Fan, 2006). With this body of research in mind, learning experiences were expressly designed to encourage a broad range of students to see themselves, at least potentially, as both scientists and programmers, and to envision CS as a more appealing world to inhabit.

Methods

The mixed-methods research agenda for the broader project this study represents is guided by three questions: 1) To what extent is there evidence that a CT-S integrated learning sequence is likely to improve student learning outcomes and catalyze dispositional shifts toward STEM academic and career pathways?; 2) What specific design features and instructional strategies of the learning sequence and instructional materials are most important for broadening student participation in CS?; 3) What aspects of the learning sequence are most important to support sustainability of CS and science integration? Findings presented for this paper are focused on the first of these research questions, with analyses of data for the other two questions ongoing.

Design-based research. The instructional sequence examined in this paper has been extensively piloted (instruction led by the research team) and field tested (instruction led by classroom teachers) over the two years prior to the research trial reported herein. Based on iterative classroom pilot testing (215 students, 10 class sections in two western states) and insights from project advisors, we revised the unit, then implemented the revised version in field trials in 8 middle school science classrooms in two states during the 2018-2019 school year. Findings from this field trial suggested promise for student learning, with significant learning gains pre- to post-instruction for n=391 students (Authors, 2020a). The field trials enabled researchers to gather validity evidence for administered modifications to the dispositional measures (see discussion of measures below), and to focus and refine our professional learning supports for teachers. With research ongoing, emerging insights from our DBR process, including implications for pedagogical design and scalability are reported elsewhere (Authors, 2020b). Because reported findings reflect an instructional sequence developed over 3 years of iterative design based research, data sources for the broader study are included below.

Data source	Sample
Teacher and district administrator interviews	n=8 (5 teachers & 3 administrators; multiple interviews with each)
Observations of developer-led classroom pilots for each of two 10-lesson instructional sequences	114 total observations across all piloted lessons for each unit Unit 1: 215 students in 10 class sections Unit 2: 260 students in 9 class sections
Observations of teacher-led implementation for one unit	15 total observations across 7 lessons in multiple sections for 2 teachers and approximately 325 students.
Teacher surveys and daily reflection logs	n=10 middle school science teachers (2018-2019) n= 9 middle school science teachers (2019-2020)
Field Trial 1 (2018-2019) Findings reported in (Authors, 2020a & 2020b)	n=391 middle school science students
Research Trial (2019-2020) SUBJECT OF CURRENT ANALYSES	n=381 middle school science students
Development team and stakeholder meeting notes	Meeting notes cover two years of iterative unit development work with curriculum developers, learning science researchers, educators, and an external expert panel.

Figure 1: Data sources for full study

Analysis of outcomes. Reported outcomes relate to a research trial conducted during the 2019-2020 school year with n=381 middle school science students in three public school districts from two western states. Schools representing just over 1000 students expressed interest in participating, and all who were interested were invited to participate. Of that group, approximately 600 students completed the unit, and we were able to match pre/post responses for 381 students. Unmatched responses (e.g., from incorrect user names or absences at pre or post) are not included in the current analyses.

Measures. Reported findings for changes in computational thinking focus on pre/post changes in student performance on the Assessment of Computational Thinking (see Snow et al., 2017 for background on items; see Witherspoon et al., 2017 for technical report on evidence for measurement validity). Reported findings for dispositional changes are drawn from the suite of Activation Lab measures (Dorph et al., 2016), specifically, the scales for Competency Beliefs (Chung et al., 2016) and Valuing Science (Chung et al., 2016). Some items on each of the Activation Lab scales were modified to focus on computing and computer science instead of science for this administration (e.g., items about the perceived importance of science were revised to focus on programming).

Findings

Changes in student knowledge and dispositions over the course of instruction

Computational Thinking. Analysis using a paired samples t-test of a pre/post survey for n=381 students revealed that students who participated in the 2 week/10 lesson integrated computational thinking in science learning sequence demonstrated significant learning gains on an external measure of CT (0.522***; effect size=0.32).

Competency-beliefs for STEM. Analysis of a retrospective post instruction survey for n=368 students revealed that students who participated in the 2 week/10 lesson learning sequence reported significantly greater confidence with STEM after instruction compared to prior to instruction (1.129***; effect size=1.01).

Valuing STEM. Analysis of pre/post mean differences for n=380 students revealed that students who participated in the 2 week/10 lesson learning sequence assigned significantly greater value to STEM academic and career pursuits after instruction (0.778***; effect size=0.95).

	Paired Differences							
				95% Confidence Interval of the Difference				
Construct	Mean Differnece	Std. Deviation	St. Error Mean	Lower	Upper	t	df	Sig. (2-tailed)
Computational Thinking	0.522	1.642	0.084	0.357	0.688	6.207	380	0.0000
Competency Beliefs	1.12899	1.11735	0.05825	1.01446	1.24353	19.383	367	0.0000
Valuing Computer Science	0.77763	0.8218	0.04216	0.69474	0.86052	18.446	379	0.0000

Figure 4: Paired samples t-test of mean differences in valuing STEM from pre- to post-instruction

Analysis of variance between subgroups

The ultimate objective of the instructional sequence is to support broader participation in STEM academic and career pathways by confronting long standing gender inequities in the computer science domain. As such, we examined each of the constructs reported above for variation between gender groups that would suggest inequitable learning experiences.

