Priming the Pump: Reflections on Training K-5 Teachers in Computer Science

Michele Roberts
Indiana University
901 East 10th Street
Bloomington, IN 47408
microber@iu.edu

Kiki Prottsman
Code.org
1501 4th Ave., Suite 900
Seattle, WA 98101
kiki@code.org

Jeff Gray
University of Alabama
Box 870290
Tuscaloosa, AL 35487
gray@cs.ua.edu

ABSTRACT

Much well-deserved attention in K-12 Computer Science (CS) education has focused recently on the successful launch of the College Board's new AP CS Principles course, which is breaking participation records and broadening CS participation. To further leverage the national investment in a successful high school CS program, however, it is important to create, sustain and study a continuous CS pipeline that begins early and spans all grade levels. This experience report articulates the characteristics of Code.org's K-5 CS Fundamentals (CSF) program and summarizes the experiences of adopting the CSF curriculum to support large-scale, university-driven K-5 Professional Development (PD) programs across two states in different geographical regions of the USA. An overview of Code.org's CSF curriculum and PD survey data is provided, followed by a summary of each state's experience. A set of lessons learned offers recommendations for those considering implementation of statewide PD programs in K-5 CS; future plans are discussed to investigate observations from this experience report within a formal research setting.

ACM Reference format:

Michele Roberts, Kiki Prottsman, and Jeff Gray. 2018. Priming the Pump: Reflections on Training K-5 Teachers in Computer Science. In *SIGCSE '18: 49th ACM Technical Symposium on Computer Science Education, Feb. 21–24, 2018, Baltimore, MD, USA.* ACM, NY, NY, USA, 6 pages. https://doi.org/10.1145/3159450.3159560

1 INTRODUCTION

The recent surge of interest in K-12 Computer Science (CS), particularly from the large number of students participating in the first offering of the AP CS Principles (CSP) exam in May 2017, highlights the need for increased availability of a CS pathway of experiences across all K-12 grade bands. Concepts that were previously taught only in high school AP courses (e.g., algorithms and core programming language constructs), or taught first in introductory college level courses, are now being introduced at all K-12 grade bands using age appropriate curricula with engaging activities [1]. Furthermore, a new focus on equity and serving the needs of a diverse set of learners has also materialized as a key thread within the CSforAll movement, driving the development of new curricula and teacher professional development (PD) strategies.

CS PD efforts face several challenges. While many education programs offer pre-service preparatory coursework in digital literacy, few programs offer preparation in CS. Due to the long-standing lack of CS pre-certification pathways, states have large "install bases" of in-service teachers with little or no training in CS. Complicating this challenge, few university College of Education faculty possess CS content knowledge (CK). Several third-party curricula and PD providers have emerged with different approaches for teacher preparation. In this experience report, we summarize our experience as a curriculum author (Prottsman) and university CS faculty (Roberts and Gray) who utilized Code.org's CS Fundamentals (CSF) curriculum and PD plan to support large-scale K-5 outreach programs in two different states. We share our common and individual experiences in training over 3,000 teachers across nearly 100 workshops. As an experience report, our observations are not as formal as a deep research study, but we offer a glimpse into the unique opportunities and common challenges that we faced during three years of outreach to elementary school teachers, and advocate for CS university personnel to partner with third-party CS educational providers to accelerate delivery of a K-12 pipeline that produces CS teachers who will then train a new wave of K-5 computational thinkers.

In the next sections of this experience report, we provide an introduction to the Code.org CSF curriculum (Section 2) and an overview of the efforts to bring the curriculum to two states (Section 3). Code.org's national workshop survey data is summarized and compared to workshop data from our two respective states (Section 4). A series of lessons learned are described (Section 5), including an argument for increased CS faculty involvement in CS education, leading to concluding remarks and future research plans (Section 6).

