# The impact of an issue-centered problem-based learning curriculum on 6th grade girls' understanding of and interest in computer science

By: Anne T. Ottenbreit-Leftwich, Kyungbin Kwon, Thomas A. Brush, Michael Karlin, Minji Jeon, Katie Jantaraweragul, Meize Guo, Hamid Nadir, Fatih Gok, Parama Bhattacharya

Ottenbreit-Leftwich, A., Kwon, K., Brush, T., Karlin, M., Jeon, M., Jantaraweragul, K., Guo, M., Nadir, H., Gok, F., & Bhattacharya, P. (2021). The impact of an issue-centered problem- based learning curriculum on 6th grade girls' understanding of and interest in computer science. *Computers and Education Open*, doi: 10.1016/j.caeo.2021.100057

**Made available courtesy of Elsevier: http://dx.doi.org/10.1016/j.caeo.2021.100057**



## Abstract:

Many stakeholders have suggested that if girls are not engaged in computer science at an early age, the field of computer science (CS) (and computing-related fields) will continue to lack gender diversity. We need more explorations of how curriculum impacts girls' interest in computer science at the elementary level. The focus of this research project was to examine how an elementary problem-based learning (PBL) curriculum impacted girls' understanding of and interest in CS. By utilizing a curriculum focused on a problem that emphasized social activism, we hypothesized that PBL CS could increase interest in CS for girls. This study used a mixed-method sequential explanatory research design to examine the implementation of a CS PBL curriculum with four 6th grade teachers and their 263 students. We collected student CS knowledge pre-tests and post-tests, student attitude surveys, student focus group interviews, teacher interviews, and researcher observation reflection notes. The knowledge tests were analyzed using a mixed measures ANOVA. The survey responses were analyzed using ANOVAs and effect sizes for each comparison were calculated using partial eta squared. The qualitative data was analyzed using a thematic approach that required four researchers to come to a consensus. Results suggested that although girls had less understanding of CS at the beginning, by the end of the curriculum, girls and boys performed similarly. However, there were still elements that were not successful. Girls' attitudes towards computer science were significantly lower than the boys at the end of the curriculum. Conceptually, the curriculum needs to specifically address how CS could impact values held by girls

**Keywords:** Elementary education | Gender studies | Teaching/learning strategies

## Article:

## 1. Introduction

There is a current underrepresentation of women in computer science (CS) [40]. This underrepresentation is problematic not only from an equity perspective (e.g., Education Development Center, 2017) but also from an innovation and workforce perspective (e.g., [14]; EDC, 2017). When women are not involved in CS, the gender imbalance in CS can bring negative social, economic, and scientific consequences [39]. Scholars have indicated that the lack of female engagement in computer science is often due to (a) masculine culture identities, (b) a lack of sufficient early experience, and (c) gender gaps in self-efficacy [12]. High school girls are significantly less likely to take computer science classes than high school boys and girls start to lose interest in STEM and CS-related fields at the late elementary/middle school grades [12]. Furthermore, in order to broaden participation, students need "culturally relevant curricula that support diverse ways of approaching CEd and diverse ways of expressing one's knowledge" ([4], p. 66). Previous studies have shown that girls tend to value CS experiences that have social and/or societal impacts (e.g., [50]). Therefore, with these perspectives in mind, we sought to explore how to build a relevant curriculum that might engage girls in CS at this critical age.

### 1.1 Why is diversity critical in CS?

According to the U.S. Bureau of Labor Statistics (BLS) Employment Projections for 2010–2020, more than half of the anticipated STEM jobs that will be created in this country will be in computing [49]. Other projections have indicated that jobs related to computing will generate half a million new jobs between 2014 and 2024 [25]. But not all these of jobs will be computer scientist positions. In fact, Kaczmarczyk, Dopplick, and the Education Policy Committee (2014) found that 63% of current computing jobs were in industries outside of computer science (CS), ranging from agriculture to automobile manufacturing. In addition, jobs outside of CS will still require basic CS skills [18,46]. These data suggest that we should also focus on establishing a broader workforce for diverse industries that has fundamental CS skills and knowledge as well as the capability to solve computing-related problems in different fields.

According to Blikstein and Moghadam [4], the necessity for computing education expands beyond the labor force. Through interviews with 14 computer science education experts, they identified four distinct rationales and positions for computer science education: (1) labor market, (2) computational thinking, (3) computational literacy, and (4) the equity of participation. In terms of the equity of participation, Blikstein and Moghadam described that many experts advocate for equity in CS education because knowledge of this area is "crucial for civic participation and informed decision-making" (p. 64). For example, as computer science is applied to a wide range of societal problems (e.g., the deployment of neighborhood policing), having diverse perspectives and understanding may equate to solutions that meet the needs of a broader population. In addition, the experts pointed out that those without an understanding of CS will be vulnerable to cybercrimes or manipulations with data science, and will likely have less access to technology influenced or adjacent job opportunities.

### 1.2. How gender diverse is computer science?

The CS workforce is not currently diverse and does not reflect the general population. According to data from the Bureau of Labor Statistics [21], although women represent 57% of professional occupations in the US workforce, they represent only 26% of professional computing occupations. The lack of gender diversity in CS continues from the workforce down to the post-secondary levels. For example, in 2016, women represented 19% of computer science bachelor's degrees, which was down from 37% in 1985 . According to the most recent NSF survey report on Women, Minorities, and Persons with disabilities in science and engineering, of all STEM fields, women represented the smallest percentage of post-secondary graduates in computer science and engineering [40].

What makes women become interested in CS? In one survey of 189 post-secondary CS-related female students, Malik and Al-Erman ([60]) asked students what factors influenced them to pursue computer science. The highest rated factors were focused on internal interests: 'Make the world a better place,' 'personal interest,' and 'personal abilities.' The authors suggest that female CS students wanted to learn CS to "make the world a better place" (p. 64). This persists even for younger girls. Bryant et al. [8] found that through middle school girls' participation in a summer camp designed around computing for social good, girls' self-efficacy around computing increased to become even with the boys' self-efficacy. In another study of 27 fifth through eleventh grade girls in a CS summer camp, Hur et al. [33] suggested that girls needed "explicit focus on the ways in which CS can be used to help others" (p. 115) since they value opportunities to solve problems that benefit social or societal situations.

### 1.3. Do female participate in CS in K-12?

Researchers have suggested that the low female participation in computer science is directly related to K-12 pipeline problems [22]. In fact, in 2019–2020, over 115,000 boys took an AP CS exam, while fewer than 50,000 girls took an AP CS exam; only 29% of those who took an AP CS exam were girls [15]. However, this number is slowly increasing each year; 31% of those taking AP CS exams in 2021 were girls [55]. K-12 schools have not been successful creating learning environments that encourage girls to pursue computer science. Girl participation in K-12 can influence them to pursue post-secondary computer science classes [22,48]. For example, Google [30] surveyed 1000 women and 600 men to examine why they chose to pursue a post-secondary CS degree. Results suggested that pre-college experiences in CS held the most influence over a women's decisions to pursue CS in college [30].

Few studies have investigated female interest and participation in computer science at the K-12 level. Research has suggested that an engaging and relevant curriculum can impact girls' decisions to pursue a career in CS. In a 2011 case study, Goode and Margolis examined the impact of the Exploring Computer Science (ECS) curriculum. The ECS curriculum has a large focus on incorporating a culturally relevant curriculum into the CS classroom. After initial pilot testing of the curriculum with 300 students (42% female), the authors found that the curriculum led to increases in students' perceptions of CS usefulness, their beliefs about the appeal of CS, their perceptions of CS as enjoyable, their motivation to preserver through difficult problems, and their likelihood to participate in CS courses in the future [29].

Venkataraman et al. [51] surveyed 127 students enrolled in a northeastern United States STEM high school and found that high school girls described barriers to entering CS such as perception, interests, confidence, and experiences. In a follow up survey study with second grade students at a STEM elementary school ([52]), the attitudes of second grade girl and boys were more consistent. In fact, when these two groups were contrasted, Venkataraman et al. [52] found that there were significant differences between second grade students and ninth grade students.