Computational Thinking. At the end of the pre and post surveys, students were asked to self-identify gender, with options for male, female, non-binary, and “prefer not to say.” To examine gender differences, we ran a one-way ANOVA and found no significant differences between groups. A Tukey post hoc test confirmed no significant difference between pre-instruction mean scores for students who identified as female (3.48 ± 1.65; n=178) compared to those who identified as male (3.80 ± 1.81; n=173), nor for post-instruction mean scores (4.04 ± 1.58 for females, compared to 4.37 ± 1.64 for males). Accordingly, learning gains for the two groups were nearly identical, with a mean increase of 0.56 (effect size=0.35) for students who identified as girls and a mean increase of 0.57 (effect size=0.35) for students who identified as boys.

Competency-beliefs for STEM. Both males and females reported significant increases in competency-beliefs for STEM after instruction. To further examine gender differences, we ran a one-way ANOVA test and found a statistically significant difference between groups for both the pre- ($F(3,366) = 5.333$, $p = .001$) and post- ($F(3,354) = 5.852$, $p = .001$) components of the retrospective post-instruction survey. A Tukey post hoc test revealed female students reporting significantly less confidence with STEM both prior to and after instruction (Pre: 3.08 ± 1.25; Post: 4.26 ± 0.925) than their male counterparts (Pre: 3.61 ± 1.38; Post: 4.68 ± 1.03). There was no statistically significant difference between females and either non-binary students or students who preferred not to state gender. This suggests that the learning sequence, on its own, is insufficient to achieve parity in student beliefs about their own competency for STEM. However, there is strong evidence that the learning sequence is making considerable progress toward this goal. To wit, females reported a larger increase in competency-beliefs than males, with a mean difference of 1.20 (effect size: 1.14; n=175) compared to a mean difference of 1.10 (effect size: 0.950; n=165) for males.

Valuing STEM. Both males and females reported significant increases in value assigned to STEM after instruction. To further examine gender differences, we ran a one-way ANOVA test and found no significant differences between groups.

Discussion

Taken together findings offer strong evidence of promise for integrating computational thinking into core middle school science classrooms, with significant learning gains and positive dispositional changes observed. We also found evidence that instructional design focused on addressing gender inequities in STEM may have contributed to equitable outcomes across both knowledge and dispositional metrics. While outside the scope of the present study, student and teacher interviews suggest learning experiences grounded in solving real-world problems was a key motivator for equitable engagement.

Given the increasingly computational world in which we live, and in which much of modern science operates, developing evidence-based models for the integration of computational thinking in core science classrooms is critical for the teaching and learning of science. Accordingly, the evidence presented in this paper demonstrating significant student outcomes for this simulated internship model for CT-S integration will be of high interest to the NARST community. In particular, given NARST's goal of helping all learners achieve science literacy, we are eager to share with fellow NARST members evidence that an instructional sequence emphasizing collaborative learning, with students designing solutions to real-world problems, can result in large positive effects for girls' disposition toward STEM. A conference presentation at NARST would also provide an opportunity to explore the instructional design in more detail and engage NARST members in discussion of outcomes in relation to that design.

Acknowledgments

Material is based upon work supported by the National Science Foundation under Grant No. XXXXX. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Anderson, N., Lankshear, C., Timms, C., & Courtney, L. (2008). 'Because it's boring, irrelevant and I don't like computers': Why high school girls avoid professionally-oriented ICT subjects. *Computers & Education*, *50*(4), 1304-1318.

Authors (2020a)

Authors (2020b)

Buffum, P. S., Frankosky, M., Boyer, K. E., Wiebe, E., Mott, B., & Lester, J. (2015, August). Leveraging collaboration to improve gender equity in a game-based learning environment for middle school computer science. In *2015 Research in Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)* (pp. 1-8). IEEE.

Carter, L. (2006). Why students with an apparent aptitude for computer science don't choose to major in computer science. *ACM SIGCSE Bulletin*, *38*(1), 27-31.

Charleston, L. J. (2012). A qualitative investigation of African Americans' decision to pursue computing science degrees: Implications for cultivating career choice and aspiration. *Journal of Diversity in Higher Education*, *5*(4), 222.

Chung, J., Cannady, M. A., Schunn, C., Dorph, R., & Bathgate, M. (2016). Measures technical brief: Competency Beliefs in Science. Retrieved from http://activationlab.org/wp-content/uploads/2018/03/Competency-Beliefs-Report-3.2-20160331.pdf

Chung, J., Cannady, M. A., Schunn, C., Dorph, R., & Bathgate, M. (2016). Measures technical brief: Valuing Science. Retrieved from http://activationlab.org/wp-content/uploads/2018/04/Values-Report-3.3-20170306.pdf

Denning, P. J. (2017). Computational Thinking in Science. *American Scientist*, 105(1), 13.

