Where's the Computer Science in Invention?
An Exploration of Computer Science in High School Invention Projects

Leigh Estabrooks[1], Helen Zhang[2], Anthony Perry[1], An-Me Chung[3], and Stephanie Couch[1]

[1]Lemelson-MIT Program, School of Engineering, Massachusetts Institute of Technology, Cambridge, MA
[2]Lynch School of Education, Boston College, Chestnut Hill, MA
[3]CSforALL, Chicago, Illinois

December, 2019

Table of Contents

Rationale, Motivation, and Importance for the Research Study

A growing number of individuals and organizations are engaged in deliberate efforts to teach people how to approach problem finding and problem solving in ways that utilize the processes and practices employed by accomplished inventors. The term "Invention Education" (IvE) has been embraced by many of these entities as a way of referring to their work and their type of educational offering(s)—offerings that IvE researchers have described as "an emerging and transdisciplinary field of study" (Invention Education Research Group, 2019). The claim that IvE is transdisciplinary implies that the concepts, processes, practices, and ways of thinking (or mindsets) that are important for working as an inventor are drawn from multiple fields and disciplines. The National Research Council's (2014) definition of transdisciplinary research references

> comprehensive frameworks … aligned with problem-oriented research that crosses the boundaries of both academic and public and private spheres … mutual learning, joint work, and knowledge integration are key to solving "real-world" problems. The construct goes beyond interdisciplinary combinations of existing approaches to foster new world views or domains. (p. 45)

The term "transdisciplinary" is often used interchangeably with "convergence" when describing the ways in which knowledge must be synthesized across different disciplines while working to invent. The National Research Council (2014) defines convergence as:

> Integration of knowledge, tools, and ways of thinking from life and health sciences, physical, mathematical, and computational sciences, engineering disciplines, and beyond to form a comprehensive synthetic framework for tackling scientific and societal challenges that exist at the interfaces of multiple fields. (p. 1)

Few studies are available to confirm invention educators' claims that student inventors learn by integrating knowledge and using tools and ways of thinking from multiple disciplines as they work to create technological solutions that are useful and novel. Little has been written that can inform our understandings of which disciplines and fields of study are common to the work of young inventors in elementary, middle, and secondary schools in the United States. This paper presents findings from an initial study in which we examined computer science (CS) education and the learning of CS principles and practices within the context of teachers' and students' participation in a year-long, transdisciplinary, invention education experience. We aimed, in this first pass, to develop a greater understanding of ways IvE aligns with and diverges from the field of CS education.

Given time and resource constraints, the study is limited to survey data from 15 teams of high school students (n=96) who participated in a year-long invention project in the 2018–2019 school year, and an in-depth examination of CS aspects being learned by students in three of those 15 team-based invention education projects. The 15 teams of students were drawn from a single program known as InvenTeams, which has been in existence for 15 years and is administered by the Lemelson-MIT (LMIT) Program. Survey data for all students, as well as the actions and lived experiences of the three teams of students and educators in the program, were compared to the core CS concepts and practices set forth in the 2016 K–12 Framework for Computer Science Education. The Framework was selected as the focal point for this telling case, given its strong support in the United States among educators, education organizations, policy leaders, and the private sector. The Framework is supported by many states, eight global technological corporations (e.g., Amazon, Apple, Google, and Microsoft), 49 educational organizations (e.g., Afterschool Alliance, College Board, ISTE, PLTW, and University of

Washington Computer Science and Engineering), and 35 individuals (Statements of Support, n.d.).

The two-pronged goal of understanding the transdisciplinary nature of IvE and understanding ways the work recognized as IvE and as CS converged has informed the research foci of this study, which aimed to:

- Explore CS concepts and practices identified in the K–12 Computer Science Framework that were observed in the work of young inventors engaged in the InvenTeams initiative;
- Investigate how and under what circumstances students engaged in and leveraged computing practices as they worked to invent; and
- Investigate how students perceived the impact of the CS knowledge and capabilities learned through work on their invention project.

There are references to both CS and computational thinking (CT) throughout the study. Fraillon et al. (2019) defined CT as "an individual's ability to recognize aspects of real-world problems which are appropriate for computational formulation and to evaluate and develop algorithmic solutions to those problems so that the solutions could be operationalized with a computer" (p. 27). The 2018 Assessment Framework for the International Computer and Informational Literacy Study argues that CT consists of two strands—conceptualizing problems and operationalizing solutions—each of which has various processes (Fraillon et al., 2019, p. 28). Wing (2006, 2008) argued that everyone, not just CS students, should be educated in computational thinking, the approach of "solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science." The focus of this study extended to CT because we concur with this argument; additionally, many of the educational organizations that support the K–12 Framework for Computer Science

Education—including CSforALL, Code.org, Girls who Code, NCWIT, and NCP—embrace both CS and CT.

InvenTeams as Representative Study Sites for IvE and CS Education

The LMIT Program's national high school InvenTeams grant initiative was selected as the focus of this study, in part, because it has a 15-year history of working with educators to help young people learn to invent. More than 6,000 students have participated in the program's two youth initiatives, JV InvenTeams for middle schools and InvenTeams for high schools. The program is administered by the Massachusetts Institute of Technology (MIT) School of Engineering and funded by the Lemelson Foundation. Our approach to conducting the study, as program administrators and as researchers examining our own programs, was guided by principles of practice common to those who embrace ethnography as epistemology, or a way of knowing (Green, Skukauskaite & Baker, 2012). We have attempted, for example, to avoid ethnocentrism by bracketing our own knowledge, or points of view, to uncover the insider knowledge of the students and teachers that is visible in the actions and discourse among teachers, students, and others. We worked, in conducting our analysis, to set aside what we believed we knew about the program and the CS Framework to re-examine the accounts and lived experiences of InvenTeam teachers and student participants.

InvenTeams became a national grants initiative of the existing Lemelson-MIT Program in 2004. The initiative has offered around 15 grants of up to $10,000 each year since then; to date, 257 InvenTeams have been funded to conceptualize, design, and build technological solutions to real-world problems. High school science, technology, engineering, and mathematics (STEM) teachers in public and private schools may apply for a grant. Adults leading clubs or homeschooling with a 501(c)3 or non-profit status may also apply. Teams consist of fewer than

20 members, plus one or two teachers. Participating students have specific areas of responsibility on the teams, including Administrative, Communications, Financial, Sustainability, and Technical roles. Technical roles are responsible for the mechanical and electrical/electronics construction of the invention prototype. Occasionally, a Systems Integrator role is also assigned. The roles were intentionally designed by program developers to resemble roles that exist in company-based product development teams. Technical mentors, located in the local communities of student teams, inform students' work.

InvenTeams participants commit to a year-long inventing process that begins in the fall and culminates in a capstone event, known as EurekaFest, held at MIT in June. This multiday event requires that each team present their technological solution—a working prototype—to a problem of their choosing to their peers and publicly showcase their invention to the MIT community. Prior to EurekaFest, InvenTeams are required to hold mid-grant technical reviews, where they solicit and receive feedback from stakeholders, within their community in late winter. The LMIT Program staff support the teams throughout the grant year with periodic check-ins, resource recommendations, and site visits. Since the initiative's inception, eight InvenTeam projects have been granted U.S. patents. The event map of the eight phases of the InvenTeam grant cycle are included in Table 1 (Estabrooks & Couch, 2018).

Table 1

Event Map with Phases of an InvenTeam Cycle

Phase	Phase description	Months	Duration in weeks	Activities	Milestone
		Teacher-focused phases			
1	Recruitment for and submissions	Oct.–Apr.	24	Application requires invention proposal,	April deadline

	of initial applications			information on school, resumes of teachers, letter of support from administrator, and statement of interest for invention projects.	
2	Finalists' selection by LMIT staff	Mid-Apr.	2	Evaluation of teachers' applications, utilizing rubric.	35 finalists receive Excite Award to attend EurekaFest
3	Professional development	Mid-June	1	Excite Award recipients attend professional development during EurekaFest at MIT, view current year's InvenTeam projects, receive feedback on proposed invention projects and guidelines for submitting the final application.	Invitation to submit final application
4	Summer work with students	July–Aug.	8	Excite Award recipients work with students to form teams and complete final application that fully defines the invention and the process that teams will use to reduce the invention to practice.	Final applications submitted
5	Judging	Sept.	4	Regional jury review and rank applications, make recommendations to LMIT for staff to make final selection.	15 teams selected and notified
			Team-focused phases with teachers, mentors, and students		
6	Invention project launch	Oct.–Feb.	20	Grant agreements signed, procurement cards released, communications and financial training for	Mid-grant technical review

				teams, and on-site visits from LMIT while teams iteratively build, test, and refine invention prototypes based on results and feedback; beginning of year survey.	
7	Post-technical review	Mar.–June	12	Final invention modifications and prototype building, raise travel funds to attend EurekaFest at MIT, plan travel, complete end of year survey.	Working prototype shipped to MIT
8	Capstone event	Mid-June	1	Teams travel to MIT, showcase inventions, present to their peers, meet collegiate inventors, and attend seminars in preparation for college and participation in the innovation economy.	EurekaFest

The longevity of the InvenTeams program and historical records preserved by program administrators provided researchers with an opportunity to examine changes in the selection and use of technology in invention projects across a 12-year period (2008–2019). Each team, as noted above, received a grant from the program to support their costs of building a working prototype. Teams had control over their designs and used funds to purchase materials and components they needed to build their project, within the allotted amount. An analysis of the purchasing data for InvenTeams, shown in Figure 1, revealed changes in the purchase of technical tools that paralleled increases in availability of well-documented computing development platforms. The data provided evidence of shifts in the hardware and electronics purchased by students to build their inventions. Robotics kits, for example, were purchased in all years except 2012. Arduino boards were purchased as early as 2009, with an increase in the number purchased in 2013 and each year thereafter. Over half of the teams purchased Arduino-based technologies in 2017 and 2018. Raspberry Pi computers were purchased components for inventing beginning in 2013, soon after the Raspberry Pi technology became available in the United States.

Figure 1. Technologies purchased by InvenTeams for inventing between 2008 and 2019.

The year 2013 also saw the introduction of the phrase "Internet of Things" to the Oxford dictionaries (Ashton, 2015). Internet of Things (IoT)—connected objects that can send and receive data—have become more popular with the high school inventors participating in InvenTeams. Figure 2 shows that records for two InvenTeams projects in 2008 and 2009 included elements that would be considered IoT. The two projects were undertaken prior to the widespread availability of affordable technologies and user interfaces. Figure 2 also shows that five 2019 InvenTeam projects, or one-third of the total projects in 2019, were IoT projects.

Mobile application development was first utilized by teams in 2016 and has continued each year thereafter.

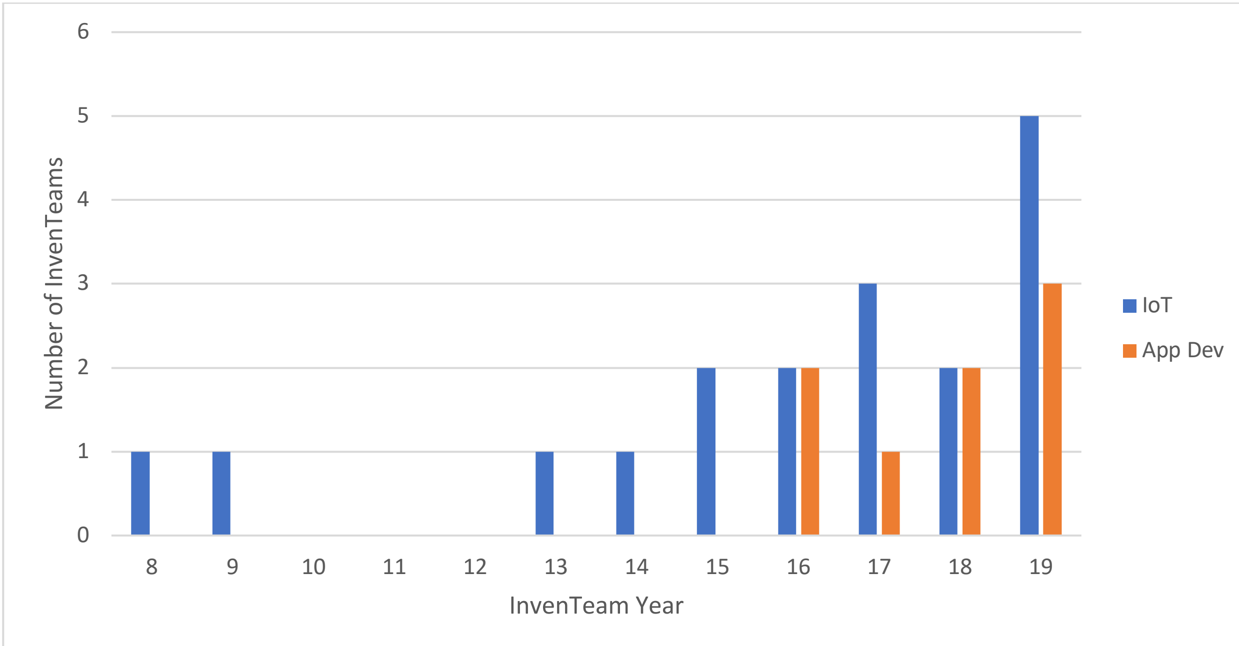

Figure 2. Internet-connected invention prototypes with IoT or mobile apps created by InvenTeams (2008–2019).

The InvenTeams' purchasing data and records surrounding the use of particular technologies offer evidence of the act of creating a technological invention—a requirement for receipt of a grant. The historical records of purchasing data, however, did not offer insights into why certain hardware and software were selected, learned, and applied to create useful and unique solutions to problems the students identified.

Research Methodology

Instruments and Data Collection

We employed a mixed-method design (quantitative and qualitative analysis) to investigate whether and how IvE and the concepts and practices in the CS Framework overlap with the work of InvenTeams. The data assembled for the study included:

- purchasing records from InvenTeams over a 12-year period (2007–2019);
- data from a voluntary end-of-year online survey made available to students participating in all fifteen 2018–2019 InvenTeams that included 35 questions (n=96); and
- artifacts from three of the fifteen 2018–19 InvenTeams, including:
  - a description of the teams' invention prototypes and annotated code,
  - invention statements written by each team,
  - blog posts made by team members during the year-long invention project,
  - presentation materials from talks given by each team at the end-of-year capstone event known as EurekaFest,
  - an end-of-year report from each team created pursuant to the grant agreement,
  - de-identified high school transcripts to show team members' CS coursework,
  - invention notebooks or journals,
  - transcripts of a one-hour interview of the teachers/facilitators of the three InvenTeams,
  - transcripts of a one-hour interview with up to four students in each team.

The survey made available to 2018–2019 InvenTeams students on all 15 teams was designed to gather each student's perceptions of their work on the invention project. Survey questions investigated students'

- demographics;
- experience of computing and computer science (e.g., CS courses taken in school and out-of-school settings, programming languages they are familiar with, CS resources they used during invention, and perceived gains in CS skills during their InvenTeam year);

- the year-long experience as an InvenTeam member;
- personal next steps (high school, college, career); and
- satisfaction with the InvenTeams program.

Table 2 shows the categories and the 11 survey questions added to the pre-existing InvenTeams end-of-year survey to address this study's objectives. The questions were added as a second section to the survey used by program administrators in prior years, after students had been asked about their demographics and before the questions regarding the year-long InvenTeam project work. These 11 questions were developed to ascertain the students' preparation through in-school and out-of-school programs for inventing and experiences with creating computational artifacts. A final open-ended question about students' perception of computer science in relation to their own lives was asked to give students the opportunity to discuss their views about computer science in words of their own choosing.