2. OVERVIEW OF CODE.ORG'S CS FUNDAMENTALS CURRICULUM

In February of 2013, a father looking to create interest in adding computing to his children's school curriculum posted a YouTube video called "What Most Schools Don't Teach" [2]. Hadi Partovi's video was viewed by more than 12 million people in the first two weeks and tapped into public interest emerging from a National Science Foundation (NSF) initiative to create K-12 computing pathways. Continuing the momentum created by the video, Partovi established a non-profit organization called Code.org with the mission of "... expanding access to CS and increasing participation by women and underrepresented minorities." To support an

SIGCSE '18, February 21–24, 2018, Baltimore, MD, USA
© 2018 ACM ISBN 978-1-4503-5103-4/18/02…$15.00
https://doi.org/10.1145/3159450.3159560

awareness-raising Code.org event called Hour of Code, Partovi partnered with Kiki Prottsman, founder of a children's computing curriculum called Thinkersmith, to produce a basic, free computing course that teachers could utilize in the classroom. Participation in the Hour of Code was widespread and immediate, soon involving over 14 million participants. Not only did teachers utilize the software released for the Hour of Code, but they also began to use the accompanying Thinkersmith curriculum. Recognizing both the teacher interest in computing and the national scarcity of pre-certification CS pathways, Code.org expanded the curriculum to a four course series and began to offer free, one-day PD workshops to K-5 educational stakeholders interested in learning the CSF course content. In 2014, one-hundred CS education enthusiasts were invited to join Code.org as K-5 CSF Facilitators, and were introduced to the CSF curriculum and PD mission at a national training program.

The CSF curriculum and the PD workshops that supported it were grounded in CS educational research and aligned with the broadening participation initiative that has been key to the NSF's directive around creating CS K-12 pipelines. The CSF curriculum was situated within frameworks such as the ACM/CSTA K-12 Task Force Curriculum Committee [3], and other similar inventories of recommended K-5 CS subject matter. In CSF, programming content is delivered following the style of "Parson's Programming Puzzles," a teaching method that reinforces student programming mastery [4]. Key concepts in the CSF curriculum include computational thinking [5]; problem solving [6]; programming (algorithms, abstraction, control structures, variables, functions, event triggers); introductory data representation; and digital citizenship.

Four key elements are particularly noteworthy about the Code.org CSF curriculum: (1) Computing activities are placed within the broader context of computational thinking; (2) Instruction includes both online as well as offline, "unplugged" activities (the latter concept established by the CS UnPlugged curriculum made available in the public domain in the early 1990s [7]). The unplugged activities are typically delivered in discovery-based group activities, furthering knowledge mastery and reinforcing the collaborative facet of computing; (3) Children are introduced to programming via a block-based language, in which much of the required "punctuation" of a language is abstracted away by instructional blocks, which are then dragged and connected into programs. Block-based languages are accepted by many as an efficient on-ramping platform to teach programming skills to beginners [8]; (4) The CSF concepts are introduced simply in early courses, and iteratively refined in later courses, supporting a spiral model that scaffolds foundational ideas for subsequent deeper learning.

The four key curricular features are delivered such that students progress through the content by completing scaffolded online coding puzzles and engaging in offline, discovery-based activities. Teachers attending the CSF PD workshops are provided with a free textbook with numerous unplugged lesson plans that span multiple courses, a swag bag, and food. Those completing the workshop are provided a certificate and an opportunity to complete a survey. Code.org will postal mail each survey respondent a box of resources that support the unplugged activities in the textbook.