This may suggest that sometime between second and ninth grade, boys' and girls' perceptions of CS change. Some have suggested the gender gap between boys and girls in CS begins as early as 5th grade [48]. Therefore, we need to engage girls in CS experiences early in their K-12 schooling. Hur et al. [33] examined the impact of a three-day CS-focused summer camp on the career choices of 27 girls (5th-11th grade). Many of the girls in the study perceived that CS was difficult or 'geeky,' indicating that many of their female friends may not be interested in CS due to these stereotypes.

Scholars have attempted to investigate different ways of introducing CS in K-12 (e.g., [26,41]). For example, Franklin et al. [26] examined how 4th-6th grade students experienced various programming tasks with a visual block-based coding curriculum. Although students were able to understand the basic concepts, the more complicated CS concepts were challenging, especially to the younger students that had not yet mastered certain mathematical concepts.

Many of the elementary CS programs have utilized Scratch to engage students in CS activities. Resnick, the founder of Scratch stated that the newest version was designed for students to "design, create, experiment and explore with new technologies. They're willing to work hard on projects that connect with their interests and that provide opportunities for collaboration and expression" ([45], para. 12). However, the design of the learning activities, curriculum, and teaching practices are equally important to consider.

Several studies specifically focused on investigating whether various curriculum efforts have impact on girls' understanding of, interest in, and attitudes toward CS. For example, Tsan et al. [48] examined 5th grade students enrolled in an elective CS course, specifically investigating the gender compositions of collaborative groups and how this relates to student understanding. They found significant differences in the artifact quality based on gender composition – with all girl groups scoring significantly lower than other group compositions.

Other studies have shown differences in how girls and boys approach the development of Scratch projects. Funke and Geldreich [27] examined 58 4th grade students and found that girls and boys created different types of programs. After examining the blocks used by boys and girls, the boys used 70 different blocks, while the girls used 40 different blocks. Based on the analysis of the different types of blocks used, it seemed that the girls tended to use blocks that belonged to a story development, whereas the boys tended to use blocks that belonged to game development. In another study, Hsu [34] examined 46 5th grade students Scratch projects and found that boys tended to use more built-in costumes, more diverse sensing blocks, and incorporated more diverse sensing blocks than girls. However, girls were more likely to integrate positive feedback, create their own costumes, and had significantly better use of looping in their programs. These two consistent studies show the importance of providing girls with options to create different types of programs (such as stories), incorporate their own costumes, and introduce them to a wider range of blocks.

Based on these studies, we conclude that elementary CS Scratch curriculum can engage female students if it includes relevant curriculum or problems with aspects of storytelling aspects, creativity, and options to create different types of programs. In addition, upper elementary CS curriculum should also incorporate elements of different career options and how CS can be used creatively by most careers in the future.

### 1.4. What curriculum models can increase girls' interest and understanding?

Extensive research conducted over the past decade has demonstrated that problem-based learning (PBL) can enhance both student engagement and students' academic achievement with challenging content in K-12 settings [7]. A number of meta-analyses focusing on the implementation of PBL in K-12 environments concluded that PBL instruction is more effective than traditional, teacher-centered instruction with regard to student achievement (e.g., [44,53]). Studies across different subject areas have shown that students participating in PBL instruction tend to be more engaged, have better academic performance [3,54] and that PBL instruction can appeal to girls and increase their interest in the subject area (e.g., [9]). In fact, in a case study of K-12 girls in science PBL learning, Buck and colleagues suggested that PBL was successful in increasing girls' engagement with science due to (1) technology-based approaches, (2) authentic and relevant contexts, and (3) interpersonal aspects associated with PBL.

Research examining the use of PBL to teach CS has demonstrated mixed results. Some studies have shown that if CS projects are too open-ended, students may have difficulties completing them [13]. For example, Dwyer et al. [23] implemented a project where 4th grade students integrated computational thinking into a physics project. Results of the study demonstrated that the students struggled with the CS portions of the curriculum due to a lack of prior background knowledge. The authors suggested that teachers integrate introductory block-based programming into their instructional activities. Therefore, the incorporation of CS knowledge needs to be thoughtfully integrated, especially at the elementary level. CS naturally presents student-centered PBL opportunities that can also contribute to social engagement [28]. For example, 5th grade students who were enrolled in the CSteach curriculum that explicitly focused on social justice issues, demonstrated increased attitudes towards CS and increased motivation to further pursue CS experiences in the future [20].

Other studies have shown that integrating inquiry-based instructional approaches such as PBL with CS instruction is one way to increase female interest in CS. There are several CS curricula that utilize an inquiry-based approach at the high school level (e.g., Exploring Computer Science, Mobile CSP), and research suggests that these approaches improve student engagement [32], student achievement [11,32], and student attitudes towards CS [32,42]. However, although there are some examples of CS curriculum at the elementary level that integrate inquiry-based instructional approaches (e.g., Project Lead the Way's Launch program), there has been little research examining whether this instructional strategy is successful with elementary-age students.

The purpose of this study was to examine whether a CS curriculum that integrated a problem-based learning instructional approach would positively impact girls' knowledge of and interest in CS. We collaborated with 6th grade teachers to develop an integrated PBL unit focusing on the socially-relevant problem: "How can we support a culture of kindness in our school?" By selecting a problem for that engaged students in social activism, we hypothesized that the unit could be particularly engaging to girls ([28]). We employed a sequential mixed-methods explanatory research approach to examine the following research question: To what extent does a computer science student-centered (problem-based learning) curriculum impact 6th grade girls' understanding of and interest in CS?

## 2. Description of the CS PBL curriculum

To develop the CS curriculum used in this study, we collaborated with four 6th grade teachers at four different elementary schools in the same Indiana school district. Our collaborative curriculum design efforts led to the development an initial curriculum that teaches

CS block-based coding in a shorter time frame than most other curricula and directly addressed Indiana CS K-8 standards. In addition, we incorporated a PBL activity that provided students with the opportunity to utilize their newly-acquired CS skills to address a socially relevant problem.

The first component of the curriculum consisted of a 10 h Block-Based Computer Science curriculum using Scratch to directly target the 6th grade standards. Utilizing frameworks from Brennan and Resnick [6], the CS Framework [17], and the Indiana CS K-8 standards, our curriculum focused on Computational Thinking (CT) concepts (Sequences, Loops, Event, Condition, Parallelism, Data, Operator) and CT practices (deciding topics, decomposing tasks, developing programs, demonstrating programs). The second component of the curriculum focused on a student-centered PBL activity that incorporated a social impact focus problem.

After developing the block-based coding curricular activities, we then collaborated with the teachers to develop a PBL curriculum that would allow students to apply their CS skills to a relevant problem. Our teacher partners chose to focus the PBL activity on the driving question: How can we create a culture of kindness in our school? The activity introduced concepts of kindness and heavily incorporating elements of the partner school district's recently adopted social-emotional curriculum. Students were scaffolded on how to conduct research on this topic, and provided with resources to support their research. Then, they used their new CS skills to collaboratively design and develop an “app” that would address the driving question based on their research. Students were provided with different types of apps that they could develop using Scratch: Choose Your Own Adventure Story, Quiz Game, or Interactive App. These choices were presented because based on previous literature, we were aware that girls and boys developed Scratch projects differently (e.g., [27]). Students were able to list their top choice for the app they wanted to develop. Teachers grouped students using this choice and other teacher-informed considerations (ability levels, working relationships, etc.). Groups were typically a mix of boys and girls, which precluded us from comparing the final projects based on gender.

By providing students with choices and an opportunity to impact their community, we believed this could potentially increase CS understanding and engagement, particularly with girls (e.g., [8,33]). Khan and Luxton-Reilly [37] stated that computing for social good may help close the gender gap in computer science by “taking students’ values into consideration for structuring the content, practising socially relevant first day activities and creating meaningful assignments and resources” (p. 4). Examples of the apps that students developed included a game that provided suggestions on ways to be kind to their peers, and a “kindness tracker” that recorded students’ acts of kindness throughout the day and provided words of encouragement to students. Previous studies had also shown that females tended to use more positive feedback within their Scratch projects, so we thought that this may increase girls’ engagement in the Scratch projects [34].

