Teacher Perceptions of the Adaptation of the New Computer Science (CS) Curriculum: An Evaluation of CS Curriculum Implementation.

Suhkyung Shin
Texas Tech University

Jongpil Cheon
Texas Tech University

Sungwon Shin
Texas Tech University

Abstract

The purpose of this study is to assess the adaptation of the newly developed Computer Science (CS) curriculum in middle schools and address teachers' challenges during its implementation. Since CS has been introduced as a new subject in K-12 schools in the U.S., it is essential to determine the current implementation of CS curriculum from teachers' perspectives in order to enhance the CS education movement. Considering the fact that curriculum implementation interacts with multiple factors in K-12 school systems, this study assesses the current status of CS curriculum implementation in middle schools based on four categories: (1) teacher, (2) curriculum, (3) context, (4) students. The participants were middle school teachers who taught Introduction to Programming course in the Southwestern United States. The findings showed that teachers had positive attitudes toward CS education and received enough opportunities for professional development in CS. However, teacher sought effective instructional strategies to enhance students' understanding of CS concepts and better engage their students in CS learning. As for the curriculum, teachers perceived the scope, sequence and learning objective of the CS curriculum were appropriate for the purpose, whereas they needed more time to enhance students' understanding of CS concepts. In addition, the findings suggested that some schools do not have appropriate equipment or facilities. Based on our findings, recommendations for implementing CS curriculum to support CS education and implications for future research are presented and discussed.

Introduction

Computer Science (CS) is becoming an essential component of students' educational development for computational thinking skills including problem-solving skills, analytic thinking skills, and creativity (Atmatzidou & Demetriadis, 2016; Papasterigiou, 2009). In an effort to promote these skills, there are numerous CS education initiatives. Nationally, Computer Science for All called for a $4.1 billion budget to support K-12 teacher professional development and provide high-quality instructional resources (Office of the Press Secretary, 2016). Further initiatives including Code.org, CS first, and Code Academy have expanded access to instructional materials to support K-12 CS education.

In recent years, new CS curricula have been developed and implemented in K-12 schools in the U.S., and school policy has been changed to categorize CS as a part of the math and science core curriculum across states (Alba & Huett, 2017; Nager & Atkinson, 2016). For example, at the high school level, CS counts as a core graduation requirement in 23 states and the District of Columbia (Stanton et al., 2017). Specifically, Arkansas, Texas, Virginia, and West Virginia now require public high schools to offer at least one CS course. At the middle school level, CS education has also grown. For example, several states including Nevada and Indiana, aim to promote student access to CS education through computing courses in elementary and middle schools (Stanton et al., 2017). Despite some efforts and initiatives to enhance middle school CS education, less attention has been given to middle schools compared to the high school level. Thus, to promote CS education in middle schools, CS curriculum for middle schools' students needs to be developed and further supported.

In addition to school policy, the current adaptation and implementation of new CS curriculum should be evaluated to better support the new movement. Several researchers have noted that teachers may face challenges throughout the new curriculum implementation process (Brown et al., 2013; Finger & Houguet, 2009; Sentance &

Csizmadia, 2017; von Wangenheim, Alves, Rodrigues, & Hauck, 2017). For example, it requires them to changes in both subject matter and pedagogical knowledge (Alba & Huett, 2017; Thompson et al., 2013). Since this may be an unfamiliar area for some teachers, they may not have enough experience to teach CS contents, or they may feel ill-equipped to teach new contents knowledge. As a result, they may need to seek appropriate pedagogies for teaching CS, such as algorithms, or programming, to facilitate students' computational thinking skills. However, we have yet to understand how teachers have adopted the newly-developed CS curriculum and how they perceive their adaptation. In order to understand the current status of the CS curriculum and identify areas of improvement, the challenges that teachers encounter must be examined.