Pedagogical Knowledge (PK) is similarly embedded deeply in the CSF PD. Because teacher pre-certification computing pathways are scarce in the US, the curriculum was designed for inexperienced CS educators. Teacher lesson plans and resources (including teacher-facing demonstration videos such as a video on the "Conditional Cards" activity that is discussed later in Section 3 [9]) are provided for the curriculum content. To support the national initiative of broadening participation and in keeping with their organizational mission, the Code.org CSF curriculum was specifically designed to help broaden participation with the inclusion of activities that foster collaboration, student engagement and equity. The training workshop agenda is built around a Teacher-Learner-Observer (TLO) model, influenced by [10] and successfully implemented in PD workshops for the landmark Exploring Computer Science (ECS) course [11]. TLO is essentially a role-playing activity in which workshop attendees (teachers) break into groups and take turns as a teacher guiding a lesson, or as a student engaged in active learning. Meanwhile, a Code.org-trained facilitator observes the role-playing session, providing thought-provoking questions at the end of the exercise. The facilitator's goal is not to coach teachers on their lesson delivery, but to re-focus participants' thoughts on their own teaching practice and the challenges specific to delivering the curriculum at their schools with their students. This not only gives educators the ability to plan lessons in a friendly environment, but also allows them to empathize with learners, looking for potential areas of engagement as well as distraction. A somewhat hidden but critically important benefit of the TLO model is that it provides teachers with a comfort zone to explore their understanding of a fundamental concept in a student role, as well as experiment with effective ways to teach the fundamental concept while in the role of teacher. From the perspective of Code.org, the TLO model has been particularly beneficial to a mission of accelerating novice teacher preparation.

2.1 A Focus on Program Assessment

Program assessment has been a key part of Code.org's practice since its founding. After each PD workshop, attendees are asked to complete a short, post-treatment Likert scale survey designed to tap several important teacher measures (as well as facilitator effectiveness and workshop logistics). To encourage responses, surveys are typically requested at the close of PD workshops, resulting in nearly a fifty percent response rate (nationally). Survey responses are overwhelmingly positive. The data confirm the assumption that attendees are typically new to CS, with an average rating of 5.34 out of 6 from respondents indicating that "This workshop was suitable for my level of experience in teaching CS." Workshops have resulted in a significant teacher confidence gain, with an average response rating of 5.37 out of 6 to the survey item "I feel more prepared to teach the material covered in this workshop than before I came." As a strong measure of workshop evaluation, the survey item "I would recommend this workshop to others" received a nationally averaged rating of 5.53 out of 6.

As the national computing landscape began to change, with states beginning to implement policy changes supporting CS education, Code.org contracted with an outside evaluator to obtain a more detailed, independent review of program effectiveness. In 2015, University of Chicago's Outlier Research and Evaluation group published their 2014-2015 results of survey and interview data gathered from stakeholders in partner sites (i.e., schools or districts with a formal agreement with Code.org) [12]. Data from 744 completed surveys from a 7,000 respondent sample size of K-5 stakeholders provided valuable insight into first-year CSF program implementation [12]:

- Respondents were largely (almost 75%) technology "facilitators" working with multiple sections of students;

- Respondents reported challenges identifying which courses should be used in which grade levels given that students had no prior exposure to computational themes in a formal course. Course 3, for example, presumes completion of Course 2; but until a local pipeline is established, prior experience is missing. With some overlap, most respondents reported beginning grades K-2 in course 1, starting grades 3-4 in course 2, and grade 5 in course 3;
- Computing content was delivered almost exclusively (83%) in a "computing class," even though 65% of those interviewed "agreed" or "strongly agreed" that connections between computing and other K-5 subject matter areas were easily drawn;
- While only 17% of survey respondents reported integration of the curriculum to supplement other subject areas, math was the most likely supplemented subject (65%);
- Teachers reported that students found the curriculum "engaging," and those teachers who were interviewed noted some broadened participation, both from girls as well as some students disinterested in math or science.
- More than half of the respondents indicated a need for more content knowledge, with 55% disagreeing or strongly disagreeing with the statement "I have all the skills I need to teach Computer Science well."