## 3. Method

### 3.1. Research context and participants

This study took place over the course of nine weeks. The participants for the study included four 6th grade elementary science teachers and 263 6th grade students in a school district in Indiana (see Table 1). The four teachers volunteered to participate in the collaboration and all their students were required to complete the CS activities. However, students were not required to participate in the research activities. There were over 300 6th grade students that completed the CS curriculum activities, but not all students completed all the research activities. Of the over 300 6th grade students that participated in the curriculum, only 263 students completed all the research activities and those students were included in the study.

Table 1. Demographics of school and teacher.

| Name | Elementary A | Elementary B | Elementary C | Elementary D |
|---|---|---|---|---|
| Teacher | • 14 yrs teaching<br>• 4 yrs robotics coach<br>• no CS background | • 26 yrs teaching<br>• no CS background | • 19 yrs teaching<br>• no CS background | • 18 yrs teaching<br>• no CS background |
| School | • Rural STEM school<br>• Title I school 56% FRL<br>• 6.5% multiracial, 3.8% Black, 7.1% Hispanic, and 82% white.<br>• 53 6th-grade students | • Rural school<br>• 31% FRL<br>• 5% multiracial, 1% Black, 2% Hispanic, and 93% white.<br>• 90 6th-grade students | • Suburban school<br>• 18% FRL<br>• 6.5% multiracial, 3% Black, 11% Asian, 4% Hispanic, and 75% white.<br>• 125 6th-grade students | • Suburban school<br>• 10% FRL<br>• 7% multiracial, 3% Black, 11% Asian, 4% Hispanic, and 75% white.<br>• 30 6th-grade students |

Although all of the participating teachers were experienced teachers, none had any CS experience or had taught CS concepts to their students. One teacher supervised the schools robotics club, but no teachers had received formal CS professional development. All four teachers also had limited experience integrating problem-based learning strategies into their instruction.

Each teacher was paired with a coach experienced in teaching CS at the elementary level. These coaches would model and/or co-teach lessons with the teachers for the block-based coding lessons. While teachers were provided with the materials in advance, most of them indicated that their most significant professional development occurred during the modeling/co-teaching sessions.

### 3.2. Design

This study employed a sequential mixed-methods explanatory approach ([59]). We applied this explanatory approach to examine students' experiences to discover understandings of the curriculum intervention with the goal of re-thinking the design or approach. With this approach, we could expand on the relationship between gender and CS interest/understanding. We collected both quantitative data (tests, surveys) and qualitative data (interviews, observations, and reflections), and specifically utilized the qualitative data to help explain the findings of the quantitative data particularly in relation to students' understandings and interests in CS.

### 3.3. Data sources

Data sources collected in this study included a student CS knowledge pre-test and post-test, student attitude survey, student focus group interviews, teacher interviews, and researcher observation reflection notes. Each of these sources is described in more detail below.

#### 3.3.1. CS knowledge pre-test and post-test

The researchers and teachers collaboratively developed a 14-item multiple-choice assessment designed to measure students' knowledge of basic computer science and coding principles. These were developed based on CS concepts [6,17,35]. Eight of the test items focused on basic computer science concepts (sequences, loops, event, conditional, operators, variables, parallelism), and six of the items focused on CS practices (debugging of the eight concepts). Parallel forms of the test were administered to students on three occasions: prior to the beginning of the unit (pre-test), immediately after the completion of the block-based coding portion of the unit (post-test 1), and immediately after the completion of the PBL portion of the unit (post-test 2). The pre-test and post-test 1 were delivered 3 weeks apart in the Fall of 2018. Post-test 2 was delivered in the Spring of 2019. See Fig. 1 for an example of the format of the test.

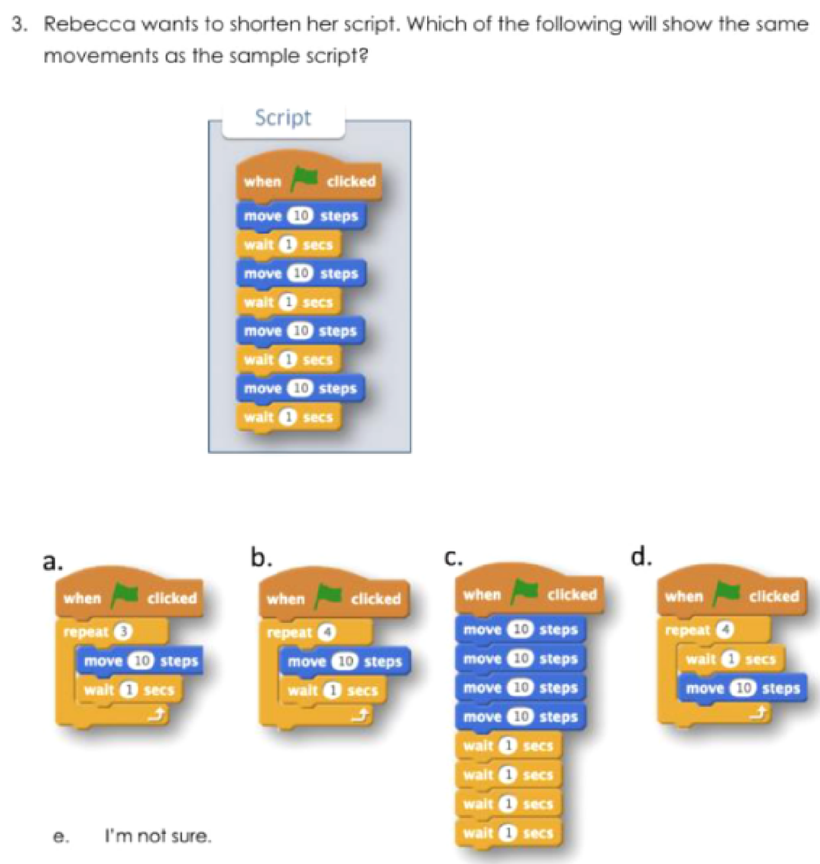


**Fig. 1.** Excerpt from CS knowledge test.

3.3.2. Student attitude survey

This 16-item closed-ended survey was administered to students after the end of unit activities and was designed to measure: (1) Self-efficacy regarding programming (4 items, 2) Computational Thinking skills (6 items); and (3) Attitude toward CS (6 items). The items were adopted from literature and modified in consideration of the characteristics of target students [10,43]. Cronbach's alphas for scales were calculated and suggested high internal consistency reliability, such as self-efficacy, CT skills, and attitude were 0.83, 0.90, and 0.84, respectively. See Fig. 2 for a sample excerpt from the survey.


**Fig. 2.** Excerpt from student attitudinal survey.

3.3.3. Student interviews

Student interviews were conducted with 27 students in four different focus groups. Thirteen of the students participating in the focus group interviews were girls. Focus group interviews were used to stimulate discussion and prompt cascading conversations where students build upon each other contributions [31]. This was particularly important in this study, since the intervention was implemented over several months and students may have needed additional prompting from their peers to address specific interview questions.

The teachers were asked to select students that would represent a range of gender and ability levels. Focus groups ranged from 4–6 students, and each group had equal representation of boys and girls. Each interview lasted approximately 30 minutes. Students were asked questions related to their content knowledge and interest in CS (e.g., “What are your thoughts about the work that we've done with Scratch?,” “Which activities did you find the most difficult? Why?,” “How did you and your team solve problems you ran into?,” and “Do you think you would ever want to learn more about computer science? Why or why not?”).

3.3.4. Teacher interviews

Pre- and post-interviews were conducted with each teacher to identify the successes and challenges of the curriculum, specifically as it related to students' CS interests and understandings. Questions included "Do you think that students were receptive to the coding activities in class? Did it seem like they mattered to them? Why or why not?" and "Do you think the computer science lessons have been successful? Why or why not?" Each teacher interview lasted approximately 30 minutes.

#### 3.3.5. Observation reflection notes

In each classroom, there was at least one researcher present as a participant-observer for every class. In this capacity, the researchers observed the teachers' interactions with the students, as well as provided some assistance to the teacher to help answer students' questions when needed. After each class, the researcher would write a general reflection on the observation of the lesson for that day. Each reflection focused on how students responded to the curriculum, specifically focusing on student comprehension of the CS content, student collaboration and construction during the app design and development, and student engagement/interest in the curriculum.