Furthermore, since school systems interacts with multiple variables when implementing a new curriculum in classrooms, we need to highlight other relevant factors, such as students and context, which may affect the new CS curriculum in K-12 school settings. For example, possible issues would be lack of students' readiness for learning CS, lack of students' engagement, appropriateness of scope and sequence of developed curriculum, and lack of infrastructure across schools (Sentance & Csizmadia, 2017; Wangenheim et al, 2017). Considering that teachers, students, and school contexts vary and interact together in dynamic ways when implementing a new curriculum across schools (Remillard, 2005), it is critically important to explore multiple factors influencing the CS curriculum implementation so that we can provide a holistic diagnosis of the current status of CS education initiatives.

Purpose of the Study

The purpose of this study is to assess the current adaptation of newly developed CS curriculum and identify teachers' challenges and the research questions of this study are:

1. How have teachers adopted the new CS curriculum?
2. How do teachers perceive their adaptation to the new CS curriculum?
3. What are the challenges to implementing the new CS curriculum?

Method

Context and Participants

The "Introduction into Programming" was developed as a CS education initiative by the College of Education with the support of an urban school district funding in 2016. It was implemented in the 30 public middle schools within the school district. Participants in this study were 12 middle school teachers (25% males, 75% females) who taught the course using the newly developed curriculum in 2016.

The CS Curriculum Design

The "Introduction into Programming" curriculum includes various activities, such as storytelling, animation, and games using Scratch, Google CS First, and Code.org. This eighteen-week long curriculum consists of 75 lessons, and the first and second six weeks focus on different CS concepts with unplugged activities and programming projects, and the third six weeks focus on HTML and CSS (See Figure 1).

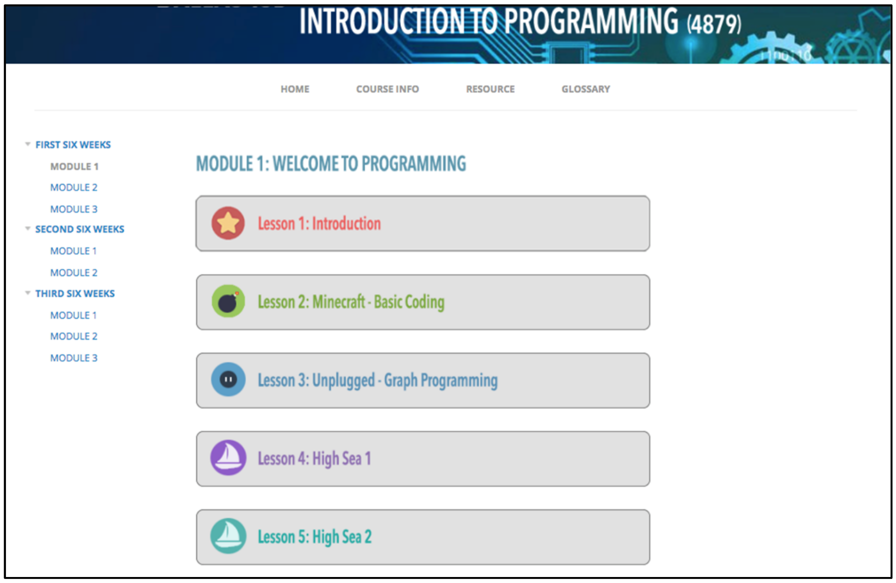

Figure 1. Students CS Website

Data Collection and Analysis

Online-Survey. By referencing the components of the teacher-curriculum relationship proposed by Remillard (2005), we developed four constructs to assess the curriculum implementation. Our proposed framework includes these four categories: (1) teacher, (2) students, (3) curriculum, (4) context. The survey items adapted from previous studies (Friday Institute for Educational Innovation, 2012; Sharp, Hopkin & Lewthwaite, 2011; Ko & Lee, 2003; Ross, McDougall, Hogaboam-Gray & Lesage, 2003) which were modified to fit our research context in order to measure the CS curriculum implementation. The online survey consisted of 83 Likert-scale items with 15 open-ended questions. The survey included questions about how teachers used the curriculum in their teaching and how they perceived the CS curriculum implementation using the four categories. The open-ended questions explored teachers' challenges in implementing the CS curriculum.