Table 2

CS/CT Questions Added to End-of-Year InvenTeams Survey

Categories of questions	# of questions	Sample questions
Extracurricular & enrichment history	3	• What activities did you participate in during elementary, middle, or high school? Select all that apply by grade span. (Matrix of activities by grade span) • What one activity was most impactful for you personally? Why? (Open-ended) • What one activity was most impactful for your work on the InvenTeam? Why? (Open-ended)
School & learning history	4	• What level of classes have you taken in high school? Select all that apply. (Radio button selection) • What math classes have you taken and where? Check all that apply. (Matrix of math classes by high school, college/university, community technology center, informally/online)

		• What computer science and technology classes have you taken and where? Check all that apply. (Matrix of computer science classes by high school, college/university, community technology center, informally/online) • What engineering or robotics classes have you taken and where? (Matrix of computer science classes by high school, college/university, community technology center, informally/online)
CS skill levels	2	• What was your level of skill for using the following before this school year? (Matrix of computer programs by novice, developing, and expert) • What was your level of skill for using the following after this school year? (Matrix of computer programs by novice, developing, and expert)
CS resources	1	• What resources have you used to gain information on programming this year? Check all that apply. (Radio button selection)
CS in your future	1	• How do you envision computer science impacting your future? This may include classes you may choose to take or, more broadly, your future work. (Open-ended)

The three 2018–2019 InvenTeams selected as telling cases (Mitchell, 1983, 1984) for more in-depth analysis were purposefully identified and invited to participate in the study (Creswell, 2012). The sites are at the extremes on a continuum (Rubin & Rubin, 2012), thus allowing for maximal variation sampling (Creswell, 2012) within InvenTeam sites. The extremes included places where the inventing was done (at school sites and in an out-of-school-time "club" setting), types of schools (public regular school and public regular school/charter), teacher/facilitator credentials (former engineers, informal educator, mathematics/science teachers, and technical education teacher), a low-income indicator (Title 1 school designation), and size of schools. The sampling method was employed in order to generate three representative

InvenTeam cases that could illustrate the CS/CT aspects of the broader set of work students engaged in as part of their work as inventors within different contexts.

The researchers recruited the three InvenTeams by first purposefully recruiting teachers via email in the spring of 2019, near the end of their InvenTeam year. Teachers from the three teams that agreed to participate were each asked to recruit up to four students from their team who had experiences with technology. The researchers suggested that the interviewee students have complementary CS experiences (Rubin & Rubin, 2012). Semi-structured research interviews were designed to yield insights into what team members experienced during their use of CS as part of their daily activities. Interviews were scheduled for the teachers and students during EurekaFest, a capstone event for all 15 teams, at MIT in June 2019. Four teachers—at least one from each of the three teams—were interviewed in a single focus group; one team had co-teachers and asked if both teachers could participate in the study. Students from each of the three teams were interviewed in three separate focus groups (one per team). In total, 12 students and four teachers from three InvenTeams participated in the study.

The interviews were video recorded and the recordings were transcribed. The semi-structured student interview was designed to explore students' invention experiences with computer science. It included seven questions that examined students' choices of the technologies, experiences of inventing with CS/CT (challenges and successes), sources of CS/CT learning, and their views on what constrained their invention. Sample interview questions included the following: Why did your team choose to use the specific hardware and software in your invention? Can you tell me about a problem that you encountered with the hardware or software during the inventing process? How was the problem solved? What did you/your team learn? Can you explain the programming the team did for your invention prototype?

The teacher interview aimed to gain insights into how teachers from non-CS disciplines supported students to invent, as well as their views on the connections between CS/CT and invention. Questions such as “How did you support your students during the invention experience?” allowed for emic understandings of the teachers’ role as facilitators of student teams. Questions such as “What do you think was the most difficult experience you encountered during this project with respect to helping students invent with technology?” were used to help triangulate information gathered from teachers about their roles and pedagogical practices. Understanding the experiences and beliefs of non-CS teachers is critical to the emerging field of invention education because many are not CS teachers or educators with strong backgrounds in CS/CT. The study of the teachers facilitating the work of students on InvenTeams offered an opportunity to see how educators without CS/CT backgrounds supported the development of technological inventions when they did not identify as CS teachers. Our ability to determine the facilitation strategies employed by the teachers could make visible the ways other educators—those interested in promoting invention education in their schools or organizations—could help students develop CS/CT skills, even when they do not possess such knowledge or skills themselves.

Data Analysis

Quantitative Data Analysis

End-of-year survey data for all students participating on InvenTeams (n=96) was first prepared and organized for analysis by exporting the data from Survey Monkey to an Excel file, and then into Stata 15 for analysis. Descriptive statistics were used to compare CS-related practices (i.e., computer programming languages they are familiar with, CS course-taking, and resources used to learn CS skills) amongst binary categorical variables (i.e., gender, school with high percentage of students qualifying for free or reduced-price meals, underrepresented minority in STEM, and the study population). Pearson's χ^2 likelihood test was used to investigate categorical variables (resources for learning CS, activities, and having taken at least one CS course) and the two-sided independent samples t-test was used to compare the mean intensity of course-taking patterns. The Wilcoxon rank-sum nonparametric test was used to compare the perceived change in skill level over the course of the grant period for a variety of CS-related technologies. Students rated their skill levels for different types of technologies based on three ordinal choices (expert, developing, and novice).

Qualitative Data Analysis

The qualitative data analysis was limited to the three teams. Participants in each of the three teams chosen for in-depth analysis were assigned a color pseudonym to help maintain anonymity. Researchers from LMIT utilized holistic coding as an exploratory method since they already had a general idea of what to investigate, based on an initial reading of transcripts from semi-structured interviews with the three groups of students (Saldana, 2013).

Analyzing Student Interviews (three teams). To explore evidence of students' engagement with CS as part of their work on their invention project, the researchers examined posters, reports, presentations, and transcripts of one-hour interviews with the students from each of the three InvenTeams. First, the researchers created a list of codes that corresponded to the five core concepts and seven core practices of the K–12 Computer Science Framework (see Table 3). They then determined which concepts and practices were evident in the students' documents and discussion of their technological invention. Researchers individually coded the transcripts. This was followed by intercoder agreement checks between the researchers. Lastly, a researcher from CSforALL, an organization known for its involvement in developing and promoting adoption of the K–12 Computer Science Framework, reviewed and verified the data analysis and findings for accuracy of the CS concepts and practices.

The coding rubrics were based on the five CS/CT concepts and seven CS/CT practices outlined in the K–12 Computer Science Framework. The Framework describes a baseline level of literacy for students across the grades, instead of standards for specialized CS courses. This way of depicting literacy levels aligns with InvenTeams because InvenTeam students were not engaged in the systematic study of CS. They developed CS/CT literacy throughout the invention experience on an as-needed basis in response to the demands of their project. Further, the Framework was written for all students. It has been widely recognized and has heavily informed the adaptation and development of state-specific CS standards, including those in California (California Department of Education, 2019). As a national grants initiative, InvenTeams has been successful in engaging students with diverse backgrounds in inventing, including females, students of low socioeconomic status, and students from racial and ethnic groups. It is not feasible at this time to investigate how each invention project overlaps with the state-specific CS standards of each team, since standards do not exist in all states. A generic CS framework as a point of reference for all learners is, therefore, more appropriate to this study.

While working on the analysis, the researchers first clarified the descriptions of the concepts and practices in the Framework in ways that added specificity surrounding the types of evidence that aligned with two of five concepts and three of seven practices. Agreed-upon interpretations were added to the codes used by researchers to make them more suitable for analyzing the InvenTeam data. Interpretations of the five CS concepts were as follows:

- Computing sytems (C1) was deemed to include forward-thinking or future aspects of the invention—even if there was not a physical artifact—since these discussions were based on knowledge of the integration of hardware and software.
- Algorithms and programming (C4) was deemed to include artificial intelligence since AI supports complex execution of algorithmic thinking.
- Collaborating around computing (P2) was deemed to include technical mentors as collaborators.
- Testing and refining computational artifacts (P6) was deemed to include systematic testing to troubleshoot computational artifacts that ultimately lead to some conclusion; trial and error were not considered. Also, efficiency and practicality were considered enhancement attributes.
- Communicating about computing (P7) was deemed to include the teams' discussions about their own intellectual property protection.

Analyzing Student Invention Artifacts (three teams). Student interviews and other artifacts created by the three InvenTeams were analyzed by researchers. Artifacts included invention statements, reports, and presentations submitted by all teams to fulfill the requirements of the program. The coding rubrics for the CS/CT concepts and practices were used to examine these artifacts. The analysis of the artifacts contributed to the evidence base for students' understanding of CS concepts and practices, and allowed for triangulation with data from the interviews. Table 3 shows the coding rubrics we used, sample student quotes from the interview transcripts, and corresponding evidence found in the artifacts.

Table 3

Coding Rubrics of Student Interviews and Artifacts

Framework concepts & practices	Description (additions to the Framework in italics)	Direct quotes and location of artifact evidence (coded team color)
C1 computing systems	The hardware and software that make up a computing system that communicates and processes information in digital form. This includes explanations or descriptions that demonstrate an understanding of the hardware and software. *This does not necessarily need to be work that has been accomplished; this could be future plans.*	"We have created a device that measures different sleep factors, such as heart rate, core body temperature, brain waves, and eye movement. We have also created a mobile app that measures alertness, tries to measure alertness through reaction-time games and different measures like that." (Blue t eam) Evidence: Presentation, Poster, and Final Report
C2 networks and the Internet	How the computing devices communicate with each other, the networks connecting computing devices to share information, and resources.	"When they all communicate through the server and the Pi takes readings from the probes, it calculates and makes sure it goes to an average data structure to make sure." (Orange team) Evidence: Poster and Final Report
C3 data and analysis	How the data is generated, collected, stored, visualized, and processed to better understand the world and make more accurate predictions.	"When they all communicate through the server and the Pi takes readings from the probes, it calculates and makes sure it goes to an average data structure to make sure." (Orange team) Evidence: Poster and Final Report
C4 algorithms and programming	The programming and algorithms designed to accomplish a specific task.	"We developed the mobile app with Swift, so we started with just making the games ... the reaction time games. We do that by setting a random timer between 3 and 7 seconds to determine when the picture will change. We did animals, like rooster and a sheep, for waking up and going to sleep. A random

		timer is used for switching the images so that it's not the same every time, so people will not just get used to clicking it at the exact same time.” (Blue team) Evidence: Presentation, Poster, and Final Report
P1 fostering an inclusive computing culture	How the InvenTeams students include the unique perspectives of others and reflect on one’s own perspectives when designing and developing computational products; how they address the needs of diverse end-users during the design process to produce artifacts with broad accessibility and usability.	“I wasn’t one of the original team, they put me on because they needed a different perspective for computer programming.” (Orange team) Evidence: Presentation
P2 collaborating around computing	How the InvenTeams students: 1. Cultivate working relationships with individuals possessing diverse perspectives, skills, and personalities; 2. Create team norms, expectations, and equitable workloads to increase efficiency and effectiveness; 3. Solicit and incorporate feedback from, and provide constructive feedback to, team members and other stakeholders; and 4. Evaluate and select technological tools that can be used to collaborate on a project.	“When we had our mid-grant technical review, we talked to one of the university professors, and she recommended just trying to start off at like a local university hospital, and implementing our system there and then building it up from there, so I think really getting more connected with our local resources within a hospital or whatever that is, I think that would be probably where we're going to start to build it from there up.” (Red team) Evidence: Presentation and Final Report (Red team)
P3 recognizing and defining computational problems	How the InvenTeams students defined the problem, broke it down into parts, and evaluated each part to determine whether a computational solution is appropriate.	“We had to build a robotic arm to take them out of the water and put them into a sponge and then back into the water to keep our system automated and then because the peristaltic pumps are 12 volts and neither Arduino or Raspberry Pi can supply 12 volts, it can supply up to five, we have a very simple motor driver board called an L298N that

		can receive positive or negative data from the Raspberry Pi or Arduino or whatever control you're using. It has a 12-volt power supply and then it can power the 12-volt motor." (Red team) Evidence: Final Report
P4 developing and using abstractions	How the InvenTeams students 1. Extract common features from a set of interrelated processes or complex phenomena; 2. Evaluate existing technological functionalities and incorporate them into new designs; 3. Create modules and develop points of interaction that can apply to multiple situations and reduce complexity; and 4. Model phenomena and processes and simulate systems to understand and evaluate potential outcomes.	"On the AI side of things, what we're doing right now is modeling it within a computer itself, not on the device itself yet. Just to validate the AI part of it and the evolution part." (Blue team) Evidence: Final Report
P5 creating computational artifacts	How the InvenTeams students 1. Plan the development of a computational artifact using an iterative process; 2. Create a computational artifact for practical intent, personal expression, or to address a societal issue; and 3. Modify an existing artifact to improve or customize it.	"We did create a dummy for testing the data but it's not quite running yet with all the vitals. So, we're just generating the range. Just random numbers." (Red team) Evidence: Presentation and Final Report "We wrote a program that we could use to test the wristband and track and then we graph that. That helped us figure out that the power supply wasn't working as well." (Blue team) Evidence: Presentation and Final Report
P6 testing and refining	How the InvenTeams students	"Well, with Swift, I feel like there's a lot of unknown errors. There's a lot

computational artifacts	1. Systematically test computational artifacts by considering all scenarios and using test cases; 2. Identify and fix errors using a systematic process (troubleshoot); and 3. Evaluate and refine a computational artifact multiple times to enhance its performance, reliability, usability, *efficiency, practicality*, and accessibility.	of words may mean the same error. There's this one that's sigabrt and I still don't know even now. It can mean so many things that are wrong with the code and so every time I got it, I didn't know what the problem was so I just had to look up every single issue that could be wrong with the code and just trouble shoot and trial and error on what would work and what wouldn't. I guess just seeing what you already found ... I got the error a lot of times so I would just try the same things. And see what would-Yeah, it's just in my brain now. I just know, yeah." (Blue team) Evidence: Presentation and Final Report
P7 communicating about computing	How the InvenTeams students describe, justify, and document computational processes and solutions using appropriate terminology consistent with the intended audience and purpose; how they articulate ideas responsibly by observing intellectual property rights and giving appropriate attribution. Includes teams' desires for intellectual property protection.	"We have a provisional patent so far, and I think the goal for most of us is to get the non-provisional. We haven't decided as a team … we haven't had the official talk.." (Blue team) Evidence: Presentation

Analyzing the Teachers' Interviews (three teams). Interviews with the teachers of the three teams (n=4) were analyzed using individual in vivo coding initially, followed by a strategic recoding based on the reseachers' agreed-upon categories (Saldana, 2013). The InvenTeams teachers were not CS teachers or teachers with strong CS backgrounds. The interview questions, therefore, were designed to investigate how these teachers suppported students' technological inventions. The analysis focused on the facilitation strategies employed by the four teachers from the three InvenTeams. Two researchers first coded the transcripts using their individually created emergent categories. They then discussed and combined the emergent categories, and re-analyzed the transcripts using the new categories. They met once more to discuss the codes and refine the categories; afterward, a researcher from CSforAll reviewed and finalized the coding categories and the codes to ensure a high interrater reliability.

Collaboration Between LMIT and CSforALL

The research team conferred via telephone or video conference to finalize findings from the telling cases, determine what additional research was warranted, and agree on next steps. The team was comprised of four researchers from the Lemelson-MIT Program and one researcher from CSforAll. The researcher from CSforAll provided guidance on the use of the Framework, offering insights into ways of interpreting the Framework, guiding the discovery of appropriate concepts and practices for the grade span, and assisting with the review of coding schema.