### 3.4. Procedure

The primary activities for the initial and final days of the unit were the administration of the pre- or post-test. For the other class sessions, students completed specific activities to build their CS concepts and practices. During the final week of the unit, students developed culminating presentations in which they presented their "apps" that addressed the driving question for the unit (see Table 2 for an outline of the unit implementation). Each day of the unit was observed by one researcher. Teacher post-unit interviews were conducted after the completion of the unit.

### 3.5. Data Analysis

In terms of the knowledge test, a mixed measures ANOVA was conducted with gender and assessment time (pre-test, post-test I, post-test II) as the independent variables. This analysis provided the opportunity to examine the extent to which groups improved over time, as well as whether a specific group improved significantly more over time compared to another group (e.g., gender x time interaction). To examine differences between gender in the survey responses, ANOVAs were conducted on each survey subscale. Effect sizes for each comparison were calculated using partial eta squared [16].

Qualitative data sources (e.g., interviews, observations) were analyzed by researchers for trends and patterns related to the research question. These trends and patterns were represented by codes and themes that emerged directly from the data [5]. After transcribing the data, a team of four researchers reviewed all the data and discussed potential codes. Once they had reached an agreement on codes and definitions, they categorized all data and presented these codes to each other. When disagreements occurred, all four researchers came to consensus on one code before continuing [47]. Based on our research questions, the codes were grouped into two main themes to focus on the students' understanding of and interest in CS. By conducting initial coding schemes independently and then coming to consensus as a group, we were able to reduce the presence of ambiguity or bias in our coding process. This contributed to the trustworthiness of our data analysis [1]. After separating the codes into the two distinct areas (understanding and interest), we examined the main themes that emerged, paying close attention to any differences between the girls and boys. We did not recognize any emerging differences from the boys' and girls' interview responses, nor from the teachers' responses. We used the interview responses to help support and clarify our results ([59]).

**Table 2.** Pilot CS curriculum week-by-week topics, activities, and Indiana CS standards addressed.

| | | |
|---|---|---|
| Week 1: CS Introduction & Foundations | Fall 2018 Activities | CS Standards |
| Students are introduced to the basic ideas of computer science, hardware, software, and computer components. | Videos and discussions on relationship with humans and machines. Unplugged activities on binary and communication. | 6–8.CD.1<br>6–8.CD.4<br>6–8.IC.2 |
| Weeks 2–4: Extending CS Knowledge | Fall 2018 Activities | CS Standards |
| Students are introduced to Scratch and the functions of different Scratch blocks. | Students create at least 5 programs in Scratch (e.g., a dance party, a maze, a quiz game, and a variables game, and functions). Final project has students create their own program incorporating these ideas. | 6–8.DI.1<br>6–8.CD.2<br>6–8.PA.2<br>6–8.PA.3<br>6–8.NC.2 |
| Week 5: Contextualizing the Problem | Spring 2019 Activities | CS Standards |
| Students introduced to the PBL problem of "How can we create a culture of kindness in our school?" | Videos and discussions on creating a culture of kindness. Students research how acts of kindness in their school and daily lives can contribute to a culture of kindness. | N/A |
| Weeks 6–7: Research and Design | Spring 2019 Activities | CS Standards |
| Students design and develop | Students spend 8 lessons | 6–8.DI.1 |

| | | |
|---|---|---|
| an "APP" using Scratch to address the PBL problem. | researching, planning, designing, and developing their Scratch project. Multiple check-ins and scaffolds help support development | 6–8.CD.2<br>6–8.PA.2<br>& 3<br>6–8.NC.2 |
| Week 8: Presenting | Spring 2019 Activities | CS<br>Standards |
| Students present their final Scratch projects. | Students present and share their final projects with their peers, teacher, other students, and external visitors | 6–8.NC.1<br>6–8.IC.1 |

**4. Results**

4.1. Student understanding and ability

4.1.1. Student pre-post test results

As previously discussed, the CS knowledge test assessed students' understanding of CS concepts (sequences, loops, event, conditional, operators, variables, parallelism) and debugging capabilities. This study sought to investigate whether any differences existed between girls' and boys' understanding of CS. Tables 3 and Fig. 3 present the mean scores of pre-test, post-test 1, and post-test 2 for CT concepts and debugging skills by gender.

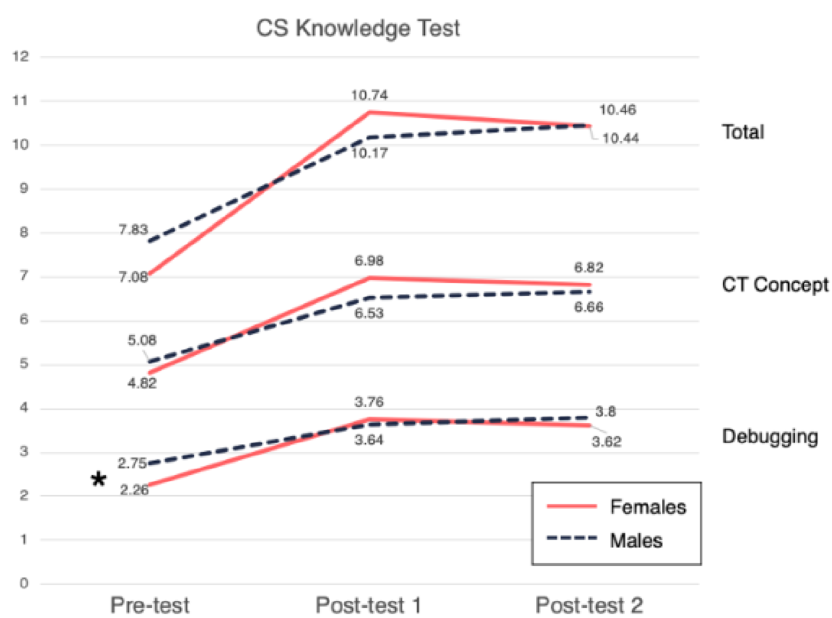


**Fig. 3.** CS knowledge tests based on genders.

The results of a mixed ANOVA showed that students' overall test scores improved significantly, F (2, 392) = 220.98, p< .001, $\eta_p^2$ =0.53. Regarding the difference between gender, no significant main effect was confirmed. There was a significant interaction between the times of tests and the gender of students, F (1, 196) = 4.87, p= .03, $\eta_p^2$ = 0.02. Post-hoc tests comparing adjacent times revealed a significant main effect of time between pre-test and post-test 1 (after Scratch lessons), F(1, 196) = 313.28, p < .001, $\eta_p^2$ = 0.62, and a significant interaction effect of gender and time on scores, F(1, 196) = 15.20, p < .001, $\eta_p^2$ = 0.07, but not between the post-test 1 and 2. On the pre-test, a marginally significant difference between genders was confirmed. Boys (M = 7.83) demonstrated higher scores than girls (M = 7.08), F(1, 196) = 3.49, p = .06, $\eta_p^2$ = 0.02.

We also examined the test scores based on two sections of the test: concepts and debugging. Regarding concepts, the results of a mixed ANOVA revealed significant main effects of time, F(2, 392) = 167.90, p < .001, $\eta_p^2$ = 0.46. Regarding the difference between gender, no significant main effect was confirmed. Post-hoc tests comparing adjacent times revealed a significant main effect of time between pre-test and post-test 1 (after Scratch lessons), F(1, 196) = 222.57, p < .001, $\eta_p^2$ = 0.53, and a significant interaction effect of gender and time on scores, F(1, 196) = 8.43, p = .004, $\eta_p^2$ = 0.04, but not between post-test 1 and 2. In post-test 1, a marginally significant difference between genders was confirmed. Girls showed higher scores (M = 6.98) than boys (M = 6.53), F(1, 196) = 3.42, p = .07, $\eta_p^2$ = 0.02.