Data Analysis. Descriptive statistical analyses were performed on the survey data to investigate the implementation of the new CS curriculum and teachers' perceptions to the new curriculum. Content analysis was used to examine data from open-ended responses in the survey to identify teachers' challenges in teaching CS.

Procedure

The online survey was distributed to thirty middle schools by means of the school district office in spring 2017. Among 30 middle school teachers, 12 teachers completed the survey and their responses were included in the analysis.

Results

Use of the New CS Curriculum in Teaching

To answer the first research question, we investigated how the CS teachers used the new curriculum in their instruction. As shown Table 1, out of the 8 modules, teachers used the first five modules, but only two teachers used the module to introduce HTML, and no one used the last two modules.

Table 1. Used modules in CS course (*N*=12, Multiple responses)

Modules	Used Module (N)
First Six Weeks	
M1- Welcome to programming	11
M2 – Tell your story	10
M3 – Be an artist	10
Second Six Weeks	
M1 – Game Design	12
M2 - Animation	9
Third Six Weeks	
M1 – Getting Started with HTML	2
M2 – Getting Organized with HTML	0
M3 – Getting Stylish with CSS	0

In addition, teachers were asked which module was the most interesting and challenging for their students. An analysis of survey data showed that 75% teachers responded that the "Game design" module was the most interesting, and 25% of teachers replied that "Game design" and "Animation," respectively, were the most challenging modules for their students (See Table 2).

Table 2. The most interesting and challenging modules for students (*N*=12)

Modules	The most interesting module	The most challenging module
First Six Weeks		
M1- Welcome to programming	2 (16.67%)	1 (8.33%)
M2 – Tell your story	1 (8.33%)	0 (0.0%)
M3 – Be an artist	0 (0.0%)	1 (8.33%)
Second Six Weeks		
M1 – Game Design	9 (75.0%)	3 (25%)
M2 - Animation	0 (0.0%)	3 (25%)
Third Six Weeks		
M1 – Getting Started with HTML	0 (0.0%)	2 (16.7%)
M2 – Getting Organized with HTML	0 (0.0%)	1 (8.33%)
M3 – Getting Stylish with CSS	0 (0.0%)	1 (8.33%)

Teachers Perceptions of Their Adaptation to the New CS Curriculum

The second question was about how teachers perceived their adoption of the new CS curriculum. The four categories (i.e., teacher, curriculum, student, context) were used to evaluate the CS curriculum implementation. In the teacher category, our analysis of survey data showed that teachers had positive attitudes toward CS education (M = 4.83, SD = .39). For example, 83% responses strongly agree with the statement: "I support the inclusion of Computer Science education in middle school curriculum" (M = 4.83, SD = .39). Moreover, all teacher responses either agree or strongly agree with the statement, "I am confident that I can explain key Computer Science concepts to students with examples" (M = 4.5, SD = .52), suggesting that teachers showed high confidence in teaching CS (See Table 3).

Table 3. Results of teachers' perceptions of their adaptation: Teacher (*N*=12)

Construct	Number of Items	Cronbach's α	M	SD
Attitude toward CS education	3	1.00	4.83	.39
Subject Knowledge	5	.53	4.37	.40
Teaching Efficacy	3	.96	4.56	.50
Intention	3	.76	4.03	.87
Total	14	.711	4.43	.35

Regarding the curriculum, the results showed that teachers overall perceived the scope, sequence and learning objective were appropriate for the purpose (See Table 4). Among the constructs, the mean scores of materials was high (*M* = 4.06, *SD* = .66), indicating that teachers perceived the student curriculum website was useful for students and their teaching. On the other hand, instruction time was the lowest score (*M* = 2.83, *SD* = 1.10). For example, in response to the survey item, "Each lesson's learning activities could be completed within the given instructional time," over 42% of the responses either strongly disagreed or disagreed.