Results

Selective Descriptive Statistics from Survey Results (all InvenTeams students)

A total of 96 students participating in InvenTeams in 2019 responded to the end-of-year experience survey, a 61% response rate, with nine of these students identified as being part of the interview population. The full population was 42% female and 23% underrepresented minorities in STEM. Seventy-one percent of the students came from high schools with at least 40% of students qualifying for free and reduced-price meals. All students participated on an InvenTeam, but not all students were necessarily directly involved with computational aspects of the invention process.

Students on InvenTeams responding to the survey have taken a range of types of high school classes. Students were allowed to select more than one category or type of class; their selections are listed in Table 4.

Table 4

Types of High School Classes Taken by InvenTeams Students

Type of high school class	% of student responses
Career and technical education	53%
General education	72%
Honors	85%
Advanced placement	75%
Dual credit	37%
Other	4%

Results of CS- and Engineering Course-Taking Patterns for All InvenTeams Students

Fifty-two students (54.17%) in the overall population (n=96) reported taking at least one CS course and 68 students (70.83% of total) reported taking at least one engineering course. Table 5 shows that male students reported taking significantly more CS courses (average of 1.309) than female students (average of 0.675). Male students also took more engineering courses (average of 2.254) than female students (average of 1.450).

Table 5

Average CS and ET Courses Taken by Male and Female Students

Gender	Computer science course	Engineering tech. course
Male	1.309	2.254
Female	0.675	1.45
p – value*	.05	.05

Note. *Two-sided independent samples t-test

Table 6 demonstrates that students at schools in which 40% or more of the students qualified for free and reduced-price meals (FRPM) reported significantly fewer CS courses (average of 0.8053) compared to students at schools with less than 40% free and reduced-price meal eligible students (average of 1.464). There was no significant difference in the average number of engineering courses taken by the two groups (average of 1.926 for high FRPM and 1.821 for low FRPM).

Table 6

Average CS and ET Courses Taken by Students According to Free or Reduced-Price Meals

Income	Computer science course	Engineering tech. course
High FRPM	0.853	1.926
Low FRPM	1.464	1.821
t	2.204	
p - value	.0300	>.05

Note. *Two-sided independent samples t-test

Table 7 indicates that students who identified as underrepresented minorities (URM) reported taking fewer CS courses (average of 0.773) and engineering courses (average of 1.318), compared to students who did not identify as an URM (average of 1.108 and 2.066, respectively), though the two populations were not significantly different when compared with the two-sided independent samples t-test.

Table 7

Average Numbers of CS and ET Courses Taken by Students from Underrepresented and Non-Underrepresented Minority Groups in STEM

Race/ethnicity	Computer science course	Engineering tech. course
URM	0.773	1.318
Non-URM	1.108	2.066
t	1.0970	
p - value	.02754	>.05

Note. *Two-sided independent samples t-test

The Pearson χ^2 likelihood test was then used to determine if the binary variables of gender, percentage of students qualifying for free or reduced-price meals at each school, underrepresented minority, and study population predicted the likelihood of having taken at least one course in CS or engineering. Columns A and B of Table 8 show that male students were significantly more likely than females to have taken at least one CS course (38 males versus 15 females) and were more likely than females to have taken at least one engineering course (43 males versus 25 females). Columns C and D in Table 8 show results for students from the three InvenTeams who were interviewed. Males were significantly more likely to have taken at least one CS course (5 males versus 3 females). No significant differences, however, were observed in engineering coursework (4 males versus 4 females). We noted that the percentages of males on the three InvenTeams reporting at least one CS course (100%) and/or at least one engineering course (80%) were higher percentages than the course-taking patterns of all males within the total InvenTeams population (67.9% and 76.8% respectively). Similarly, the percentages of females on the three InvenTeams reporting at least one CS course (75%) and/or at least one engineering course (100%) were higher percentages than the course-taking patterns of all females within the total InvenTeams population (37.5% and 62.5% respectively).

Table 8

Number of Male and Female Students Who Took at Least One CS or ET Course

	All students (n=96)		Students from the three InvenTeams (n=9)	
	Column A	Column B	Column C	Column D
	Computer science	Engineering technology	Computer science	Engineering technology
Male	38 (67.9%)	43 (76.8%)	5 (100%)	4 (80%)
Female	15 (37.5%)	25 (62.5%)	3 (60%)	4 (100%)
χ^2	10.8630			
p - value	.001	>.05	>.05	>.05

Results of CS Resources for Learning (all InvenTeams students)

Students were asked to identify which resources they used to learn CS skills and concepts during the grant year. The resources cited by students are listed in Column A of Table 9. YouTube was cited by 65.63% of all students and was the most common response for both males and females (Columns B, C, D, and E of Table 9). “Teacher” was the second most cited response at 56.25% of both males and females, with nearly equal percentages of students from both genders (Columns B, C, D, and E of Table 9). “Trial-and-error” (51.04%), and “Mentor” (38.54%) were the third- and fourth-highest ranked resources cited by all students (Columns B and C of Table 9). A lower percentage of males (Column D of Table 9) cited these as resources in comparison to females (Column E of Table 9). A significantly higher percentage of males ($p<.01$), conversely, cited GitHub as a resource. GitHub is a web-based development platform for hosting open-source projects and supporting collaborations among people working in the IT field. Programmers can easily share challenges they encounter, solutions, and the code they created on the platform. Students can search for problems similar to the ones they experienced, find technical solutions, and utilize and modify code that is already written. Books were also cited as a resource by less than 21% of all students (Column C of Table 9). Differences between the resources cited by males versus females are depicted in Figure 3.

Table 9

CS Resources for Learning Cited by all InvenTeams Students

Resource	Number cited (all students)	% Cited (all students)	% Cited (all males)	% Cited (all females)	χ^2	p
Column A	Column B	Column C	Column D	Column E		
YouTube	63	23.08%	24%	22%		
Trial-and-error	49	17.94%	17%	19%		
Teacher	54	19.78%	18%	23%		
Mentor	37	13.55%	12%	16%		
GitHub	23	8.42%	11%	4%	7.6038	.006
Other	27	9.89%	10%	9%		
Book	20	7.33%	8%	7%		

Note. χ^2 statistic and p - values are listed only for statistically significant results.

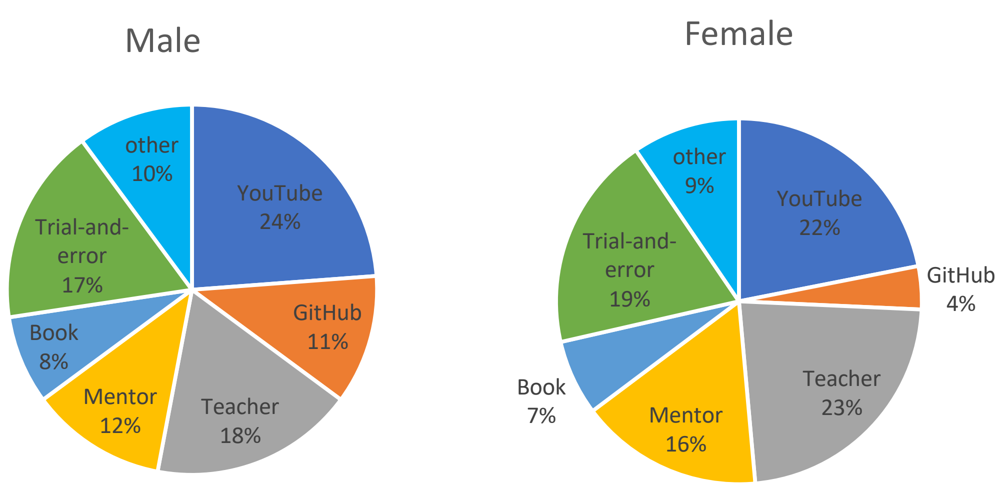

Figure 3. Percentages of resources male and female students used to learn computer science skills and concepts during the InvenTeams grant year.

Results of Perceived Changes in CS Skill Levels for all InvenTeams Participants

Students (n=96) were asked to self-rate their perceived skill level as “Novice,” “Developing,” or “Expert” on a variety of technical tools that included common programming languages, solid modeling, microprocessors, and scripting. The request, made at the end of the InvenTeam year, asked students to separately record their recollections of skill levels they possessed at the beginning of the school year and then those they believed they possessed after having participated on the InvenTeam. Many students did not complete the skill ratings (54.02% of the total rating opportunities were left blank). Response rates of other survey questions were higher, with an average of only 27.59% left blank. Students without any knowledge or experience with the particular tool listed on the survey may have skipped the question because they may not have even considered themselves as qualifying for the lowest category. Thus, missing responses were interpreted as a fourth category, representing a skill level of “less than novice.”

Table 10 presents percentages of students’ self-reported skills of each technical tool before and after the InvenTeams experience. A plus sign in the last column of each row indicates a positive growth in the number of students reporting that particular skill. The last column of Row 1 in Table 10, for example, shows that the percentage of students who believe that they possess HTML skills that are at the “expert” level increased by 4.17% between the start of the school year and the end of the year, after their InvenTeam experience. The percentage in students reporting the top two levels of “expert” or “developing” increased across the time period for eight technologies appearing on the survey. Technologies from the survey where no significant difference in self-reported skill level at the two time periods—using the Wilcoxon signed-rank test—are not included in Table 10. These included: CSS, Scratch, MatLab, R, Java, Blender, Minecraft, and Processing. This supports the notion that some students experienced growth in CS skills as a result of learning opportunities across the year, some of which may be attributable to their InvenTeam experience. Further analysis is needed to determine whether the students who left items blank, and are therefore presumed to be students with skills that are categorized as "less than novice," are the same students across all eight technologies. If so, this may suggest the existence of a "tech resistance" of some type on the part of the students who left the items blank. Non-responses on survey items ranged from a low of 34% on AutoCAD to a high of 44% on SolidWorks and Fusion 360.

Table 10

Students' Self-Reported Differences in CS Skill Levels After the InvenTeam Year

Row	CS skill and level	Before InvenTeams	After InvenTeams	% Increase or decrease
HTML is a markup language commonly used for web development.				
1	HTML expert	6.25%	10.42%	+ 4.17%
2	HTML developing	15.63%	19.79%	+ 4.16%
3	HTML novice	41.67%	35.42%	- 6.25%
4	HTML less than novice	36.46%	34.38%	- 2.08
Python is a programming language commonly used with Raspberry Pi platform.				
5	Python expert	3.13%	8.33%	+ 5.2%
6	Python developing	25.00%	27.08%	+ 2.08%
7	Python novice	34.38%	27.08%	- 7.3%
8	Python less than novice	37.50%	37.50%	No change
App Inventor is used to create Android applications.				
9	App Inventor expert	5.21%	9.38%	+ 4.17
10	App Inventor developing	13.54%	16.67%	+ 3.13%
11	App Inventor novice	40.63%	32.29%	- 8.34%
12	App Inventor less than novice	40.63%	41.67%	+ 1.04%
SolidWorks is a solid modeling computer-aided design (CAD) software.				

13	SolidWorks expert	2.08%	5.21%	+ 3.13%
14	SolidWorks developing	13.54%	14.58%	+ 1.04%
15	SolidWorks novice	40.63%	36.46%	- 4.17%
16	SolidWorks less than novice	43.75%	43.75%	No change

AutoCAD is a solid modeling computer-aided design (CAD) software.

17	AutoCAD expert	5.21%	14.58%	+ 9.37%
18	AutoCAD developing	21.88%	25.00%	+ 3.12%
19	AutoCAD novice	38.54%	27.08%	- 11.46%
20	AutoCAD less than novice	34.38%	33.33%	- 1.05%

Fusion 360 is a solid modeling computer-aided design (CAD) software.

21	Fusion 360 expert	2.08%	6.25%	+ 4.17%
22	Fusion 360 developing	8.33%	14.58%	+ 6.25%
23	Fusion 360 novice	45.83%	35.42%	- 10.41%
24	Fusion 360 less than novice	43.75%	43.75%	No change

Arduino is a single-board microcontroller.

25	Arduino expert	0.00%	9.38%	+ 9.38%
26	Arduino developing	15.63%	38.54%	+ 22.91%
27	Arduino novice	48.96%	27.08%	- 21.88%
28	Arduino less than novice	35.42%	25.00%	- 10.42%

Machine learning is reducing and making predictions with complex data sets.

29	Machine learning expert	2.08%	3.13%	+ 1.05%
30	Machine learning developing	12.50%	17.71%	+ 5.21%
31	Machine learning novice	40.63%	35.42%	- 10.42%
32	Machine learning less than novice	44.79%	43.75%	- 1.04%

The gains for each skill before and after the InvenTeam year shown in Table 10 are represented in Figures 4 through 11.

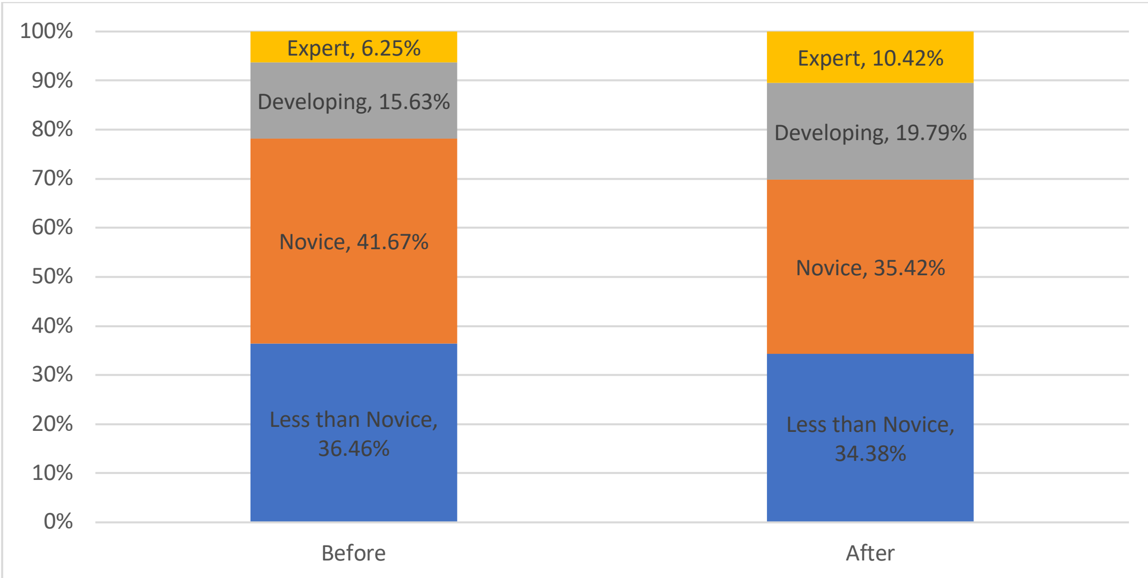

Figure 4. Self-reported HTML skills before and after InvenTeam experience.