Regarding debugging skills, the results of a mixed ANOVA revealed significant main effects of time, F(2, 392) = 93.00, p < .001, $\eta_p^2$ = 0.32. Regarding the difference between gender, no significant main effect was confirmed. Post-hoc tests comparing adjacent times revealed a significant main effect of time between pre-test and post-test 1 (after Scratch lessons), F(1, 196) = 141.75, p < .001, $\eta_p^2$ = 0.42, and a significant interaction effect of gender and time on scores, F(1, 196) = 9.52, p = .002, $\eta_p^2$ = 0.05, but not between post-test 1 and 2. On the pre-test, a significant difference between gender was confirmed. Boys (M = 2.75) showed higher scores than girls (M = 2.26), F(1, 196) = 5.16, p = .02, $\eta_p^2$ = 0.03.

Discussion of test results

Student test results suggest that participation in the unit had a positive effect on students' knowledge of CS concepts and practices – particularly with regard to debugging skills. Overall, there was a significant improvement in students' knowledge of basic CS concepts and programming, and these gains were sustained after completion of the PBL portion of the unit.

There were significant differences between the boys' and girls' scores on the pre-test. However, these differences were not sustained with the post-test scores after both the CS portion of the unit and the PBL portion of the unit. The significant interaction between gender and the initial pre-test scores may suggest that the CS unit activities reduced the initial knowledge gap between the boys and the girls. In addition, the fact that the gains in girls' CS knowledge and skills (as measured by the post-test) were sustained after completion of the PBL unit activities suggests that the PBL activities may have provided additional opportunities for students to apply their knowledge in a meaningful way, which may have facilitated their retention of the knowledge. It is important to note that most of the students did not complete the PBL portion of the unit until approximately three months after the initial CS unit activities. This delay in students' application of their CS knowledge to address the PBL unit problem makes post-test results for students after completion of the PBL portion of the unit even more encouraging.

However, further research is necessary to determine in what ways the PBL unit activities may have engaged girls in their application of CS knowledge and thus provided the means for them to retain that knowledge throughout the unit. By further examining how students used CS concepts to develop a program that addressed an authentic and socially-relevant issue, we may reveal the effects of PBL lessons more accurately.

#### 4.1.2. Student interviews

##### Curriculum experiences

During student interviews, different students discussed which components of the curriculum were most challenging and what they were able to accomplish. In general, both boys and girls seemed to have a wide range of activities that were most challenging to them. For example, one boy mentioned that the initial coding activity was the most difficult: "The beginning maze game, the dancing one, I did awful at, and I didn't really get it. But the beginning maze game, it was simple and something that people would want to do because it was pretty fun." In another example, one girl described a jumping activity as the most challenging: "We had to make the sprite jump over the other sprite. We had to time it correctly and if we didn't have the measurements right it wouldn't work." There were a wide range of responses to this question; there were no discernible gender differences.

Both girls and boys explicitly stated that they were initially challenged by the idea of coding, but the lessons made them more knowledgeable about CS. For example, one girl explained that "When I didn't know how to code, I thought it was really difficult and hard to understand. Then with the lessons, I got more knowledge of it, and it helped me." Another girl mentioned "I learned a lot. It was hard for me at first, but I learned now to code." In another example, a boy described that the lessons helped build his CS knowledge: "I kind of feel like all of [the lessons] were really helpful…we learned a few things at a time, and then the last one we got to see how it all comes together."

When students were asked what part of the curriculum helped them learn the most, one girl specifically mentioned the app creation: "I would say probably the culture of kindness because that's the [activity] we did work on for a longer period of time. We pulled everything we had into one big project. So that was probably most influential learning experience." In the final

PBL app creation project, we did not see any differences between boys' and girls' CS knowledge. Student teams for the culminating CS PBL app usually ranged between 2 and 4 students. Some were mixed gender, some were all girls, and some were all boys. It seemed that student personalities were more indicative of who performed the most coding. For example, in one mixed gender team, one girl explained that she did most of the programming work: "me and [the boy in my group] agreed that he would do the art stuff. And I would do all the coding. All the coding sounds like a lot more, but we had at least 50 costumes that had to be edited." The 'art stuff' included the costumes, backgrounds, and development of characters. In another mixed gender team, one boy said: "I did my best to take the lead because nobody else in my group was in Lego Robotics, so they didn't have as much knowledge, and I did my best to so as much work as I could so they didn't have to do as much." This was fairly typical throughout the student interviews, indicating that in many teams, certain students were primarily responsible for the coding aspects of the project. However, both genders were represented as those primarily responsible for coding; it may suggest that those responsible for coding felt more confident or comfortable, but this would require more focused data.

Need more time and practice

Both girls and boys mentioned they wanted more time with Scratch, as they felt that they still needed to improve their CS knowledge. Several girls specifically mentioned that they needed more time and practice to get familiar with coding: "If we would have explored the pieces to make code more, I feel like the games that we made would have had more detail, and it would have been better because we would be more comfortable with Scratch" and "if we started it earlier in the year, then we can get more familiar with Scratch, and then we could learn more about it." For most of the students, this was their first experience with coding and Scratch. However, several of the boys in each school had been involved in after school robotics clubs, while only one girl at School A was participating in robotics. This could be one explanation as to why girls had lower average pre-test scores, but this still requires additional focused data. Perhaps as coding and CS becomes more pervasive at the younger grades, this may be a less common concern.

4.1.3. Teacher interviews and researchers observation reflections on understanding

All students gained knowledge

Teachers were asked whether they saw any student group differences in student performance and understanding. Teachers stated that they believed the curriculum enabled all students to be successful in gaining CS knowledge. For example, Teacher C stated: "Overall, we had a lot of participation, and we had a lot of interest, and I think there's a lot of learning going on". Teacher C also mentioned how students were able to apply information learned during the lessons to showcase their understanding of CS: “[the lessons] were successful because the kids were successful finishing the game, learning the skill, and then applying it next week. When they got to make their own projects, they saw a lot of those things being repeated.” In the researchers' observations, they also noted that “all students in the classroom were able to replicate the block-based coding curriculum activities” (Researcher B).

Gender differences in understanding

None of the teachers mentioned a difference in student understanding based on gender during their post-unit interviews. Two teachers were explicitly asked if there were differences between boys' and girls' performance, as a follow-up prompt. One teacher stated: “My girls surprised me. I had several that I could really see doing this for a living, being a coder, you know?... I'm thinking of one in particular that's really quiet and meek that was a rockstar” (Teacher B). Classroom observation data indicated that students were all able to complete the coding tasks from the block-based coding initiative. Researchers observed no explicit gender differences in student capabilities during the Fall individual structured coding experiences. When teachers were presented with the pre-test results showing that girls scored significantly lower than the boys, one teacher suggested that this could have been due to the fact that “…it seems like 6th grade boys play more video games than 6th grade girls (by a lot). That is the only thing I can think of to explain it” (Teacher D).

However, researchers B, C, and D specifically observed that during teachers' demonstrations during the block-based coding curriculum, boys were more interactive and asked more questions. For example, in classroom D, when a teacher was modeling how to make a quiz game, a boy asked about the concept of initialization. Another boy volunteered to answer the question saying, they could add an event block to make the operation that enables variable to become zero. The teacher complimented the students' contributions and expanded on these by describing initialization. Researcher D noticed that there were fewer such interactions with the girls. Other researchers' reflections also indicated that the boys in their classrooms were more vocal in their participation during class, often interjecting answers or ideas about the coding process. Therefore, although there seemed to be no differences in understanding, data suggests that boys may have had a higher level of engagement and participation in classroom coding activities, thus showcasing their CS knowledge.

4.2. Student interest (confidence) motivation, attitudes, disposition

4.2.1. Student attitude survey

The student attitude survey focused on three areas: (1) computer programming self-efficacy; (2) attitude toward CS; and (3) computational thinking. The survey revealed that students had middle ranges of self-efficacy toward programming (M = 3.7), attitude toward programming (M = 3.3), and CT confidence (M = 3.6). Regarding gender differences in the survey responses, there was no statistically significant difference between girls and boys in self-efficacy and CT. However, regarding attitudes towards programming, boys showed more positive attitudes (M = 3.43) than girls (M = 3.08), $F(1, 261) = 7.999$, $p = .005$, $\eta\frac{2}{p} = 0.03$ (see Table 4).