Table 4. Results of teachers' perceptions of their adaptation: Curriculum (*N*=12)

Construct	Number of Items	Cronbach's α	M	SD
Scope	3	.97	3.86	.86
Sequence	3	.93	3.69	.82
Learning Objectives	3	.98	3.67	.99
Standards	3	1.00	3.83	.83
Complexity	3	.96	3.58	.95
Time	3	.98	2.83	1.10
Materials	3	.899	4.06	.66
Promoting Higher-Order Thinking	3	.82	3.75	.74
Assessment	3	.95	3.19	.99
Total	27	.96	3.61	.64

In contrast to the teacher side, the mean of teacher perceptions of students was the lowest in the four categories (*M* = 3.33, *SD* = .29). An analysis of survey data indicated that the students lacked prior programming skills (*M* = 2.33, *SD* = .72). For example, over 66% of teachers either strongly disagree or disagree with the statement: "My students have prior programming experiences" (*M* = 2.08, *SD* = .79). To gain in-depth understanding of students' challenges, we asked teachers what challenges their students faced in the CS course. An analysis of open-ended questions suggested that students' lack of prior programming skills may impact individual students understanding of CS concepts and instructional time. In open-ended responses, many teachers mentioned that their students have no prior programing experience and have a hard time understanding the idea of essential concepts. For example, one teachers commented, "They are unable to use the concepts introduced by the CS First activities to create original programs on their own" [Teacher 3].

Table 5. Results of teachers' perceptions of their adaptation: Student (*N*=12)

Construct	Number of Items	Cronbach's α	M	SD
Prior Computer Skills	3	.93	4.11	.78
Prior Programming Skills	3	.64	2.33	.72
Interests	3	.96	3.67	.71
Difficulty*	3	.80	3.22	.87
Total	12	.71	3.33	.29

* Reversed items

Results also revealed that teachers needed more time to enhance students' understanding of CS concepts. As some teachers stated, teachers spent instructional time in explaining basic information rather than engaging them in programming activities. As a result, teachers felt they did not have enough time to complete the designed activities. As one teacher mentioned, "Majority of students minus a handful of 130 students have no experience in any capacity with regards to programming, coding, coding concepts, or all of the above. I had to baby step them in more slowly than the curriculum called for but in the end it worked out well….." [Teacher 4].

As for the context, overall, the teachers perceived that they received enough opportunities for professional development in CS (M = 4.11, SD = .73), and have received appropriate administrators support (M = 3.89, SD =.76) and technological support (M = 3.47, SD = .74).

Table 6. Results of teachers' perceptions of their adaptation: Context (N=12)

Construct	Number of Items	Cronbach's α	M	SD
Admin Support	3	.90	3.89	.76
Technical Support	3	.86	3.47	.74
Professional Development	3	.76	4.11	.73
Resources [Facility/Materials]	3	.90	3.86	.90
Total	12	.62	3.83	.39

Teachers' Challenges Implementing the New CS Curriculum

Teachers challenges of implementing the CS curriculum were identified from open-ended questions in the four categories (i.e., teacher, student, curriculum, context), and these are discussed below.

Teacher. The analysis of teachers' open-ended responses indicated that they feel challenges in (1) lack of teaching methods for differentiate instruction, and (2) lack of CS content knowledge. Over 50% of teachers reported the level of programming skills of their students vary, and they have different grade level students in the same class. Some teachers struggled with addressing diverse students' needs and they sought effective pedagogical approaches, specifically differentiated instruction to better engage students in CS learning. Some teachers examined:

> *"Managing the speedy students who complete everything quickly and the slower students who struggle to complete their programs on-time" [Teacher 3].*
> *"Having 6th, 7th, and 8th grade together in the same class...." [Teacher 6.]*

Although many teachers perceived they have received appropriate professional development, a few teachers felt a lack of CS content knowledge and sought further training, specifically on the topic of HTML and CSS. For instance, one teacher stated: "For me it will be the HTML and CSS until I can fully understand it" [Teacher 10].

Curriculum. Although it was not frequently mentioned, a few teachers expressed concern regarding assessing students' learning outcomes. For example, when asked about teachers' challenges implementing the CS curriculum, one teacher mentioned: "Assessing students' computational thinking skills" [Teacher 3].