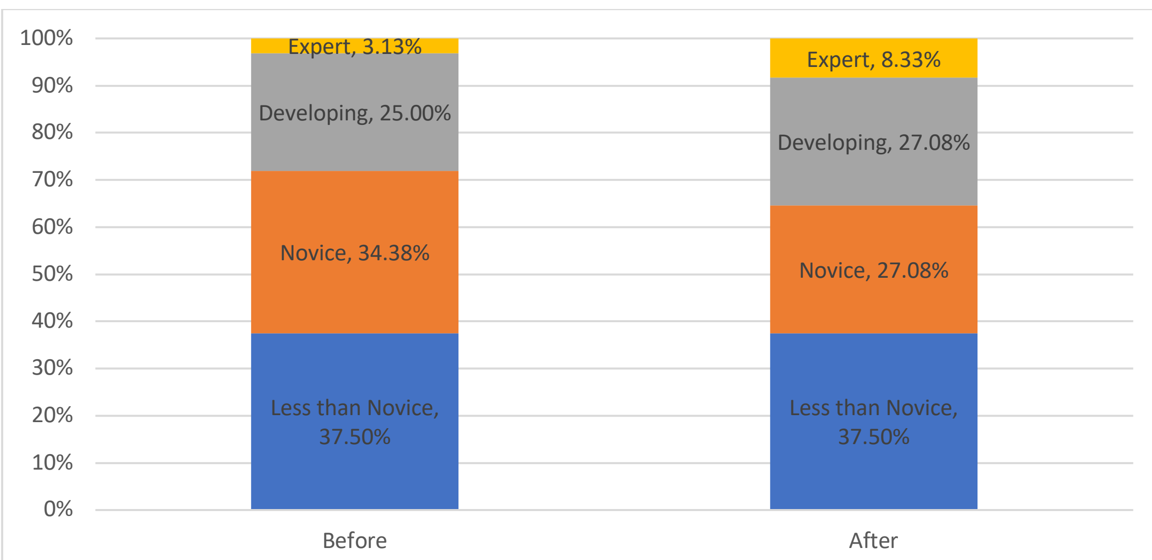

Figure 5. Self-reported Python skills before and after InvenTeam experience.

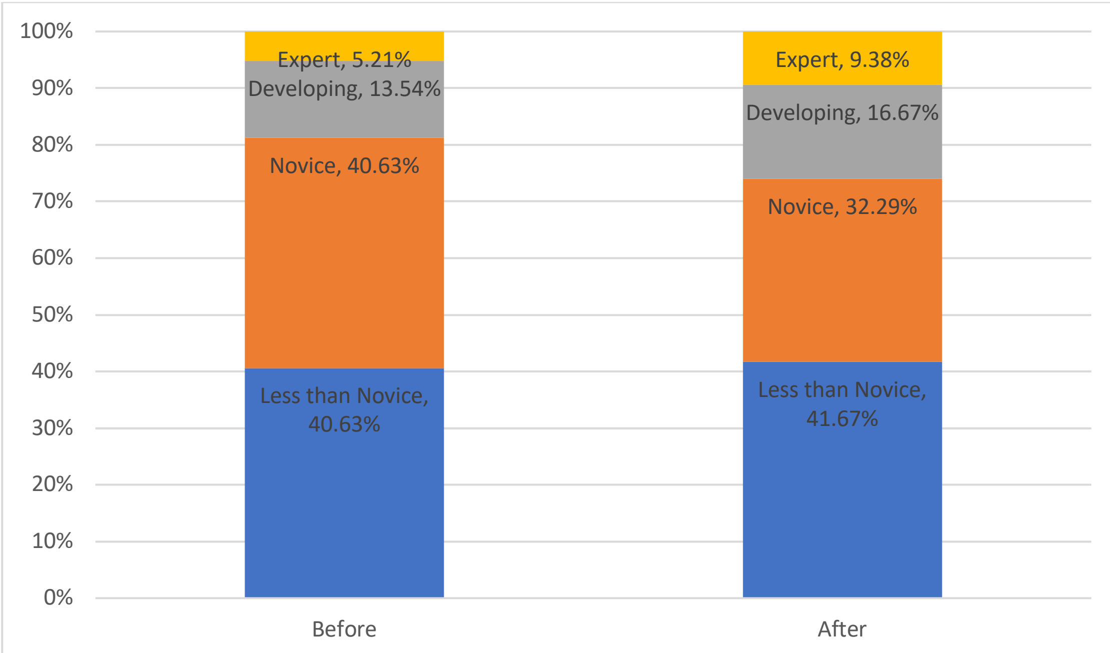

Figure 6. Self-reported app inventor skills before and after InvenTeam experience.

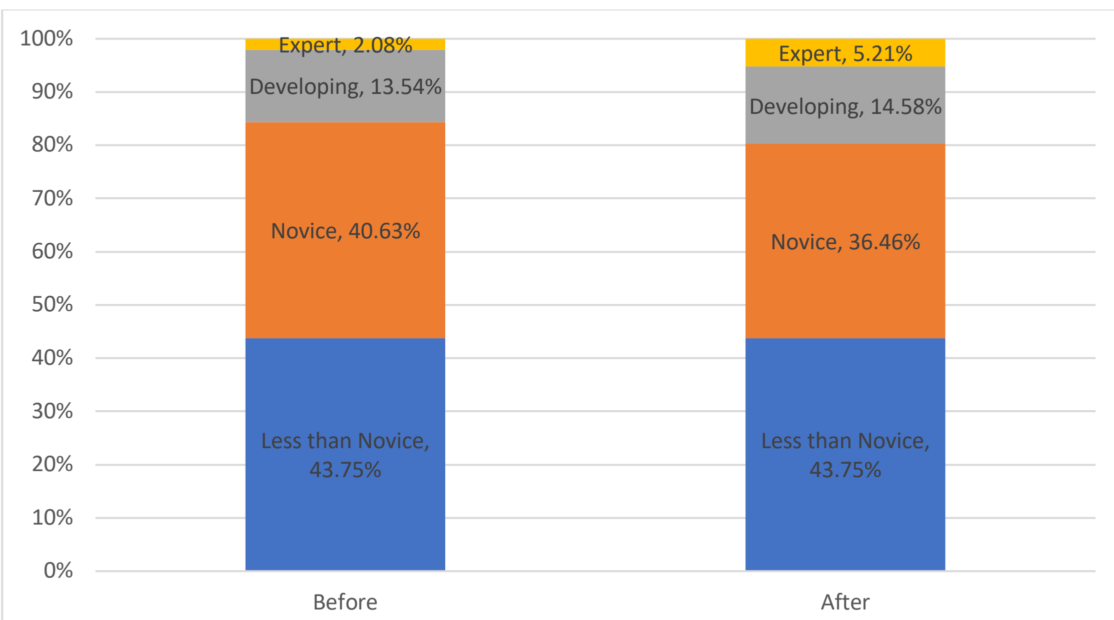

Figure 7. Self-reported SolidWorks skills before and after InvenTeam experience.

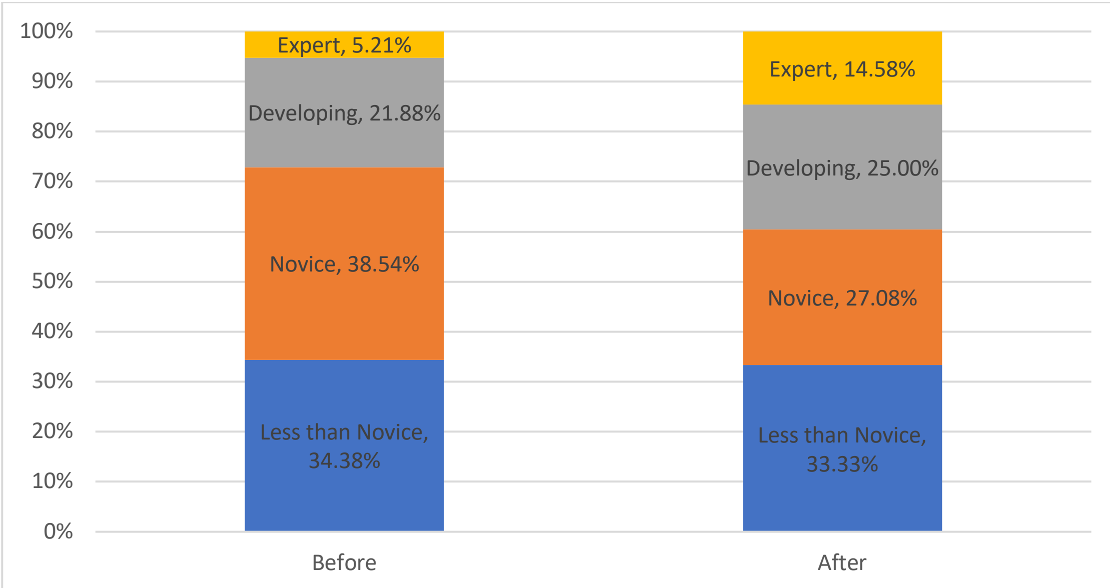

Figure 8. Self-reported AutoCAD skills before and after InvenTeam experience.

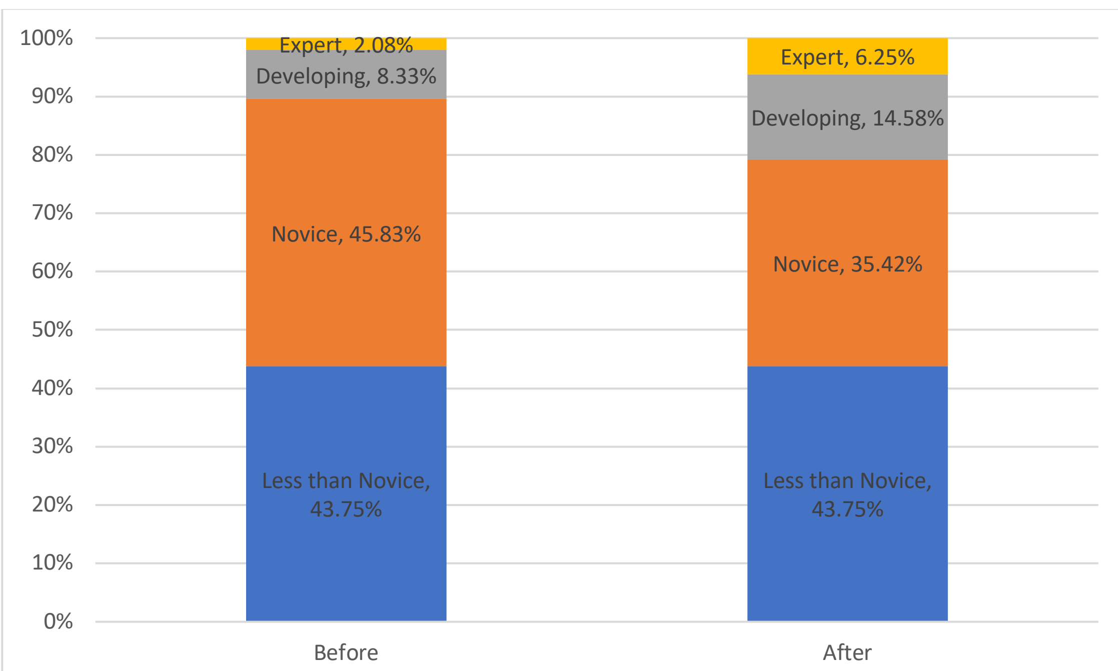

Figure 9. Self-reported Fusion360 skills before and after InvenTeam experience.

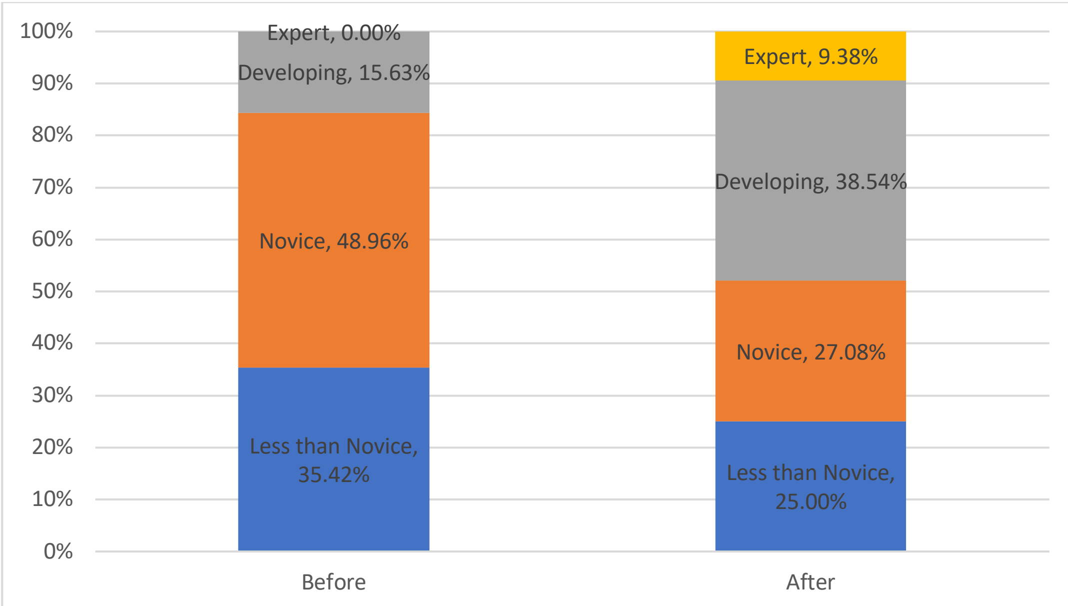

Figure 10. Self-reported Arduino skills before and after InvenTeam experience.

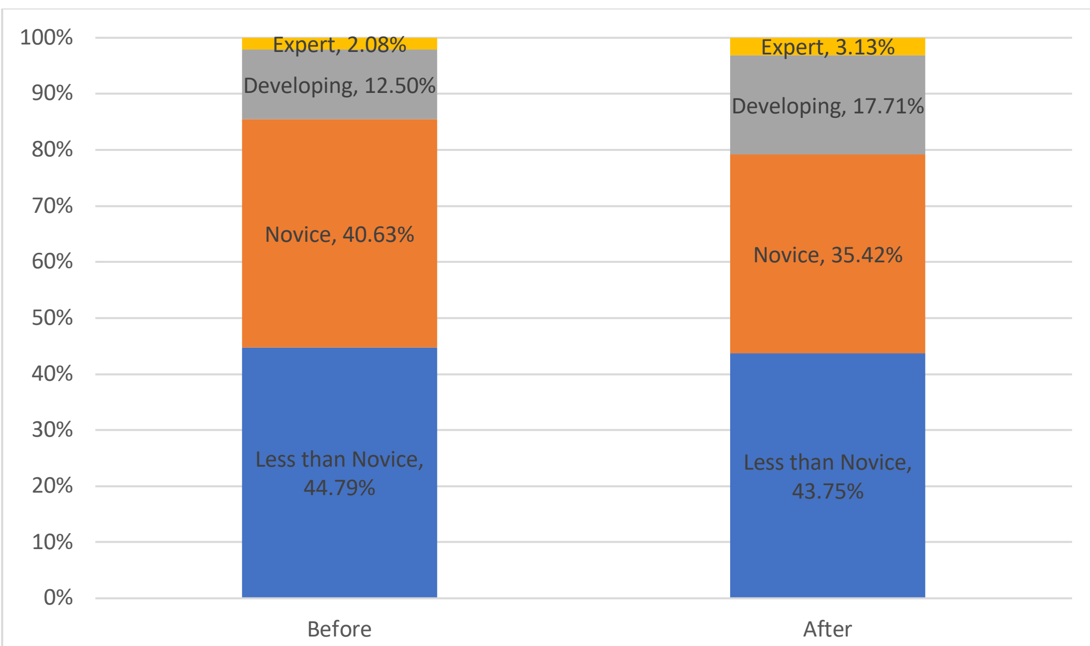

Figure 11. Self-reported machine learning skills before and after InvenTeam experience.

The differences between the beginning and end of the year should not be interpreted as being solely a result of students' participation in the InvenTeams experience. Some skills may have been learned through InvenTeams, but others may have developed through other opportunities for learning. AutoCAD, for example, is the solid modeling tool required to be used by students participating in Project Lead the Way courses, and App Inventor is often included in CS courses. We cannot rule out the explanation that these experiences outside of InvenTeams affected students' perceived skill levels.

Results of Activities Survey (all InvenTeams students)

Students (n=96) were asked, as part of the online survey, to indicate if they participated in a variety of activities in elementary, middle, or high school. Eighteen diverse activities were listed in a matrix ranging from sports, 4-H, and theater to activities that may have CS/CT components like hobby or maker club, science fair, and robotics. Students were able to select more than one activity at the three different grade spans. Table 11 includes the results (in descending order) of activity participation with 25 or more of the total respondents for each activity.