Dickes, A. C., Sengupta, P., Farris, A. V., & Basu, S. (2016). Development of mechanistic reasoning and multilevel explanations of ecology in third grade using agent-based models. *Science Education*, *100*(4), 734-776..

Dorph, R., Cannady, M. A., & Schunn, C. D. (2016). How science learning activation enables success for youth in science learning experiences. *The Electronic Journal for Research in Science & Mathematics Education*, *20*(8).

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

Greeno, J. G. (1998). The situativity of knowing, learning, and research. *American psychologist*, *53*(1), 5.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational researcher*, *42*(1), 38-43.

Grover, S., Pea, R., & Cooper, S. (2014, March). Remedying misperceptions of computer science among middle school students. In *Proceedings of the 45th ACM technical symposium on Computer science education* (pp. 343-348).

Guzdial, M. (1994). Software-realized scaffolding to facilitate programming for science learning. *Interactive learning environments*, *4*(1), 001-044.

Hanks, B., Fitzgerald, S., McCauley, R., Murphy, L., & Zander, C. (2011). Pair programming in education: a literature review. *Computer Science Education*, *21*(2), 135-173.

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge university press.

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., ... & Werner, L. (2011). Computational thinking for youth in practice. *Acm Inroads*, *2*(1), 32-37.

Liebenberg, J., Mentz, E., & Breed, B. (2012). Pair programming and secondary school girls' enjoyment of programming and the subject Information Technology (IT). *Computer Science Education*, *22*(3), 219-236.

Margolis, J., Ryoo, J. J., Sandoval, C. D., Lee, C., Goode, J., & Chapman, G. (2012). Beyond access: Broadening participation in high school computer science. *ACM Inroads*, *3*(4), 72-78.

Markus, H., & Nurius, P. (1986). Possible selves. *American psychologist*, *41*(9), 954.

National Research Council. (2011a). *Learning science through computer games and simulations*. National Academies Press.

National Research Council. (2011b). *Report of a workshop on the pedagogical aspects of computational thinking*. National Academies Press.

National Research Council. (2012). *A framework for K-12 science education: Practices, crosscutting concepts, and core ideas*. National Academies Press.

NGSS Lead States. (2013). *Next generation science standards: For states, by states*. National Academies Press. NGSS Lead States. 2013.

Peel, A., Sadler, T. D., & Friedrichsen, P. (2019). Learning natural selection through computational thinking: Unplugged design of algorithmic explanations. *Journal of Research in Science Teaching*, *56*(7), 983-1007.

Resnick, M. (2013). Learn to code, code to learn. *EdSurge, May*, *54*.

Sengupta, P., Dickes, A., & Farris, A. (2018). Toward a phenomenology of computational thinking in STEM education. In *Computational thinking in the STEM disciplines* (pp. 49-72). Springer, Cham.

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*, *18*(2), 351-380.

Sneider, C., Stephenson, C., Schafer, B., & Flick, L. (2014). Exploring the science framework and NGSS: Computational thinking in the science classroom. *Science Scope*, *38*(3), 10.

Snow, E., Rutstein, D., Bienkowski, M., & Xu, Y. (2017, August). Principled assessment of student learning in high school computer science. In *Proceedings of the 2017 ACM Conference on International Computing Education Research* (pp. 209-216).

Tai, R. H., Liu, C. Q., Maltese, A. V., & Fan, X. (2006). Planning early for careers in science. *Life sci*, *1*(0.2).

Voogt, J., Erstad, O., Dede, C., & Mishra, P. (2013). Challenges to learning and schooling in the digital networked world of the 21st century. *Journal of computer assisted learning*, *29*(5), 403-413.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127-147.

Wenger, E. (1998). Communities of practice: Learning as a social system. *Systems thinker*, *9*(5), 2-3.

Werner, L. L., Hanks, B., & McDowell, C. (2004). Pair-programming helps female computer science students. *Journal on Educational Resources in Computing (JERIC)*, *4*(1), 4-es.

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.

Wilkerson, M. H., & Fenwick, M. (2017). Using mathematics and computational thinking. *Helping students make sense of the world using next generation science and engineering practices*, 181-204.

Witherspoon, E. B., Higashi, R. M., Schunn, C. D., Baehr, E. C., & Shoop, R. (2017). Developing computational thinking through a virtual robotics programming curriculum. *ACM Transactions on Computing Education (TOCE)*, *18*(1), 1-20.

Figure 1. [Include all figures in their own section, following references (and footnotes and tables, if applicable). Include a numbered caption for each figure. Use the Table/Figure style for easy spacing between figure and caption.]

For more information about all elements of APA formatting, please consult the *APA Style Manual, 6th Edition*.