Student. In the student category, our analysis revealed that teachers were struggling in teaching CS due to their students' lack of (1) basic computer skills, and (2) motivation and engagement. Most teachers mentioned that many students have no basic computer skills. For example, one teacher stated that: "Some students have NO experience working with a computer at all…." [Teacher 6]. Some teachers also reported that their students are not motivated and not engaged in their learning.

Context. In context, approximately 40% of teachers reported they do not have enough headphones and have network issues in their classroom. For instance, one teachers explained, "We need new computers, a class set of headphones, fewer restrictions on student accounts, and better server setup for more consistent access to online software" [Teacher 1].

Discussion

This study investigated the current status of the adoption of the newly CS curriculum in middle schools, as well as the challenges that teachers faced in teaching CS. Based on our findings, we proposed some suggestions for implementing CS curriculum.

First, teacher competency development needs to be enhanced. The findings showed that although many teachers perceived that they received sufficient professional development opportunities, some teachers commented that they need additional training on certain topics such as HTML and CSS. Because some teachers may not have been adequately prepared to acquire the contents knowledge and teach the topic. Thus, regular professional development sessions should be offered to CS teachers based on the teachers' specific needs. In addition, further support needs to be provided to foster effective teaching practices. Result indicated that teachers face difficulties to better engage their students in learning activities due to students' mixed abilities in one classroom. Many teachers also reported that students lack motivation and engagement. To enhance students' learning experience in CS classrooms, we could encourage teachers to develop CS-specific instructional strategies, especially differentiated instruction, via professional networks or local-based communities, which may facilitate conversations among teachers, researchers, practitioners and administrators. There are number of studies on case-based learning indicating that reviewing other teachers' teaching practices using videos may enhance teaching competency (Gamoran & van Es, 2002; Seidel, Blomberg & Renkl, 2013). Thus, CS teachers could receive the same benefits by reviewing exemplary teaching practices for their areas. The examples could include a developed activity, or video cases capturing classroom practices. Authentic examples will provide teachers with more valuable contextual knowledge and support for adopting the curriculum in their context. For this, multiple exemplary materials and resources related to teacher training should be developed in CS education.

Second, regarding curriculum, we should revisit the scope and sequence of curriculum, and revised it based on gaps identified in our analysis. The results of this study indicated that many students lack basic computer skills and programming experience. To promote students' basic understanding of the content and practices, teachers used their instruction time, and as a result, they could not complete the CS activities as they were originally designed. Building students' basic computer skills, which are required to learn CS, could be addressed in advance in other courses, such as Technology Application, or elementary school level. Our findings showed that the "game design" module was both students' favorite and most challenging module in the curriculum. This implies that its complexity and entertaining content may facilitate students' engagement and motivation, and, at the same time, challenge them to complete the activity. Accordingly, to better engage students and facilitate their problem-solving processes, scaffolds can be designed in advance in anticipation of students' difficulties (Saye & Brush, 2002). Thus, to facilitate students' motivation and engagement, additional scaffolds can be designed and embedded in the activity. With respect to differentiated instruction, the current curriculum needs to be redesigned to support these teachers' needs and enable them to customize the curriculum in each school.

Lastly, regarding context, school and district administration needs to provide on-going support, especially infrastructure. The results of this study suggested that some schools may not have appropriate equipment or facilities, such as headphones, software, or networks, which hinder efforts to offer a high-quality of CS education.

The findings of this study provide useful insights and understanding about the implementation of CS curriculum with respects to four specific areas. The results can be used for building common shared goals among teachers, students and their parents, and school and district administrators to strengthen the new CS education initiatives.

Based on our findings, further research needs to be conducted on several research topics. First, more in-depth study to evaluate the newly developed CS curriculum implementation will help to identify areas of improvement. Conducting a case study based on teachers' and students' interviews, as well as observations may bring deeper understanding of individual schools' contexts. Second, rigorous measurements which can assess students' content mastery and computational thinking skills should be developed. Our findings imply that teachers experience difficulty in developing assessments. Furthermore, to improve the current CS curriculum, students' learning outcomes and performance should be identified and reflected in the revision of CS curriculum in the future.