4.2.2. Student interviews

In the student focus group interviews, students described specific activities that they believed were most engaging. Many girls mentioned that they enjoyed the flexibility and creativity within Scratch: "what I liked about the whole Scratch programming was you could look at other projects to help you get different ideas. It just was really creative. You didn't have one thing you had to do. You had baselines." Another girl specifically mentioned other creative aspects of Scratch such as "I like how you can customize what your sprite is." The open-ended activities seemed to also increase girls' interest in CS. For example, one girl described enjoying being able to incorporate their own creative ideas, "I liked making my own apps because we got to just come up with our own ideas," while another girl appreciated the ability to decide how many questions to include in a quiz game: "My favorite was the quiz game because you could do a lot of questions with it and you can make it really long, or you can make it really short." Several girls also described that they found CS interesting: "It's pretty interesting. It's just fun and a challenge." In general, the girls appreciated the choices with the project which seemed to encourage their interest in coding: "I liked making my own apps because we got to just come up with our own ideas."

Continue pursuing CS?

In the interviews, all students were asked whether the curriculum and PBL experience made them interested in computer science and whether they would want to learn more CS and/or enroll in another CS class. Most of the students said that this experience made them interested in learning more about CS. Two girls specifically mentioned that they had enrolled in a computer science class next year in middle school: "I'm taking a computer science class or something like that as one of my electives." One of these girls specifically mention this was due to her experiences with Scratch: "Yeah, I was thinking about it before but afterwards I got more interested, and I thought about it more." Although no other students specifically mentioned enrolling in CS classes for middle school, almost all of them indicated that they would be interested in learning more about computer science, as long as the project was something that was relevant to them.

Two girls specifically indicated that they would not be likely to take more computer science classes because it did not align with their interests or future career plans. For example, one girl mentioned that "I don't think I would [take a CS class] just because what I want to do whenever I grow up doesn't have anything to do with that. I want to be a pediatric surgeon.” Another girl mentioned that they would not take a CS class because it did not align with their interests: “Probably not, because I'm just not that type of person.” During student interviews, both boys and girls described that there were certain types of people who liked technology or computer science. One girl specifically described the types of people that participate in computer science: “it depends on who you are, whether you're that really that technology-into person like [Student A] or if you are more of the outdoors type... like it is not something you are used to. It is cool, but I would rather not. It just depends on kind of who you are whether you really like it, or if it is cool, or if it is like ‘Nahhh’.”

Selecting a new topic for the app challenge might increase student interest

However, many of the students (both boys and girls) indicated that they did not like the final project's focus on kindness. One boy described this during the interviews:

“I would not take that [CS] class if I knew that we were going to have to do Culture of Kindness. I did not have a lot of fun doing that. But if we were doing something different, I liked coding and the games in the beginning did inspire me. Then once we did the Culture of Kindness it wasn't as good. I feel like if we would have changed the topic then I would definitely take another class.”

Another girl agreed with this statement. She was not interested in CS before starting the unit, and although using Scratch was interesting to her, the PBL focus did not increase her interest: “I was not very interested at first, and I really did not know much about it. Then when I saw Scratch, it was a lot cooler, and I still do it now. The Culture of Kindness topic was not the best.” Many students suggested that if they could have applied their Scratch knowledge to a more engaging problem, this would have increased their interest and engagement with CS. One of the girls described that “I feel like I would enjoy it a lot more if I liked the topic more, then I would definitely be able to say I know CS.” However, the interview results suggest that many students did appreciate the ability to design their own games and apps in general. In fact, many of the students (both boys and girls) mentioned that they used Scratch at home, separate from this project.

4.3. Teacher Interviews and researchers’ reflection notes

During the post-unit interviews, all of the teachers described the curriculum as highly engaging for most students: “They loved it. Great feedback from the kids” (Teacher B). Several teachers pointed out that their students asked to explore Scratch during their free time and even lunch: “I bet there was probably a third of both classes that got on Scratch and began to program a new game and start to kind of manipulate some of those blocks around. And so that's the biggest engagement, when you can get kids doing what we're doing, if they're doing it on their own, you've got them. You've got something.” (Teacher A).

One researcher in classroom A noted that the girls in particular seemed to be more interested in sensory features of programming. They recorded their voices and make a theme song that represents their friends, which were not the features presented during the lesson. They seemed more engaged in exploring sensory features and relationship with peers. Also, they wanted to share their progress with the other girls sitting next to them. However, researchers also noted that in many mixed gender and girl-only groups, girls were engaged and often leading their group coding sessions.

When asked to elaborate on why they thought students were engaging, teachers described that students are typically motivated by computers and technology, as well as the element of student choice and creativity. For example, Teacher B stated that "I think kids just love technology. And two, being able to create and the interactive aspect of it." In another example, Teacher C mentioned that "I think there was a lot of interest. I think you are going to have that anytime it is a new topic. Especially if they get some choice in which directions they take it. It seemed like everybody was participating to some degree."

Teacher D indicated that the driving question for the unit had a positive impact on the culture of her classroom: "I thought that the culture of the classroom did change…you just saw them rallying together, and pulling each other along, and problem-solving. So it was great." Teachers A and C also mentioned that the culture of kindness seemed to impact students' overall kindness in the classroom. However, all teachers indicated that the topic may not have been engaging or relevant to their students. Teacher D described how the topic was not motivating to all students:

"some of the girls were really into [the app creation], but a lot of them weren't. They were more into the storytelling and into just creating the sprites. Some boys were not into it. I thought some of that was just the culture of kindness kind of thing... It just did not motivate them enough to really get into it. I could see if they had a bigger choice of what to do, maybe they would have got into it more"

## 5. Discussion

Many scholars have indicated that female engagement in CS is low and it is critical to engage girls in CS at an early age [12]. Studies have often referenced the importance of relevant curriculum, that engages girls through applicable problems where they have choices in how they solve those problems [32]. This study sought to examine one CS curriculum design that utilized a student-centered (problem-based learning) approach to potentially impact 6th grade girls' understanding of and interest in CS. We discuss the implications of the results as they related to girls' (1) understanding of CS and (2) interest in CS.

### 5.1. Girls' understanding of CS

These results suggest that a problem-based CS curriculum can effectively increase elementary girls' understanding of CS. This curriculum enabled students to learn about basic CS concepts initially through an individual block-based coding unit, and then by designing a larger-scale CS project that addressed a specific problem. While other CS curricular initiatives have been implemented at the secondary level, this research provides evidence suggesting that elementary students can engage in higher-level CS activities that involve designing, programming, and implementing more large-scale projects. In addition to the successful assessments, teachers perceived that all the elementary students were able to demonstrate successful understanding of CS concepts and were able to apply these CS principles to support real-world problems. Teachers did not notice any express differences between boys' and girls' understanding of CS.

However, even though the girls improved in their understanding of CS concepts and maintained these abilities through both post-tests, there were still some issues associated with students' understanding. The researchers in each classroom all observed instances of gender marginalization. For example, in several classrooms, when mixed gender groups were asked to present their apps, typically the boys presented. In addition, researchers observed that when it came to asking questions during the block-based coding lessons, boys were typically interjecting and answering questions. Other studies have shown many other studies suggest that women contribute less than men to technical content or to group discussions in team engineering projects (e.g., [57]). In an engineering education PBL study, Hirshfield and Koretsky ([58]) recommended allowing students to self-select their teams. They suggested that self-selected teams could lead to more balanced participation and discourse as students would have already built a relationship with one another.

5.2. Girls' interest in CS

Results of this study suggest that the curriculum created a space where girls increased their understanding of CS. However, our attempt to increase girls' interest in CS was not altogether successful. First, although girls indicated that they enjoyed the block-based coding game instruction and app creation activity, most were not as engaged by the topic of using their CS knowledge to address the driving question focusing on creating a culture of kindness.

In the next iteration, teachers decided to examine the culture of kindness PBL curriculum and decided to consider a more engaging problem that would resonate with girls. The teachers changed the guiding question to "How can I help others regulate their emotions?" focusing more on mental health and emotional intelligence. This problem also allowed teachers to incorporate more of their social-emotional learning curriculum that was mandated for the district. Preliminary feedback from teachers have indicated that this small switch seems to have engaged more students.

Previous research has suggested that incorporating problem-solving skills and re-orienting CS curricula around relevant and meaningful problem-solving can help broaden participation (e.g., [8,20]; [56] [37]). This incorporation of relevant, real-world problem-solving can help broaden participation by shifting students' understanding of the type of work that is done in CS and the value that work can have [8,37]. In other words, when CS is centered around relevant, real-world problems, more students are able to make connections between CS content and their own lives.