Table 11

Students' Self-Reported Activity Participation Data

Students' self-reported activities	Elementary school (n)	Middle school (n)	High school (n)	Participated in at least one grade band (n)
Public service or volunteering	19% (18)	50% (48)	62.5% (50)	69% (66)
Sports	48% (46)	60% (58)	57% (54)	69% (66)
Part-time job (afternoon/weekend)	0% (0)	4% (4)	55% (53)	55% (53)
Academic club	15% (14)	24% (23)	46% (44)	41% (49)
Orchestra or band	23% (22)	45% (43)	25% (24)	50% (48)

Robotics	6% (6)	22% (21)	41% (39)	50% (48)
Science fair	30% (29)	26% (25)	13% (12)	50% (48)
Religious or spiritual group	26% (25)	30% (29)	28% (27)	35% (34)
School government	9% (9)	17% (16)	21% (20)	30% (29)
Scouting	24% (23)	18% (17)	10% (10)	30% (29)
Hobby or maker club	5% (5)	13% (12)	25% (24)	28% (27)

Additional analyses of participation in robotics and hobby making were conducted using Pearson's χ^2 likelihood test to determine if any of the population variables (gender, income, race/ethnicity) were related to the likelihood of participation in those activities, given their potential for offering opportunities for learning CS/CT skills. Gender differences were apparent only in robotics. Male students, as shown in Figure 12, had a higher propensity to be on the robotics club/team in each grade span (elementary, middle, and high school years; $p = 0.080$). No sizable increased odds were observed at the elementary or middle school level, but they did appear at the high school level, in which 49.10% of males reported participation in robotics compared to 30% of females ($p = 0.062$). Witherspoon, Schunn, Higashi, and Baehr (2016) published research from a survey of participants in robotics competiton programs and noted a similar effect over time.

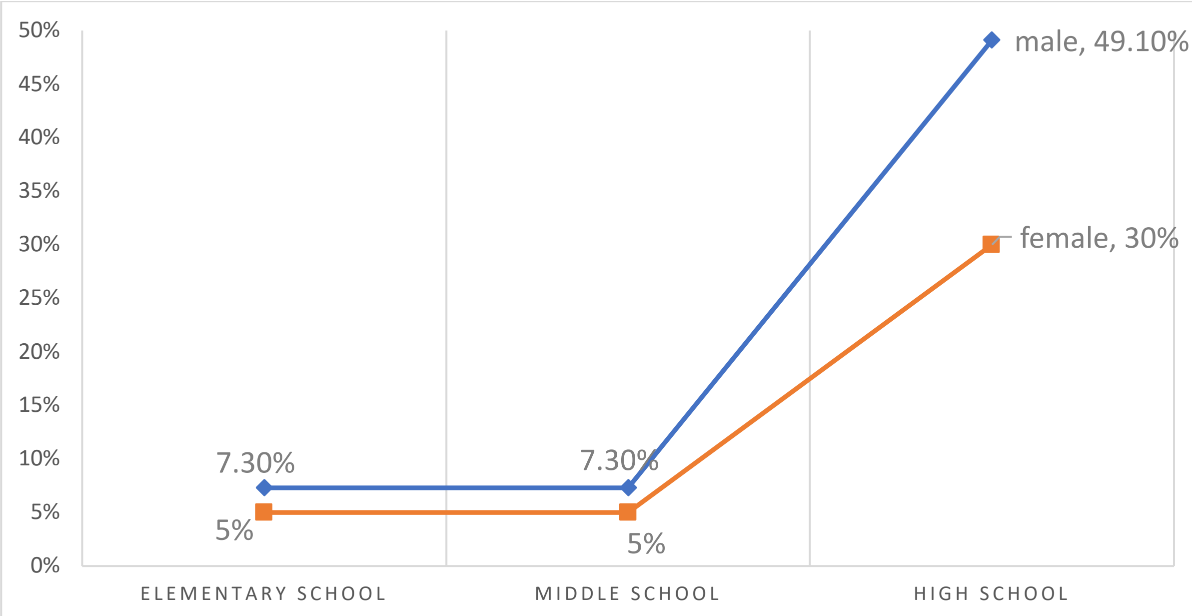

Figure 12. Percentages of male and female InvenTeams students participating in robotics from elementary to high school.

Results of Perceived CS Impact on Students' Future (all InvenTeams students)

Seventy-two percent of the InvenTeams respondents taking the online survey offered open-ended comments regarding how they conceptualized their future in relation to CS. The responses were tagged, coded, and then compiled into five categories: impact/high, impact/low, want to learn more, stated reason for CS in their future, and plan to take CS class(es) in the future. The distribution of the responses across each of the five categories is shown in Figure 13.

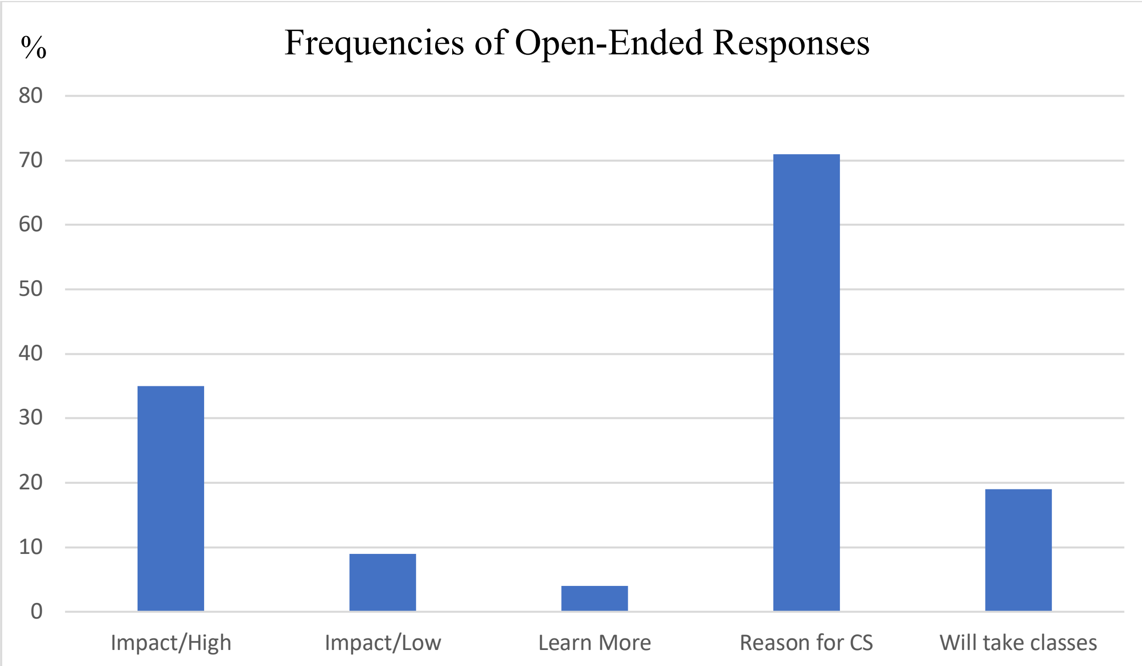

Figure 13. Students' conceptualizations of their future and CS.

Seventy percent of the responses pertained to "reasons for CS," including statements about ways students envision using CS in the future. A semantic analysis (Spradley, 1980) of students' statements (Table 12) shows that students associate the value of CS with their future occupation, such as plans to "go into software/electrical engineering in the future." One student saw links to their future occupation as well as educational goals, stating, "computers are continually becoming more prevalent in the world, so both at work and in school, having a strong foundation in computer science is essential." Another student alluded to ideas about their future self and "using computer science to extend my resourcefulness."

Table 12

Students' Conceptualizations of Their Future and CS

x is a future use of CS	That pertains to *y*
Going into software/electrical engineering in the future.	Occupation
Because the specific career I want involves a lot of stuff to do on computers.	Occupation

So both at work and in school, having a strong foundation in computer science is essential.	Occupation and schooling
Make useful things for automating work at home and school as well as in the workplace.	Occupation, schooling, and personal life
Extend my resourcefulness.	Personal life and personal capability or strength

InvenTeams Telling Cases (three teams)

We present in this section three InvenTeams as telling cases that illustrate how and in what ways invention and CS/CT are learned during students' invention projects. First, we describe the invention prototypes created by each team and the background information about each team (e.g., composition of the teams and CS course offerings in the local areas or regions). We then report results from the coding of the student interviews using the rubrics in Table 3, which depict the CS concepts and practices in the K–12 Computer Science Framework. The findings show the CS concepts and practices identified in the K–12 Computer Science Framework that these young inventors engaged with in the course of creating a working prototype of their invention. Lastly, we present findings from teacher interviews on their beliefs with regard to the relationship between invention and CS, and effective facilitation strategies and practices the InvenTeam teachers stated as being a part of their work with InvenTeam students.

Inventions of the Three InvenTeams

All three InvenTeams developed working prototypes of inventions that required their use of computing and engineering design. Students were able to reduce their ideas for an invention to practice by developing working prototypes and presenting at the culminating event for InvenTeams, known as EurekaFest. Specifically,

- The Orange team invented an automated nutrient injection system for hydroponic farmers. The system consists of three components: a probe, injector, and an internet application. Two ion probes (potassium and nitrate) measure ion concentration in parts per million within a nutrient solution, delivering to a server information about the ion concentrations. The data is then uploaded to an internet application. The application uses the data in conjunction with information inputted by the farmer about tank volume and crop type to suggest which liquid nutrients from a line of commercial nutrient products needs to be added to the hydroponic reservoir to produce optimum growing conditions. The injector pumps the nutrients into the solution when nutrients need to be added. After the nutrients are adequately mixed, the automated pump transfers the solution into the reservoir where the plants' roots are nourished.
- The Red team invented a device designed to streamline communications between emergency medical technicians and treatment centers. The device measures the emergency patient's blood pressure, temperature, blood oxygen level, and heart rate in real-time through a wrist-worn sensor package. The data is transmitted to an emergency medical technician's device via RFID and, when signal availability permits, to a web server. Hospital staff can monitor the patient's vital signs via a web application during transit to a treatment center.
- The Blue team invented a two-part adaptable sleep system to improve the sleep of teenagers. One part of the system measures five sleep indicators and applies a temperature stimulus to the wrist based on data. The system's mobile application prompts the user to play a reaction-time game every three hours during the day. At night, a headband and earpiece continuously measure the user's core body temperature (℃), brain

waves, heart rate, and eye movement to identify the stage of sleep the user is in. These measurements, along with the reaction-time data from the game played during the day, are used to determine an appropriate temperature stimulus applied through a wristband. An AI system adjusts the temperature stimulus in real-time by evaluating the input signals to best suit the user's preferences.

Information about the Three InvenTeams

As noted above, the three InvenTeams were selected to represent maximal variations within the InvenTeams sites. The teams are different from one another in many aspects, including the size of the school and grade levels represented, geographic locations, school types, concentration of students from low income families (i.e., Title 1), and the primary disciplines or subject matter regularly taught by the InvenTeams teacher(s). A profile of each participating team's site of study is summarized in Table 13.

Table 13

Profiles of Team Sites

Team	Geographic Region in U.S.	Locale (NCES designation)	Type (NCES designation)	Teacher Disciplines	Title 1 School	School size & grades
Orange	Southeast	Rural: Fringe	Regular school	AP calculus & chemistry	No	1,973 students in Grades 9–12
Red	Southwest	Suburb: Large	Regular school, charter	Career and technical educator (CAD)	Yes	468 students in Grades 6–12
Blue	West	NA	Out of school time, community club for STEM enrichment	Robotics coach, former electrical engineer	NA	100 students in Grades 4–12

The states in which these teams are located have different ways of addressing CS as a part of K–12 public education and as part of the college/university admissions policies. A review of the 2018 State of Computer Science Education report indicated that all the teams are located in states that allow CS to count toward a core high school graduation requirement. The Blue and Orange teams are located in states with K–12 computer science standards and K–12 computer science teacher certifications. Only the Blue team is located in a state that has approved a recommendation to allow CS to count toward a science eligibility requirement for admissions to the state university system. Additionally, researchers reviewed school and district CS policies. These similarities and differences in policies are summarized in Table 14.

Table 14

Computer Science Policies Applicable to Each Team

	Orange	Red	Blue
State has strategic plan for K–12 CS*	No	No	No
State has K–12 CS Standards*	Yes	No	Yes
State-level funding for K–12 CS professional learning*	No	No	No
State CS teacher certification*	Yes	No	Yes
State requirement for all high schools to offer CS *	Yes	No	No
State allows for CS to count toward a core high school graduation requirement*	Yes	Yes	Yes
State allows for CS to count toward a core, non-elective	Yes	No	Yes

admissions requirement in higher education*			
Additional school or district-level policy**	-1 CS course (including keyboarding) required for graduation -CS can substitute for a math or science graduation requirement	-STEM school—all students choose an engineering design or biomedical pathway -CS can substitute for a math or science graduation requirement	NA
Course offerings**	Course offerings in CS + Engineering Technology: PLTW Computer Programming (1,2); Flash; AP CS Principles	Course offerings in CS + Engineering Technology: PLTW CAD; CAD Certification Computer Programming; Data Science (1,2)	NA

Note. Sources: * advocacy.code.org, **publicly available school and district documents.

Exploratory Qualitative Results From Student Interviews

Using the rubrics for CS/CT concepts and practices shown in Table 3, we coded the instances where the InvenTeams students' explanations during the interviews demonstrated an understanding that aligned with the K–12 Computer Science Framework. We counted the instances and compared the frequency of instances of CS/CT concepts and practices expressed by the teams.

Five CS/CT Concepts in the Framework and the InvenTeams Experience. All three teams explained their inventions (e.g., hardware, software, and other physical parts), why they chose to incorporate computing in their invention, and how the prototypes utilized computing to achieve their performance. Their explanations of what they developed and how they developed their prototype through the year-long invention experience align with the descriptions of CS/CT concepts in the Framework. Figure 14 shows the frequency of explanations that demonstrated an understanding of CS/CT concepts by the students on the Blue, Orange, and Red InvenTeams, based on the coding applied by the researchers.