References

Alba, A., & Huett, K. (2017). Learning computational skills in uCode@ UWG: Challenges and recommendations. In P. Rich & C. Hodges (Eds.), *Emerging Research, Practice and Policy on Computational Thinking* (pp. 3-20). Springer.

Atmatzidou, S., & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems, 75*, 661-670.

Brown, N., Kölling, M., Crick, T., Jones, S. P., Humphreys, S., & Sentance, S. (2013). Bringing computer science back into schools: Lessons from the UK. *Proceedings of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE 2013)*, 269–274.

Finger, G., & Houguet, B. (2009). Insights into the intrinsic and extrinsic challenges for implementing technology education: Case studies of Queensland teachers. *Information Journal of Technology and Design Education, 19*(2), 309-334.

Friday Institute for Educational Innovation (2012). *Middle and High School STEM-Student Survey*. Raleigh, NC: Author.

Gamoran Sherin, M., & van Es, E. A. (2002). Using video to support teachers' ability to interpret classroom interactions. In S. Talley (Ed.), *Video cases. Society for information technology & teacher education, SITE 2002 Section* (pp. 2532e2536). Norfolk, VA: Association for the Advancement of Computing in Education, Retrieved from:

http://www.eric.ed.gov/ERICWebPortal/search/detailmini.jsp?_nfpb.true&_&ERICExtSearch_SearchValue_0.ED472254&ERIC ExtSearch_SearchType_0.no&accno.ED472254.

Ko, A. C., & Lee, J. C. (2003). Teachers' perceptions of teaching environmental issues within the science curriculum: A Hong Kong perspective. *Journal of Science Education and Technology, 12* (3), 187-204.

Nager, A., & Atkinson, R. D. (2016). The case for improving U.S. computer science education. *National Consortium of Secondary STEM Schools, 21* (1), 18-19.

Office of the Press Secretary (2016). FACT SHEET: President Obama announces computer science for all initiative. Retrieved from: https://www.whitehouse.gov/sites/whitehouse.gov/files/images/FACT%20SHEET%2BPresident%20Obama%20Announces%20Computer%20Science%20For%20All%20Initiative_0.pdf

Papastergiou, M. (2009). Digital game-based learning in high school computer science education: Impact on educational effectiveness and student motivation. *Computers and Education, 52* (1), 1-12.

Remilard, J. T. (2005). Examining key concepts in research on teachers' use of mathematics curricula. *Review of Educational Research, 75* (2), 211-246.

Ross, J. A., McDougall, D., Hogaboam-Gray, a., & Lesage, A. (2003). A survey measuring elementary teachers' implementation of standards-based mathematics teaching. *Journal for Research in Mathematics Education, 34*(4), 344-363.

Saye, J. W., & Brush, T. (2002). Scaffolding critical reasoning about history and social issues in multimedia-supported learning environments. *Educational Technology Research and Development, 50*(3), 77-96.

Seidel, T., Blomberg, G., & Renkl, A. (2013). Instructional strategies for using video in teacher education. Teaching and Teacher Education, 34, 56-65.

Sentance, S., & Csizmadia, A. (2017). Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Education and Information Technologies, 22*(2), 469-495.

Sharp, J. G., Hopkin, R., & Lewthwaite, B. (2011). Teacher perceptions of science in the national curriculum: Findings from an application of the science curriculum implementation questionnaire in English primary schools. *International Journal of Science Education, 17*(15), 2407-2436.

Stanton, J., Goldsmith, L., Adrion, W., Dunton, S., Hendrickson, K., Peterfreund, A., Yongpradit, P., Zarch, R., & Zinth, J. (2017, March 1). *State of states landscape report: State-level policies supporting equitable K-12 computer science education*. Retrieved from https://code.org/files/State_of_the_states.pdf

Thompson, D., & Bell, T. (2013). *Adoption of new computer science high school standards by New Zealand teachers.* ACM.

von Wangenheim, C., Alves, N., Rodrigues, P., & Hauck, J. (2017). Teaching computing in a multidisciplinary way in social studies classes in school: A case study. *International Journal of Computer Science Education in Schools, 1*(2), 3-16.