One option we would suggest is to modify the culminating activity to allow for more student choice and options. Many studies have shown that student choice and personalized learning experiences can help increase student engagement and achievement in the CS classroom [2,19]. This is especially true for underrepresented populations [29]. We plan on working with the teachers to determine methods to incorporate more student choice in the deliverables and activities. We provided options for stories, games, and interactive apps, which have previous shown to be engaging for boys and girls (e.g., [27]). Perhaps if teachers allowed students more flexibility in the types of apps they created, students might be more motivated to create an app that would impact their peers.

Results of the student attitude survey suggested that girls rated their attitudes towards CS significantly lower than the boys. This was further illustrated in student interviews. While some girls indicated they would be interested in taking computer science classes in the future, many described how certain types of students were more well-suited to computer science. Women and girls' self-perceptions of their own abilities and fit within CS typically has a strong impact on their decision to pursue a CS-related career [33]. Hur et al. [33] suggested that once girls had higher self-efficacy, they were more likely to see the relevance of CS in their future career choices. The importance of engaging women in K-12 CS courses was also emphasized by the National Science Foundation's Jan Cuny in her 2012 CS call to action: "Without an engaging computing course, women move on to college with no experience that contradicts the popular misconceptions of computing as a tedious, geeky, male endeavour with no social context and no relevance" ([55], p. 33). Therefore, we will suggest incorporate career options that will appeal to a wider range of students. This was not something that was previously incorporated into the curriculum, and needs to be addressed to help students, especially girls, see the relevance of computer science to their futures.

One of the planned improvements for the next iteration of the unit will focus on collaborating with the teachers to specifically target these assumptions about who participates in CS. One improvement may be to provide additional focus regarding how CS can be used in a wide range of disciplines and professions. For example, the Exploring Computer Science (ECS) curriculum was explicitly designed to broaden participation CS by challenging existing biases and assumptions about who can and cannot participate in CS [38]. Perhaps integrating some of the strategies utilized in the ECS curriculum can be integrated into the next iteration of this project.

## 6. Conclusion

The focus of this research project was to examine how an elementary problem-based learning curriculum impacted girls' understanding of and interest in computer science. By focusing on a problem that emphasized social activism, we hypothesized that PBL CS could increase interest in CS for girls. Results suggested that although girls had less understanding of CS, by the end of the curriculum, girls and boys performed similarly. However, there were still elements that were not successful as girls' attitudes towards computer science were significantly lower than the boys.

In the future, we plan on additional collaborations with the teachers to revise the curriculum. During our revision process, we plan on specifically trying to address engaging girls, particularly on how we can improve girls' attitudes toward computer science. In addition, since we only collected students' CS attitude data at the end of the curriculum, we also plan on collecting this before beginning the curriculum to see if it increased girls' attitudes (even though they were still lower than the boys).

For future iterations, we plan on experimenting with the topic, as well as the culminating activities and grabber activities to more specifically appeal to our girls. We suggest that this could be achieved by providing more student choice. This will become easier as the teachers implement the CS curriculum into their classrooms. As this was the first time these teachers had taught this CS curriculum, as they become more comfortable, they may be more open to expanding student choice and options. We also want to incorporate curricular elements that showcase how CS will be relevant to their futures. In the interviews with the girl students, most indicated that due to their intended career choices, CS would not be relevant or useful. This misunderstanding needs to be challenged throughout the curriculum and in the classroom, as CS will likely touch a range of careers and societal interactions Table 3.

**Table 3.** Means and standard deviation of test scores by gender.

| | Pre-test | Post-test I | Post-test II |
|---|---|---|---|
| CT concepts | 4.95 (1.78) | 6.74 (1.70) | 6.73 (1.57) |
| Girls | 4.82 (1.65) | 6.98 (1.36) | 6.82 (1.41) |
| Boys | 5.08 (1.90) | 6.53 (1.94) | 6.66 (1.71) |
| Debugging | 2.52 (1.54) | 3.70 (1.50) | 3.72 (1.68) |
| Girls | 2.26 (1.51) | 3.76 (1.33) | 3.62 (1.63) |
| Boys | 2.75 (1.54) | 3.64 (1.64) | 3.80 (1.73) |
| Total | 7.47 (2.85) | 10.44 (2.82) | 10.45 (2.84) |
| Girls | 7.08 (2.71) | 10.74 (2.33) | 10.44 (2.67) |
| Boys | 7.83 (2.93) | 10.17 (3.19) | 10.46 (2.98) |

Note. M(SD), Number of participants: girls = 93 and boys = 105.

**Table 4.** Means and standard deviations of survey responses by gender.

| | Self-efficacy | Computational Thinking | Attitude toward programming |
|---|---|---|---|
| Girls | 3.66 (0.82) | 3.47 (0.87) | 3.08 (0.92) |
| Boys | 3.81 (0.88) | 3.65 (0.95) | 3.43 (1.05) |
| Total | 3.74 (0.85) | 3.56 (0.92) | 3.26 (1.00) |

Note. M(SD), Number of participants: girls = 131 and boys = 132.

**Funding**

This work was supported by a Google CS-ER 2018 Grant.

**Declaration of Competing Interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**References**

1. Anfara VA, Brown M, Mangione TL. Qualitative analysis on stage: making the research process more public. Educ Res 2002;31(7):28–38.
2. Azcona D, Hsiao IH, Smeaton AF. Personalizing computer science education by leveraging multimodal learning analytics. In: Proceedings of the IEEE Frontiers in Education Conference (FIE). IEEE; 2018. p. 1–9.
3. Belland BR, Glazewski KD, Richardson JC. Problem-based learning and argumentation: testing a scaffolding framework to support middle school students' creation of evidence-based arguments. Instr Sci 2011;39(5):667–94.
4. Blikstein P, Moghadam SH, Fincher SA, Robins A. Computing education: Literature review and voices from the field. The Cambridge handbook of computing education research. Cambridge University Press; 2019. p. 56–78.
5. Braun V, Clark V. Using thematic analysis in psychology. Qual Res Psychol 2006;3(2):77–101.
6. Brennan K, Resnick M. New frameworks for studying and assessing the development of computational thinking. In: Proceedings of the 2012 Annual Meeting of the American Educational Research Association. Canada: Vancouver; 2012.

7. Brush T, Saye J. Successfully implementing problem-based learning in classrooms: research in K-12 and teacher education. West Lafayette, IN: Purdue University Press; 2017.
8. Bryant C, Chen Y, Chen Z, Gilmour J, Gumidyala S, Herce-Hagiwara B, Rebelsky SA. A middle-school camp emphasizing data science and computing for social good. In: Proceedings of the 50th ACM Technical Symposium on Computer Science Education; 2019. p. 358–64.
9. Buck GA, Beeman-Cadwallader NM, Trauth-Nare AE. Keeping the girls visible in K-12 science education reform efforts: a feminist case study on problem-based learning. J. Women Minor Sci Eng 2012; 18(2).
10. Cetin I, Ozden MY. Development of computer programming attitude scale for university students. Comput Appl Eng Educ 2015;23:667–72.
11. Chen CH, Yang YC. Revisiting the effects of project-based learning on students' academic achievement: a meta-analysis investigating moderators. Educ Research Rev 2019:71–81. 26 (November 2018).
12. Cheryan S, Ziegler SA, Montoya AK, Jiang L. Why are some STEM fields more gender balanced than others? Psychol Bull 2017;143(1):1.
13. Cliburn DC, Miller S. Games, stories, or something more traditional: The types of assignments college students prefer. ACM SIGCSE Bullet 2008;40(1):138–42.
14. Code.org, CSTA, & ECEP Alliance. 2019 State of Computer Science Education 2019.
15. Code.org, CSTA, & ECEP Alliance. State of computer science education: Illuminating disparities 2020.
16. Cohen J. Eta-squared and partial eta-squared in fixed factor ANOVA designs. Educ Psychol Meas 1973;33(1):107–12.
17. Computer Science Framework. K-12 Computer Science Framework 2017.
18. DeLyser LA, Mascio B, Finkel K. Introducing student assessments with evidence of validity for NYC's CS4All. In: Proceedings of the 11th workshop in primary and secondary computing education. ACM; 2016. p. 17–26. October.
19. Deng Y, Lu D, Chung CJ, Huang D, Zeng Z. Personalized learning in a virtual hands-on lab platform for computer science education. In: Proceedings of the 2018 IEEE frontiers in education conference (FIE). IEEE; 2018. p. 1–8. October.
20. Denner J, Martinez J, Lyon LA. Computing for the social good: engaging Latino/a students in K-12. ACM SIGCAS Comput. Soc. 2015;45(2):31–2.
21. Bureau of Labor Statistics. Labor Force Statistics from the Current Population Survey 2020.
22. DuBow W, Kaminsky A, Weidler-Lewis J. Multiple factors converge to influence women's persistence in computing: a qualitative analysis. Comput. Sci. Eng. 2017 Copublished by the IEEE CS and the AIP May/June 2017.
23. Dwyer H, Boe B, Hill C, Franklin D, Harlow D. Computational thinking for physics: programming models of physics phenomenon in elementary school. Engelhardt,