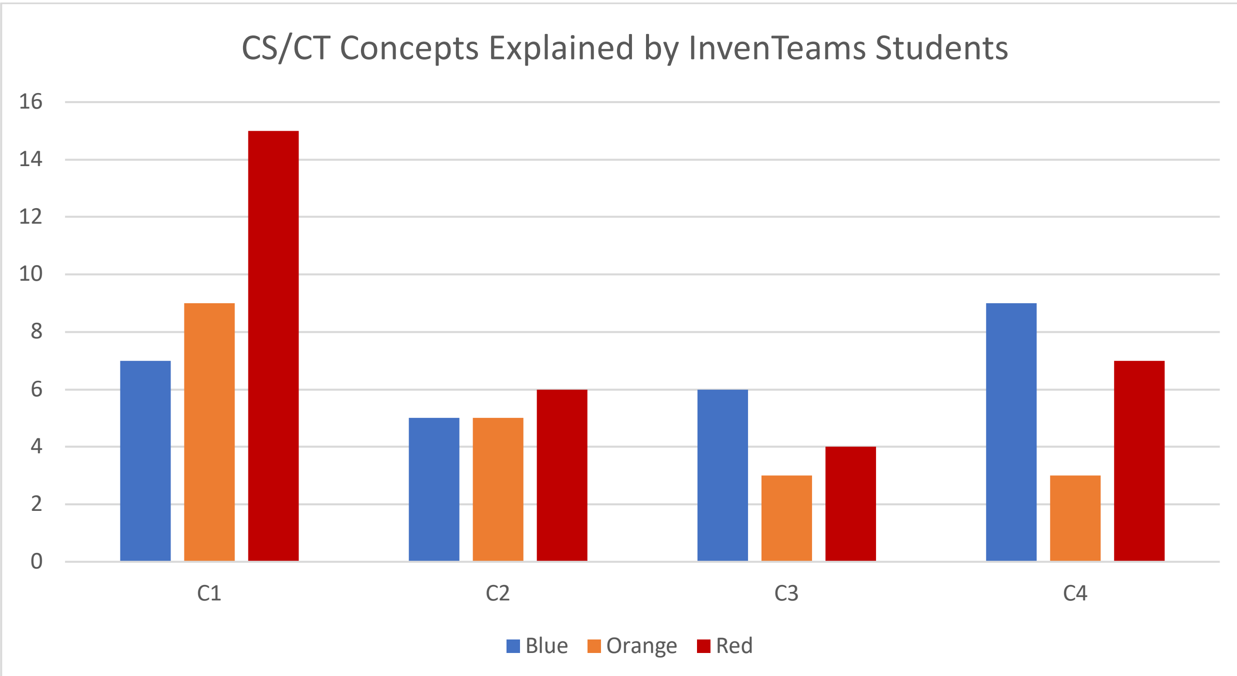

Figure 14. Frequency of InvenTeam students' explanations that demonstrate understanding of multiple CS/CT concepts. C1: Computing Systems, C2: Networks and the Internet, C3: Data and Analysis, C4: Algorithms and Programming, C5: Impacts of Computing (none of the teams explained the impacts of the computing in the interview).

The most frequently explained CS/CT concept is Computing Systems (C1), suggesting that this category contained the most evidence for CS integration by high school students into technological inventions. This included not only current, but also future work that integrates a wide variety of computing devices, such as physical components (hardware) and instructions (software). In total, 31 references to computing systems were found across the three teams. Students described the devices or components of their invention prototypes, what tasks each device performed, and how the computing devices were connected to other devices or components to achieve the goals. The invention created by each team consisted of more than one mechanical, electrical, and computational part, which may account for the high number of instances identified within this category. Students, as they invented working prototypes, needed to develop an understanding of how the components interacted with each other (e.g., how the software controls, processes, and provides information as input for hardware components).

Algorithms and programming (C4) was the concept with the second highest number of citations (Figure 14), with 19 mentions. All students explained the algorithmic thinking employed in their invention; for instance, the Blue team described the artificial intelligence algorithm they devised for their invention as follows:

> To kind of optimize the sleep profile which determines what that temperature stimulus is, we're using a genetic algorithm that's similar to NEAT, which is neuro evolution of augmenting topologies. We're basically creating a population, kind of natural selection almost, we create a population of these profiles that determine the stimulus for the user and we rate them all based on the sleep score which is based on what stages of each sleep you got and then also the day time data from the app. If the reaction time was really good all throughout the whole day, that would give a higher sleep score. We rank these profiles and then the top 50% of them we'll keep them and we'll breed new ones to replace the bottom 50%. That's basically raising the average and making better sleep profiles for that user.

These students' description of their work and their use of the NEAT algorithm is consistent with the Use-Modify-Create learning progression of CT that has been witnessed by a group of National Science Foundation-funded ITEST projects (Lee et al., 2011). Learners first interact with the computational artifacts or algorithms (the "Use" stage), then modify the artifacts to become their own (the “Modify” stage), and finally develop ideas for new computational projects of their own design that address issues of their choosing (the “Create” stage). Lee argues that, through iteratively refining and adopting the existing NEAT algorithm for their own invention purpose, students not only develop a deep understanding of the algorithm, but also deepen their CT experiences, acquired CT skills, and gained confidence in working on computational projects—in this case, an invention project.

Networks and the internet (C2) were mentioned 16 times and data and analysis (C3) were mentioned 13 times. The only concept in the K–12 CS Framework that was not evident was C5 (Impacts of Computing). While students described impacts of their invention on society, they did not parse the social implications of the digital world from the invention. They also did not discuss equity or access to technologies. One reason could be that invention and CS/CT have been so synergestically and stealthily integrated throughout the InvenTeams experience that students could not articulate what impacts are caused by the computing devices alone, versus the potential impact of the invention.

Seven CS/CT Practices in the Framework and the InvenTeams Experience. Students were asked during the interview to describe their invention journey, including how they worked with the computing devices, how the data were processed, what challenges they encountered while working with the computing system, and whether or how they solved the problem. The coding of their responses revealed various CS/CT practices they performed during the development of their invention (see Figure 15).

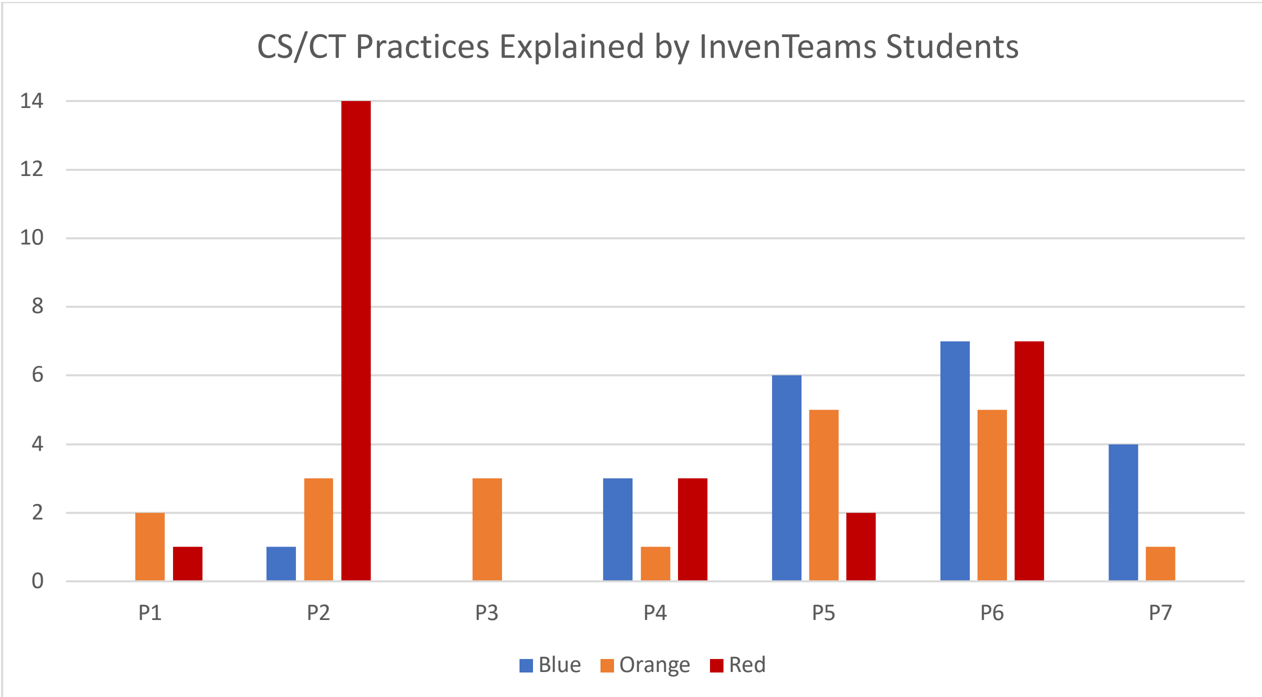

Figure 15. Frequency of explanations created by the InvenTeam students that demonstrate understanding of multiple CS/CT practices. P1: Fostering an Inclusive Computing Culture, P2: Collaborating around Computing, P3: Recognizing and Defining Computational Problems, P4: Developing and Using Abstractions, P5:Creating Computational Artifacts, P6: Testing and Refining Computational Artifacts, P7: Communicating about Computing.

The most frequent practices performed by students on the three InvenTeams were Creating Computational Artifacts (P5) and Testing and Refining Computational Artifacts (P6). A total of 32 instances were found in students' interview transcripts that explained how they created, tested, and iteratively revised the computational artifacts (and/or their inventions). This is not surprising, given that students spent the whole grant year conceptualizing, developing, and revising their inventions. Their invention prototypes consisted of multiple computational artifacts such as apps, data visualization platforms, IoTs, and microelectronics. This finding is also consistent with the Venn diagram that describes the intersection among practices in computer science, science and engineering, and math in the K–12 Computer Science Framework (2017). The diagram explicitly shows that computer science and science and engineering overlap on developing artifacts and communicating with data. Collaborating around computing (P2) was a practice performed, but cited during interviews by only the Red team.

We found, in examining computational artifacts, evidence that students engaged in this practice not only to develop invention prototypes, but also to test the invention prototypes. InvenTeams students do not have Institutional Research Board approval to perform tests on human subjects; therefore, obtaining adequate data for iterative invention development is a challenge when the invention is intended for use with humans. Two of the three teams invented computational artifacts to generate the test data. For instance, the Blue team created a mannequin head with electrodes installed inside to simulate brain waves and to send muscle movement signals and a pulse. A heater was placed inside the "ear" to mimic body temperature. The students explained,

> What we're doing right now is modeling it, just to validate the AI part of it and the evolutation part … Basically we started with writing a sleep model where it shifts through the different stages and produces those different sleep factors like heart rate and brain waves … The head does not provide the raw data, it's kind of a step of abstraction higher than the raw data, so we can start doing the test of our model and learning how that works.

Similarly, the Red team built an artifical human arm to simulate vital signs in order to test whether and how accurately their device can measure emergency patients' vital signs and communicate with the medical system in real-time.

Another CS/CT pratice that was frequently performed by InvenTeams students was Collaborating around Computing (P2). Each of the three interviewed InvenTeams included at least five members. Students needed to work with each other, teachers, and multiple technical and community mentors. Students described in the interviews how they collaborated. The Blue team formed a review board of technical advisors through their network. They also collaborated with each other, particularly on programming, using GitHub:

> We kind of had to learn how to work with each other offline, especially with code.… If they don't write any comments, nobody knows what any line does,… so we kind of learned to use GitHub to upload our code and have version control.

The Red team collaborated with professionals working in the medical industry (Hanger Clinic) and software development (Intel and Sandia National Lab):

> At the mid grant technical review, that's where our advice went … from all these people from Intel to Hanger Clinic … to the military. We had some interviews with the National Guard, emergency responders in the National Guard. After getting advice from all those different people, we finally got a solid idea of what our design should look like and what all its components should be.… as far as learning Python, that was thanks to a person who is a Python programmer at Sandia Lab. And she was up with me all evening helping with the program.

The Red team consisted of students with different expertise; they collaborated with local hydroponic farm owners:

> We tried to get people from a lot of different insights and perspectives and I think that's what really helped put the whole project together was, say we are having problems with the probes, you can pull in some of the people who are good with the math or the

chemistry behind those and help figure out what's going on. Or if we're having a structural issue we can get the people who are good with building things and help put that back together and how it's supposed to be working.

Analysis of Artifacts Collected From the Three InvenTeams

Three artifacts, consisting of posters, presentations, and final reports, were explored for evidence of the five CS core concepts and the seven CS core practices from the K–12 Computer Science Framework using the coding rubrics in Table 3. Two of the three artifacts (poster and presentation) were available for the public to view at the capstone event known as EurekaFest. One additional artifact was made available by the program administrator (final report). While there were many more artifacts that could have been explored, these three were common and available across all three of the teams participating in this study. Two researchers individually coded the artifacts, followed by inter-rater agreement. A researcher from CSforALL compared the codes for final agreement to establish inter-rater reliability, which helps to ensure the trustworthiness of the study (McAlister et al., 2017). Table 15 lists the number of instances of CS core concepts and practices evidence found in artifacts from the Blue, Red, and Orange InvenTeams.

Table 15

Evidence of Concepts and Practices Identified in Artifacts

Row	Evidence of CS concepts (C1–5) and practices (P1–7)	Presentation artifact	Poster artifact	Final report artifact	Total
1	C1 - Computing systems	5	3	9	17
2	C2 - Networks and the internet	1	4	9	14
3	C3 - Data and analysis	5	2	9	16

4	C4 - Algorithms and programming	1	1	4	6
5	C5 - Impacts of computing	0	0	0	0
6	P1 - Fostering an inclusive computing culture	3	0	0	3
7	P2 - Collaborating around computing	7	0	2	9
8	P3 - Recognizing and defining computational problems	1	0	2	3
9	P4 - Developing and using abstractions	0	0	3	3
10	P5 - Creating computational artifacts	4	0	7	11
11	P6 - Testing and refining computational artifacts	2	0	7	9
12	P7 - Communicating about computing	3	0	3	6
13	Total instances of evidence	32	10	55	97

The posters, available for public viewing at the capstone event, offered evidence of four of five CS concepts: computing systems, networks and the Internet, data and data analysis, and algorithms and programming (Table 15, Rows 1–4). The posters contained no evidence, however, associated with the seven CS practices (Table 15, Rows 6–12). This finding could be related to instructions provided to all InvenTeams, in which students were asked to focus on the technological invention for the poster. Archived versions of the students' presentations to attendees also offered evidence of four of five CS concepts: computing systems, networks and the Internet, data and data analysis, and algorithms and programming (Table 15, Rows 1–4). Additionally, they contained evidence related to six of seven practices (Table 15, Rows 6–12).

The non-public final reports submitted to the program as a condition of the grant agreement included evidence of both CS concepts and practices. The thoroughness of the final reports was variable and ranged from five to 27 pages. The most thorough final report, submitted by the Blue team, also had the most evidence of the CS concepts and practices. Like the student interviews, four of five CS concepts could be identified (Table 15, Rows 1–4) in the final reports, yet there were no artifacts that included the concept of the impacts of computing; rather, there was evidence of the impacts of the technological inventions. The reports also contained evidence of six of seven CS practices (Table 15, Rows 6–12).

Exploratory Results from Analysis of Teacher Interviews

Teacher interviews were conducted to gain insights into teachers' perspectives surrounding the relationship between invention and computation. All teachers described ways that invention and CS overlap. Teachers stated that "they [invention and computer science] walk hand in hand," and "you can't have one without the other." One teacher explained that he thought that CS is just one of the tools of invention,

> I think a lot of what I do is teaching them tools of invention. I look at computer science and microcontrollers and electronics that go with it as another tool. One of the simplest things we do is making a temperature sensor with Arduino and then we can go and measure temperature anywhere we want. We can stick it anywhere and measure temperature. We've got this new tool and that's just look at it as another tool just like a hammer.

Another teacher, citing advances in technology, stated "the way we look at creativity nowadays is very different than the way we looked at creativity years ago," and therefore, the technology and computation are indispensable to inspiring students' creative invention ideas.

Teacher interviews were also conducted to identify the strategies or approaches teachers employed during their work with InvenTeams. The following five facilitation strategy themes were identified through the coding of the interview data:

Facilitation Strategy #1: Establishing Mutual Trust Between Students and Educators. InvenTeams teachers and students trusted each other. They had known each other from educational experiences before InvenTeams. The teachers believed in students' capability to work on invention projects. For instance, the Blue team teacher had previously worked with some of the team members in afterschool robotics and invention programs, and noted, "Some of these kids were kids that I had done robotics with for some time so I knew that they had the capability to do these kind of things [sic] and they were very independent working on their own." The Orange and Red team teachers had taught the students science before working with them on an InvenTeam, and were very familiar with their students. The Orange team teacher, for example, stated that "Most of them [the students] had had me for a couple years and they've heard my stories about being in research and failing and going through that back in the '80s."