Churukian, & Jones (Eds.). In: Proceedings of the PERC. College Park, MD: American Association of Physics Teachers; 2013. P. 133-6.

25. Fayer S, Lacey A, Watson A. BLS spotlight on statistics: stem occupations - past, present, and future. Washington, D.C.: U.S. Department of Labor, Bureau of Labor Statistics; 2017.
26. Franklin D, Skifstad G, Rolock R, Mehrotra I, Ding V, Hansen A, Harlow D. Using upper-elementary student performance to understand conceptual sequencing in a blocks-based curriculum. In: Proceedings of the integrating technology into computer science education, SIGCSE. ACM; 2017. p. 231–6. March.
27. Funke A, Geldreich K. Gender differences in scratch programs of primary school children. In: Proceedings of the 12th workshop on primary and secondary computing education; 2017. p. 57–64. November.
28. Goldweber M, Little JC, Cross G, Davoli R, Riedesel C, von Konsky B, Walker H. Enhancing the social issues components in our computing curriculum: computing for the social good, 2. ACM Inroads; 2011.
29. Goode J, Margolis J. Exploring computer science: a case study of school reform. ACM Trans Comput Educ (TOCE) 2011;11(2):12.
30. Google. Women who choose Computer Science: What really matters 2014.
31. Gonzalez T, Hernandez-Saca D, Artiles A. In search of voice: theory and methods in K-12 student voice research in the US, 1990-2010. Educ Rev 2017;69(4):451–73.
32. Hoffman B, Rosato J, Morelli R. Student engagement is key to broadening participation in CS. In: Proceedings of the 50th ACM technical symposium on computer science education (SIGCSE 19). Minneapolis, MN: ACM; 2019. p. 1123–9.
33. Hur JW, Andrzejewski CE, Marghitu D. Girls and computer science: experiences, perceptions, and career aspirations. Comput Sci Educ 2017;27(2):100–20.
34. Hsu HMJ. Gender differences in scratch game design. In: Proceedings of the international conference on information, business and education technology (ICIBET 2014). Atlantis Press; 2014. p. 100–3. February.
35. Indiana Department of Education. Sixth – eighth grade computer science standards 2016.
37. Khan NZ, Luxton-Reilly A. Is computing for social good the solution to closing the gender gap in computer science?. In: Proceedings of the Australasian computer science week multi conference. ACM; 2016. p. 17. February.
38. Margolis J, Estrella R, Goode J, Holme JJ, Nao K. Stuck in the shallow end: education, race, and computing. Cambridge, MA: MIT Press; 2017.
39. Master A, Cheryan S, Meltzoff AN. Computing whether she belongs: stereotypes undermine girls' interest and sense of belonging in computer science. J Educ Psychol 2016;108(3):424. https://doi.org/10.1037/edu0000061.
40. National Science Foundation. Women, minorities and persons with disabilities in science and engineering. National Science Foundation: Division of Science Resources Statistics; 2019.

41. Ozturk Z, Dooley CM, Welch M. Finding the hook: Computer science education in elementary contexts. J Res Technol Educ 2018;50(2):149–63.
42. Pollock L, Mouza C, Czik A, Little A, Coffey D, Buttram J. From professional development to the classroom: findings from CS K-12 teachers. In: Proceedings of the 2017 ACM SIGCSE technical symposium on computer science education. ACM; 2017. p. 477–82. March.
43. Ramalingam V, Wiedenbeck S. Development and validation of scores on a computer programming self-efficacy scale and group analyses of novice programmer self-efficacy. J Educ Comput Res 1998;19(4):367–81.
44. Ravitz J. Summarizing findings and looking ahead to a new generation of PBL research. Interdiscip J Prob Based Learn 2009;3(1):4–11.
45. Resnick M. Mitch Resnick: the next generation of scratch teaches more than coding. Ed Surge 2019. https://www.edsurge.com/news/2019-01-03-mitch-resnick-the-next-generation-of-scratch-teaches-more-than-coding. January 3.
46. SREB (Southern Regional Education Board) (2016). Bridging the computer science education gap: five actions states can take.Retrieved from https://www.sreb.org/sites/main/files/file-attachments/cs_commission_2016_0.pdf.
47. Saldaña J. The coding manual for qualitative researchers. Thousand Oaks, CA: Sage; 2015.
48. Tsan J, Boyer KE, Lynch CF. How early does the CS gender gap emerge? : a study of collaborative problem solving in 5th grade computer science. In: Proceedings of the 47th ACM technical symposium on computing science education. ACM; 2016. p. 388–93.
49. U.S. Bureau of Labor. Employment projections. Statistics 2018. http://www.bls.gov/emp/ep_table_102.htm.
50. VanLeuvan P. Young women's science/mathematics career goals from seventh grade to high school graduation. J Educ Res 2004;97(5):248–68.
51. Venkataraman R, Agarwal E, Brown D. Traditional high school STEM curriculum ineffective in promoting female interest in computer science. In: Proceedings of the E-Learning in Corporate, Government, Healthcare, and Higher Education. Association for the Advancement of Computing in Education (AACE); 2013. p. 2255–60. October.
52. Venkataraman R, Agarwal E, Brown DW. Engaging K-12 students essential for reducing gender gap in computer science education. Int J E Learn 2019;18(3): 331–43.
53. Walker A, Leary H. A problem based learning meta analysis: differences across problem types, implementation types, disciplines, and assessment levels. Interdiscip J Prob Based Learn 2009;3(1):12–43.
54. Wirkala C, Kuhn D. Problem-based learning in K-12 education: Is it effective and how does it achieve its effects? Am Educ Res J 2011;48:1157–86.

55. Cuny Jan. Transforming high school computing: A call to actionCuny, J. (2012). Transforming high school computing: A call to action.ACM Inroads, 3(2), 32-36. ACM Inroads 2012;3(2):32–6.
56. Fields D, Kafai Y, Nakajima T, Goode J, Margolis J. Putting making into high school computer science classrooms: Promoting equity in teaching and learning with electronic textiles in exploring computer science. Equity & Excellence in Education 2018;51(1):21–35.
57. Wolfe J, Powell E, Schlisserman S, Kirshon A. Teamwork in engineering undergraduate classes: What problems do students experience? Women in Engineering Division Technical Session - Development Opportunities for Diverse Engineering Students 2016;June 16. https://doi.org/10.18260/p.26069.
58. Hirshfield L, Koretsky M. Gender and participation in an engineering problem-based learning environment. Interdisciplinary Journal of Problem-Based Learning 2018;12(1):1–12. https://doi.org/10.7771/1541-5015.1651.
59. Creswell JW, Plano Clark V, Gutmann M, Hanson W. An Expanded Typology for Classifying Mixed Methods Research Into Designs. In: Tashakkori A, Teddlie C, editors. Handbook of mixed methods in social and behavioral research. Thousand Oaks, CA: Sage; 2003. p. 209–40.
60. Malik S, Al-Emran M. Social Factors Influence on Career Choices for Female Computer Science Students. International Journal of Emerging Technologies in Learning 2018;13(5).