The InvenTeams teachers expressed their comfort with admitting to students that they were not experts in invention or computer science. When students struggled with programming, they worked together with the students to figure out solutions. They also supported students to find mentors and/or collaborators with the desired expertise. Their teaching is consistent with the literature on project-based learning, which calls for teachers to transition from the traditional role as a knowledge transmitter to a learning facilitator to successfully implement such learning experiences. Sawyer (2019) describes this type of non-instructional teaching supported by constructivist research (p. 4) as guided improvisation that includes an environment of trust and safety (p. 35–36).

Facilitation Strategy #2: Hands-Off, Providing Students with Flexibility. All the teachers described personal efforts to stay as "hands-off" as possible during the InvenTeams project to ensure that the experience was student-centered. The three teams were assembled by the students. A few students started with the invention idea, identified other students with the needed expertise, and then formed the team that included the teachers.

The teachers also accommodated the work plans and schedules of students. Given that InvenTeams is primarily an afterschool program, and with the busy schedules of high school students, the teachers provided workspace and were always available for students. Their accounts cited instances in which they stayed late and allowed the students to work in their classrooms in their spare time. Their accounts also included statements indicating they understood that each individual student may have their own work schedule. The teacher of the Blue team noted,

> When I started I thought, "This is the most amazing thing. The kids are going to be here every day and all day long," and it turned out that some of them weren't. It wasn't always the same ones. People would come and go because they're so busy. Most of my students were [high school] juniors. They're so busy with things and somebody might disappear for three weeks. I'm like, "did they quit?" No, and then they're back and they're working all night long the next month or something. It's just that they had so many things to do that they had to try to manage. I had to just be the guy that kept the door open for them.

Facilitation Strategy #3: Providing Resources as Needed. The InvenTeams teachers cited numerous instances in which they provided resources to support students' invention projects. The resources can be grouped into two categories:

Community Resources. One distinctive characteristic of InvenTeams is that the teams identify problems critical to the local community and develop technological solutions that are acceptable and accessible to the community members. Therefore, community engagement is of paramount importance to the InvenTeams experience. All teachers explained in the interview how they helped students to locate resources in the local community, connect with community members, and engage the community as users, testers, mentors, and collaborators of the invention projects. For instance, the teacher of the Red team helped students find mentors and collaborators from the Hanger Clinic, Intel, and Sandia National Laboratory through personal connections and the network of parents. These community members provided invaluable advice during the mid-grant technical review and helped the team determine the key components of the final design. Another example was the teachers of the Orange team inviting the owner of a local hydroponics farm to talk to the team members about the challenges of hydroponics farming—for example, that they need to send a water sample to a university lab once every two weeks, with data arriving too late for nutrients to be adjusted for agronomic needs of the crop, thereby resulting in lost opportunities and increased costs. Such conversations helped these students focus their invention projects, further motivated them, and supported them to establish a long-term collaboration with the community. The farm owner attended the mid-grant technical review, tested the invention prototypes, and offered numerous suggestions throughout the invention project.

Instructional Resources. Each teacher's own expertise and skills served as a resource for their team. Teachers brought their skills to the InvenTeams experience by delivering instructions and mentoring on the knowledge desired by students. The Orange team, for example, developed the smart nutrient additive system for hydroponics farmers, which automatically analyzes and measures the nutrient level in the water, calculates the difference between the current and optimal nutrient level for the plants (e.g., tomato), and creates and releases the mixture into the hydroponic system. The students had learned the chemistry related to plant growth (e.g., plants need three macro nutrients: nitrogen, phosphorus, and potassium). However, they did not know how to develop scientific experiments to determine the optimal nutrient level for specific plant crops. The teachers of this team expanded on their AP science coursework and use of lab notebooks for scientific research. One of the teachers explained that the students needed to:

> learn how to write their details and write all their thoughts down and things and the importance of doing that and the validation of doing that and the witnessing that goes through that process and why that happens and why that's important for patent type things, and teaching that part of the process, is something that we don't teach in a high school setting.

This led students to design a series of experiments to determine not only the best growing conditions, but also how changing one nutrient would impact plant growth. They analyzed the data collected and incorporated their findings in the iterative design of the smart nutrient system's prototype. Examples of student work are included in Figures 16 and 17.

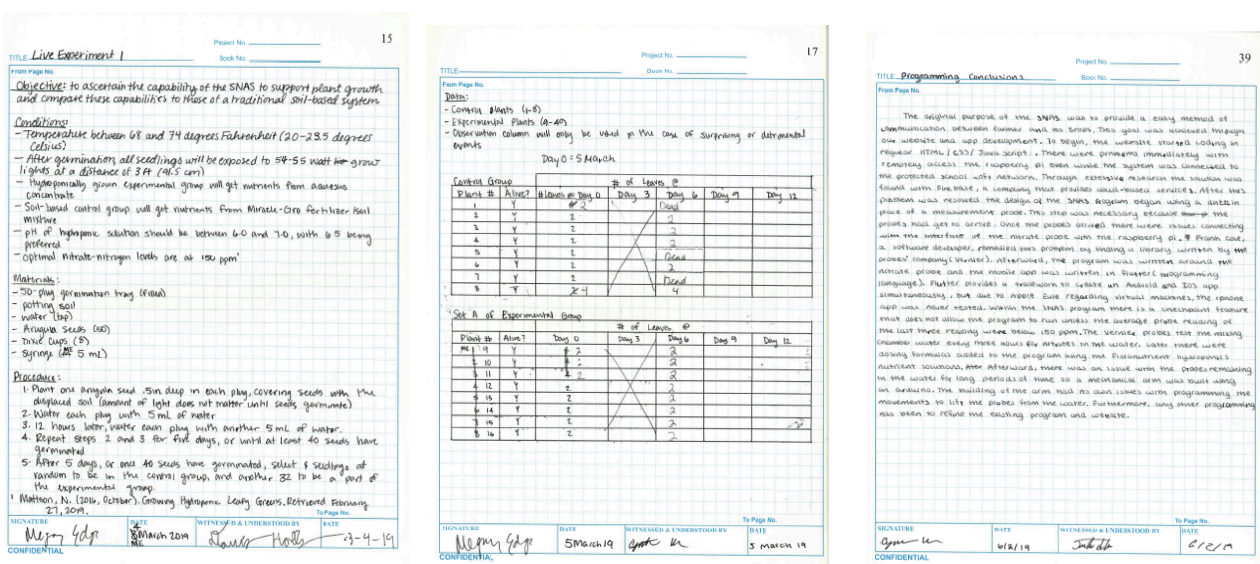

Figure 16. Examples of Orange team's experimental work and programming conclusions from their invention notebook.

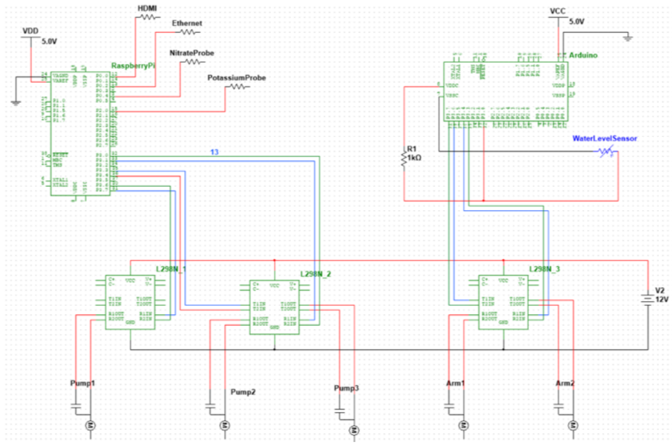

Figure 17. The Orange team's electrical schematic.

Facilitation Strategy #4: Managing Teamwork. Inventing requires teamwork. The InvenTeams that were included in this research each included five to 13 student team members. The interviews with teachers revealed that supporting the collaborations and solving conflicts among team members constituted a major part of their work. All teachers described strategies they employed to support student collaborations, including:

Establishing Communication Strategies and Tools. All InvenTeams had routine face-to-face meetings for brainstorming, troubleshooting, and check-ins of the invention work (e.g., the Red team met every day at lunchtime in the teacher's classroom, the Orange team met every day after school in one teacher's classroom). They also used instant messaging tools: the Orange and Red teams used group chats and the Blue team employed Slack and Trello with different channels for subgroup communications. These tools ensured faster and smoother communications among the students and with the teachers, as noted by the teacher of the Red team, who indicated "we needed something [the communication tool] really quick" to keep the invention work going.

Helping Students Understand and Accommodate Different Perspectives and Mindsets of Other Team Members. InvenTeams is a year-long process in which students engage in collaborative invention projects that do not have a fixed, linear goal. Teachers spoke to conflicts that were evident across the year. The Blue team teacher described his work to lead a group internal review when he recognized this problem in the team:

> [in the internal review] we did a thing where we list things that you'd like the group to keep doing, things you'd like the group to do less of, and things you'd like the group to do more of. We put that whole chart on the wall and did that. We also added to it, why are you here and what do you expect out of this? We wrote a lot of stuff on the wall and talked about it a lot and then after it we never talked about it again. It wasn't like it gave us an action plan … It was more just having that event and getting it all out on the table and talking about where we were going and why we were doing it, kind of solved a lot of the … we moved on from there.

The internal review engaged students in group reflection on the collaboration, offered an opportunity for students to better understand each other, and, according to the teacher's account, helped resolve some of the conflicts among students.

Encouraging Students to Teach Each Other. All teachers described explicit requests of students to teach each other. Some teachers described a leadership style in which they emphasized the need for students to ask for help and to rely on a colleague for help when they had a problem. The teacher of the Red team, for example, noted, "every one of them supported the effort of all other scholars." The teacher for the Red team said he was unafraid to say he did not know something. A telling example was when the teacher was approached by a student having issues with coding. He replied to the student, "Ask [Student J]. I don't know coding.

[Student J], you got this." In fact, the students' learning surpassed the teachers' knowledge. The Blue team teacher noted,

> They got to such a level where they would ask questions and it wasn't like you could ask the teacher and the teacher had the answer. I would have to go look it up or whatever. A lot of times it was easier to say, "well this guy's been working on it for two weeks, why don't you get together and try to sort it out yourselves."

By teaching others, the students reinforced and shaped their own knowledge and skills.

Facilitation Strategy #5: Coaching and Keeping Up the Morale of the Team During "Down" Times. All the InvenTeams teachers reported that coaching was crucial throughout the invention experience, particularly when students experienced great frustration and public critiques of their work. Teachers indicated that InvenTeam students experienced great frustration especially around the mid-grant technical review. The review invites community members, collaborators, and mentors for critiques and suggestions on the early prototyped solutions. One teacher indicated that their team wanted to postpone the review because they thought they did not have enough to show. The teacher for the Blue team explained,

> Leading up to the mid-grant review they were proposing to postpone it. They were like, "We're not ready. We need to do more." It wasn't so much I think that they weren't ready, it was that they were unsure of themselves. It was such a big project and there was no clear thing that once you complete this you do the mid-grant technical review. It wasn't clear when, there was no grade.

The Red team teacher indicated that their team came out of the mid-grant technical review with feedback that caused their project to "take a right turn." The teacher coached the team through the last five-week sprint and noted, "Because of everything that we did throughout the course of the year prior to that, they were ready for it. They were ready for it and they did a fantastic job."

All three InvenTeams teachers described experiences with student "burnout" after the review. The teachers indicated, though, that burnout and need for redirection were not perceived as failures. Our research shows that such times are referred to by InvenTeams students as "down" times. These down times actually stimulated students to reflect on their invention ideas, sparked creativity, and supported them to develop inventive mindsets (Estabrooks & Couch, 2018). Nonetheless, the teachers described a period of frustration. Encouraging students to persist in the face of frustration and continue with the invention work was cited by teachers as being of paramount importance. InvenTeams teachers described efforts to prepare students for the down times by sharing their own previous failed experiences (e.g., the Red team teacher shared her failed experience in research) and encouraging students to view failure as part of invention. The teacher of the Red team told students,

> It's not about our successes, it's about what we do with a failure. We're going to struggle… there is no growth without conflict and struggle. The biggest oak in the forest has to go through a lot of stuff to get to be the biggest oak... We're going to struggle with this. You're going to get outside your comfort zone and it's just using that guidance, the philosophies of learning and dealing with failure.

One of the Orange team teachers shared her prior experiences with the students, noting, "they've heard my stories about being in research [prior to teaching] and failing … they've heard me say that the only way you're ever successful is to truly fail many multiple times."

Experiences described by teachers indicated that throughout the InvenTeams experience, the teachers took actions to motivate students to persevere with the invention project, to help them adopt the feedback and critiques from the mid-grant technical review, and to support them to manage work under time pressure.

Discussion

This exploratory study investigated whether and how CS is integrated into invention education. We focused on high school students who participated in a year-long invention education program called InvenTeams. The InvenTeams students self-selected into this extracurricular activity to conceptualize, design, and build a technological solution to a real-world problem that they had identified. By examining student views on how their computer science knowledge impacts their invention projects, and what and how CS concepts and practices were learned and performed by students from three InvenTeams, this study presents preliminary evidence that CS converges with invention. It offers insights into ways convergence occurred in different contexts as students worked toward a common goal (an invention) while focusing on uncommon problems (i.e., problems unique to each team). Students' perceptions of growth in CS skills between the beginning of the school year and the end of their year-long InvenTeam experience, described in Tables 10 and 11, provided evidence for a just-in-time approach to learning to code. This approach, described by Gershenfeld (2008), is one in which students choose a technology and learn about that technology as part of their efforts to develop a solution to a problem. It can be contrasted, for example, with approaches in which students enroll in a course focused primarily on learning to code and then apply coding to particular problems during the course.

Student interviews after the InvenTeams experience show that CS is integral to their invention experience. Students naturally engaged with four of five CS/CT concepts. All seven CS/CT practices that aligned with the K–12 Computer Science Framework—such as creating, testing, and refining computational artifacts and collaborating around computing while inventing—were evidenced across the three teams, although the number of instances of specific practices varied from team to team. Computer science and invention were so synergistically interconnected in these students' InvenTeams experience that it was difficult for them to distinguish CS from invention. The analysis of the teacher interview confirmed this finding. All the teachers believed that invention and CS should be combined in technological invention projects and that CS is a new tool of invention, "just like a hammer."

Our findings indicated that across all 15 teams, females were significantly less likely (37.5%) than males (67.9%) to have taken at least one CS course in high school. Findings regarding the ability to teach CS through IvE provide educators with new possibilities for engaging more female students in CS education. The data from LMIT has shown that 40% of InvenTeams students are females who are generally well-educated in STEM, with future aspirations to attend college or university, and are not intimidated by a technological project. Yet, we learn from these students' accounts of their prior experiences that a gender disparity exists among those who have taken CS courses, with male students taking more CS courses than female students. This is consistent with findings from College Board: of the 135,992 students who took at least one AP computer science exam (AP CSP or AP Computer Science A) in 2018, 38,195 were female students, accounting for approximately 28% of the total population (College Board, 2018). Our data and previous research also indicate that, across their years of schooling, female students do not participate at the same rates as male counterparts in STEM enrichment opportunities that include CS/CT (such as robotics), and that this can exacerbate disparities in students' identities and interest in STEM college and career pathways (Couch, Estabrooks, & Skukauskaite, 2018).

The co-delivery of CS instruction with opportunities for students to learn to invent has the potential to address gender equity issues in STEM and CS. First, the framing of inventing to help people in the local community may help attract female students and spark their interests (Couch, Estabrooks & Skukauskaite, 2018). Young women can start with a non-coding role in InvenTeams and then take up the knowledge and practices as confidence grows through exposure. Second, because of the highly collaborative nature of the invention work, all students—including the "non-coders"—are afforded opportunities to acquire CS knowledge as they participate in the work and communicate with other team members. Statements made by students during the interviews indicated that even those students who are identified as "coders" or "programmers" needed more and different CS knowledge to invent; they sought out additional knowledge about CS through learning opportunities outside of the traditional learning systems (such as YouTube tutorials cited in Table 9). All student inventors on an InvenTeam interact with CS in informal, just-in-time ways. Further analysis is needed, however, to draw conclusions from the instances in which students did not rate their skill levels on particular technologies, and therefore are presumed to have skills that are "less than novice." We do not know how to interpret this data. Some students in non-technical roles, for example, may have intentionally avoided CS interactions as a whole or particular CS applications. Lastly, the IvE experience offers a concrete example of how students' lives are impacted through their interactions with CS as part of efforts to build a technological solution to a problem. Statements made by female students suggest that the experience developed their interest in CS, enabled them to recognize their capability for CS learning, and motivated them to continue with the learning. One female student on the Blue team expressed, "[after the InvenTeam,] I think I will just continue with this [learning] because I would enjoy it."

Recognizing the computational artifacts within inventing as evidence of CS learning is a novel concept to the still-nascent field of IvE. The limited recognition of CS as invention and invention as CS may be associated with the limited period of time in which K–12 projects have encompassed useful and unique engineered physical devices and devices powered by emerging technologies. Purchasing data shown in Figure 2 demonstrated that the InvenTeams initiative in the past four years witnessed a rapid growth of student projects that combine computing, networked communications, and physical invention. Educators who can envision ways of teaching students CS while also teaching students ways of working and thinking as inventors may be unwilling to teach IvE due to their perceptions that it is complicated and a stretch in comparison to their personal abilities. Invention education, and especially IvE that includes computing, may seem daunting and unattainable. However, the teachers and students in the three InvenTeams telling cases did not have extraordinarily strong CS backgrounds. The teachers were from science or engineering disciplines and the students had taken, at most, one CS course in school. Teachers adopted particular facilitation strategies to support their work with students. The cases serve as good examples of ways that non-CS teachers can facilitate learning of CS (and even learn CS along with students) through invention projects. These teachers' abilities to facilitate, guide, and coach students were cited as being paramount to students' success. The teachers did not attribute their students' success to their own technical skills or ability to code; rather, in their role as facilitator, teachers offered accounts of having guided students to resources so that they would have what they needed to invent. Through the analysis of teacher interviews, we traced the teachers' expertise in facilitating the learning process recognized as IvE. As Resnick (2017) noted, "there's a need for 'experts' in the learning process," no matter how much students learn on their own or are supported by their peers.

Recommendations

During the conduct of this study, as researchers and as staff that work with CS and IvE programs, we attempted to set aside ethnocentrism so that we could understand the emic perspectives of students and teachers in the study. We were conscious, however, of the ways the information was helping us see implications for our own work. We provide the following recommendations on essential support for invention educators and students who wish to create inventions that include computational artifacts:

Recommendation 1: Transdisciplinary Collaboration

We recommend that student invention teams consist of members with diverse expertise, and that team facilitators bridge gaps in their own knowledge and that of their students by fostering collaborations among diverse stakeholders in their communities.

Computational invention projects are transdisciplinary in nature. Considering and incorporating views from the invention and CS disciplines is essential to producing inclusive inventions with computational artifacts. Meanwhile, including diverse members on teams working to invent can ensure that students are exposed to a range of ideas, ways of thinking and approaching problems, and skills. Every team member, including the teacher, brings his/her expertise to invention. Collaboration ensures that students are engaged in navigating and negotiating conflicting ideas, disparate skills, and distinct personalities. Invention is an authentic learning experience where students learn to cultivate working relationships, create team norms, and evaluate their work with others while learning from each other.

Recommendation 2: Make Visible to Teachers the CS/CT Concepts and Practices Utilized in Invention Projects

We recommend the creation of case studies to communicate how invention projects and CS/CT are connected.

Teachers from a wide variety of backgrounds and experiences possess the facilitation and coaching skills utilized by the educators in this study. These three teachers did not have explicit training in CS education, yet their teams were successful at creating computational artifacts in pursuit of invention. Making visible the connection between invention projects and CS/CT can help teachers see those connections and can help them develop confidence in their ability to support students in CS/CT, even if they do not have a technical background in these areas. The case studies can help curriculum developers and program administrators envision new pathways for teaching CS/CT through invention education. This goal can be accomplished through the production of multimedia case studies that are useful tools for teacher learning (Hewitt, Pedretti, Bencze, Vaillancourt, & Yoon, 2003) and for creating general awareness.

Recommendation 3: Provide All Students, Especially Females and Students From Underrepresented Backgrounds, With Opportunities to Engage in Invention Education Across All Years of Schooling

We recommend invention education programs that engage young children, middle school-aged youth, and high school students in learning and applying computational thinking skills through invention.

Research has suggested that early exposure to innovation has a significant causal effect on a child's propensity to become an inventor (Bell et al., 2019). People who, in their childhood, were exposed to innovations through family, neighborhood, and environment are more likely to innovate than those who were not. With advances in technology, computation has been considered an essential 21st-century skill and a main driving force of innovations in the current and future society. It is, therefore, imperative to provide all young children with opportunities to experience computational invention projects.

Additionally, exposing all learners to computation and invention in childhood and early adolescence helps address the diversity and equity issues that are pertinent to the invention and STEM education fields. Many female InvenTeams students expressed that the invention experience reshaped their views of the roles that computation and technology play in people's lives. They were motivated to further their CS education and explore the CS field. Research studies have demonstrated that early middle school years are a critical time for forming youths' attitudes toward STEM (e.g., Maltese & Tai, 2011). Exposure to computational invention activities can influence their future career interest in STEM and cognate fields. Meanwhile, according to Bell et al., if women, minorities, and children from low-income families were to invent at the same rate as white men from high-income families, there would be four times as many inventors in America as there are today (2017). The gender gap in innovation would shrink at a much faster rate and reach gender parity in 18 years, instead of the current rate of 118 years.

Recommendation 4: Support Teachers' Facilitation of Invention Projects With Computational Artifacts by Introducing Them to CS Sources for Students' Just-In-Time Learning

With the rapid expansion of computing education in mainstream K–12 schools, the informal learning space can be a resource and partner in helping teachers facilitate opportunities for just-in-time learning. Those who have experienced CS education in both in-school and out-of-school spaces recommend that schools introduce concepts during the day, while out-of-school spaces such as museums, libraries, and afterschool programs give students opportunities for deeper, project-based, and relevant learning experiences.

A partnership with informal learning spaces can provide opportunities for interest-driven real-world learning, problem-solving, creativity, experimentation, agency, flexibility, and equity that may not be possible in schools. For example, robotics, Girls Who Code, Black Girls Code, Minecraft, Scratch, Fab Labs, Makerspaces, Boys and Girls Clubs, 4-H, Girl Scouts, and museums are examples of programs and organizations that engage young people in some kind of CS education, but may not identify as such.

Summary of Findings and Implications for Future Studies

This exploratory research provided evidence of the ability to support students' learning of CS concepts and practices through a process in which teams of high school students (InvenTeams) work to create a technological invention. Survey data from all students engaged in the 15 InvenTeams showed that, at the end of the school year, students reported differences in their skill levels with using technical tools. These technical tools included HTML, Python, App Inventor, SolidWorks, AutoCAD, Fusion 360, Arduino, and machine learning, although we cannot attribute all of the growth to being on an InvenTeam since students may have engaged in other learning opportunities. A review of the 15 InvenTeams projects in the 2018–2019 year revealed that 14 involved the use of technologies, with five being classified as having Internet of Things (IoT) components. An in-depth examination of three teams whose projects included elements categorized as IoT provided evidence of four out of five CS concepts and all seven CS practices. Concepts included Computing Systems, Networks and the Internet, Data and Analysis, and Algorithms and Programming. The only concept not evident was Impacts of Computing. Practices included Fostering an Inclusive Computing Culture, Collaborating around Computing, Recognizing and Defining Computational Problems, Developing and Using Abstractions, Creating Computational Artifacts, Testing and Refining Computational Artifacts, and Communicating about Computing.

Our findings suggest that CS can be taught through an invention education project, particularly if the project aims to produce a technological solution to a problem. The projects selected for the telling cases were limited to those with IoT components. Future studies could examine CS concepts and principles addressed by projects that involve other types of technologies. Future studies could also examine the "learning" of concepts and practices that we, as researchers, could document through the various forms of data and artifacts to determine ways of collecting information in the future for new forms of student assessment.

The research also made visible the support students received from invention educators guiding their work. The accounts of teachers' lived experiences and ways in which the teachers facilitated the work of their teams can inform understandings of the types of teaching practices and knowledge teachers must have in order to support students' work as inventors. The facilitation strategies included establishing mutual trust; a hands-off, student-centered approach with flexibility; providing resources; and coaching. It is important to note that teachers cited the community as a resource for knowledge that was needed by teams but exceeded that which the teacher possessed. This suggests that the community becomes a resource for invention and CS pursuits; but, who in the community was a resource, and how and in what ways they supported the students, remains to be studied. Implications of teachers as facilitators—assisting students with content areas that may be new to the teachers themselves—is also a topic for future studies, given the implications for teacher preparation and professional development programs.

The effort to understand ways IvE and CS education align (or not) represents a first step in examining the different disciplines and new framework for conceptualizing transdisciplinary teaching and learning in an age of convergence. Similar studies can be conducted to explore ways science and engineering concepts and practices, for example, are learned through inventing. Research that illuminates intersections or areas of overlap between the intentions and outcomes of particular education initiatives can help educators develop greater awareness of opportunities for collaboration. An understanding of similarities between disciplinary aims or aims of particular fields of study can highlight possibilities for educators involved in each effort to align teacher- and student-focused efforts in ways that may allow for a greater collective impact. A united, collective effort may help with overcoming the barriers that continue to prevent widescale adoption of both CS and IvE practices, while also fostering greater coherence in the messaging surrounding the resources needed to support both fields.

About the Researchers and Authors of this Paper

Four of five researchers are affiliated with the Lemelson-MIT Program. The program was established at the Massachusetts Institute of Technology 25 years ago. It celebrates outstanding inventors through awards and inspires young people to pursue creative and inventive lives through grants initiatives. The program has worked with educators to help young people learn to invent for the past 15 years. More than 6,000 students have participated in the program's two youth initiatives, JV InvenTeams and InvenTeams. The program is administered by MIT's School of Engineering and funded by the Lemelson Foundation.

One of the five researchers is affiliated with CSforALL. CSforALL is the national hub of the computer-science-for-all movement, with a mission to make high-quality computer science an integral part of K–12 education in the United States. Their three-pillar approach includes

Support Local Change, Increase Rigor and Equity, and Grow the Movement, and directs the work across a national and local spectrum to provide equitable and accessible K–12 computer science education to every student. CSforALL engages with diverse stakeholders leading computer science initiatives across the nation to support and facilitate implementation of rigorous, inclusive, and sustainable computer science.

Special Appreciation

We wish to thank The Lemelson Foundation for the generous support they provided for our work. One-time grant funding made it possible for two of the researchers who are not part of the LMIT staff to collaborate on this study, thereby diversifying the range of expertise represented within the team. The topics explored were sufficiently complex that no one researcher could have completed the study in isolation.

References

2018 State of Computer Science Education. (2018). Policy and Implementation. Retrieved from https://advocacy.code.org/

Ashton, K. (2015). *How to fly a horse: The secret history or creation, invention, and discovery*. New York, NY: Doubleday.

Bell, A. M., Chetty, R., Jaravel, X., Petkova, N., & Van Reenen, J. (2019). Who becomes an inventor in America? The importance of exposure to innovation. *The Quarterly Journal of Economics*, *134*(2), 647–713.

California Department of Education. (2019). *California computer science strategic implementation plan*. Retrieved from https://www.cde.ca.gov/pd/ca/cs/cssip.asp

College Board (2018). *Number of females and underrepresented students taking AP computer science courses spikes again*. Retrieved from https://www.collegeboard.org/releases/2018/number-of-females-and-underrepresented-students-taking-ap-computer-science-courses-spikes-again

Couch, S., Estabrooks, L., & Skukauskaite, A. (2018). Addressing the gender gap among patent holders through invention education policies. *Technology and Innovation*, *19*(4), 735-749. DOI: 10.21300/19.4.2018.735

Creswell, J. W. (2012). *Educational research* (4th ed.). Boston, MA: Pearson.

Estabrooks, L. B. & Couch, S. R. (2018). Failure as an active agent in the development of creative and inventive mindsets. *Thinking Skills and Creativity*, *30*, 103–115. DOI: 10.1016/j.tsc.2018.02.015

Fraillon, J., Ainley, J., Schulz, W., Duckworth, D., Friedman, T. (2019). *IEA international computer and information literacy study 2018 assessment framework*. Amsterdam, The Netherlands: Springer International.

Gershenfeld, N. (2008). *Fab: The coming revolution on your desktop—from personal computers to personal fabrication*. New York, NY: Basic Books.

Green, J. L., Skukauskaite, A., & Baker, W. D. (2012). Ethnography as epistemology. In J. Arthur, M. J. Waring, R. Coe, & L. V. Hedges (Eds.), *Research methods and methodologies in education* (pp. 309–321). London, UK: Sage.

Hewitt, J., Pedretti, E., Bencze, L., Vaillancourt, B. D., & Yoon, S. (2003). New applications for multimedia cases: Promoting reflective practice in preservice teacher education. *Journal of Technology and Teacher Education*, *11*(4), 483–500.

Invention Education Research Group (2019). Research Invention Education. Retrieved from https://inventioneducation.org › ResearchingInventEdu-WhitePaper.

K–12 Computer Science Framework. (2016). Framework view by concept, abridged. Retrieved from http://www.k12cs.org

K-12 Computer Science Statements of Support, n.d. Retrieved from https://k12cs.org/statements-of-support/

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., … Werner, L. (2011). Computational thinking for youth in practice. *Acm Inroads*, *2*(1), 32–37. https://doi.org/10.1145/1929887.1929902

Maltese, A. V., & Tai, R. H. (2011). Pipeline persistence: Examining the association of educational experiences with earned degrees in STEM among US students. *Science Education*, *95*(5), 877–907.

McAlister, A. M., Lee, D. M., Ehlert, K. M., Kajfez, R. L, Faber, C. J., & Kennedy, M. S. (2017, June 25–27). *Qualitative coding: An approach to assess inter-rater reliability*. Paper presented at the 2017 ASEE Annual Conference & Exposition, Columbus, OH.

Mitchell, J. (1983). Case and situation analysis. *The Sociological Review, 31*(2), 187–211.

Mitchell, J. (1984). Typicality and the case study. In R. Ellen (Ed.), *Ethnographic Research: A guide to general conduct* (pp. 237–241). London, UK: Academic Press.

NRC (National Research Council). 2014. *Convergence: Facilitating transdisciplinary integration of life sciences, physical sciences, engineering, and beyond*. Washington, DC: The National Academies Press.

Resnick, M. (2017). *Lifelong kindergarten: Cultivating creativity through projects, passion, peers, and play*. Cambridge, MA: MIT Press.

Saldana, J. (2013). *The coding manual for qualitative researchers* (2nd ed.). Los Angeles, CA: Sage.

Rubin, H. J., & Rubin, I. S. (2012). *Qualitative interviewing: The art of hearing data* (3rd ed.). Los Angeles, CA: Sage.

Sawyer, K. (2019). *The creative classroom*. New York, NY: Teachers College Press.

Spradley, J. P. (1980). *Participant observation.* Fort Worth: Harcourt Brace.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35.

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *366*(1881), 3717-3725.

Witherspoon, E. B., Schunn, C. D., Higashi, R. M., & Baehr, E. C. (2016). Gender, interest, and prior experience shape opportunities to learn programming in robotics competitions.

International Journal of STEM Education, 3(1), 18. https://doi.org/10.1186/s40594-016-0052-1