In response to initial feedback and scaling pressures brought by increasing enrollments, the Code.org CSF curriculum was expanded in Q3 2017 to six courses directly aligned to grades K-5. Over 59 thousand stakeholders have attended CSF workshops to date. Internal Code.org staffing has grown from a 15 to a 60-person team; K-5 CSF Facilitators (i.e., those who are trained to offer PD workshops in their regional areas) have increased from the original 100 to 175. From its successful launch, Code.org has continued to expand and evolve.

3. CS Fundamentals PD Across Two States

This section summarizes the university-driven K-5 PD Outreach programs in both Midwestern (Indiana) and Southern (Alabama) states in the USA. Both facilitators are university faculty involved in CS K-12 outreach for over a decade. In addition to participating as official pilots in the College Board AP CSP course development and PD, the faculty have also worked to provide teacher training at the elementary and middle school levels through various summer workshops and other opportunities. As the NSF CS-10K initiative gained momentum, both faculty discussed emerging elementary school computing resources (e.g., Scratch, the Foo's, Tynker) and targeted the Code.org CSF curriculum for K-5 outreach work, selected because it was free, deeply grounded in CS educational research, included both CK and PK, and was well-received by teachers participating in pilot outreach programs. Both faculty completed CSF facilitator training and began offering workshops in their respective states.

Code.org provides K-5 facilitators with a recommended workshop template that includes, among other content, a TLO activity, exploration of coding puzzles and a discussion about broadening participation in computing. While both faculty have used numerous TLO activities, one of our most utilized is from a lesson plan that uses playing cards to demonstrate how conditionals work within selection constructs in programming languages. In this activity, the facilitator introduces the control structure operation of selection and provides an everyday example (e.g., traffic lights). Attendees participate by decomposing traffic light operation into the conditional component and the consequence of the conditional (e.g., if "the light is red" then "stop"). Next, teachers divide into TLO groups, are given a deck of playing cards, and are asked to develop a short lesson using the cards that would help students grasp how conditionals work within selection. The lesson is presented by the role playing teacher group to the role playing student group, and then the facilitator offers comments or questions for group discussion and reflection. Sample facilitator input might be "Great presentation -- you have Selection spot on! Thinking about your classroom, are there ways in this lesson for you to provide students with opportunities for engagement and discovery?" Role playing teachers might then give cards to the student group, challenging them to create a conditional selection activity to share with the class.

To illustrate coding puzzles to the participants in our CSF workshops, we introduce attendees to the curriculum, puzzle layout, and block language operations, and then allow free time for exploration. Some specific puzzles are called out for additional attention, such as the back-to-back puzzle presentation in the second course that requires the same puzzle to be solved first with sequential statements only, then with the additional efficiency offered by a loop. Attendees are asked to reflect on how they might support student realization that the two puzzle solutions are logically equivalent.

For the broadening participation discussion, we introduce an exercise called "The Giraffe and the Platypus," a training activity designed to help understand "preparatory privilege" [13] in the classroom. In the exercise, two worksheets are distributed, one skewed to giraffe facts and one skewed to platypus facts. Workshop attendees are unaware there are two separate worksheets. Attendees are given a few minutes to read the worksheet material, and then the facilitator asks questions. The first question has answers available on both worksheets, but subsequent questions have answers available on only one of the worksheets. Teachers see preparatory privilege first hand, experiencing what it feels like to be a "knowledge have" or "have not," with subsequent discussion around strategies for fostering more equitable CS classrooms.

3.1 Snapshot: Indiana

In Indiana, K-5 PD outreach utilizing the Code.org CSF curriculum began in Q3 of 2015. Since program launch, over 1,517 teachers and related Indiana stakeholders have completed the free, one-day training. Training sessions have been scheduled around the state in response to local requests, as well as offered through Indiana's regional state educational service centers and onsite at supportive universities. In addition to hosted workshops around the state, "walk in" workshops have been frequently offered in centrally located Indianapolis to provide even greater access and opportunities. Along with free curriculum materials and "swag bags" provided by Code.org, workshop attendees receive Professional Growth hours issued on university letterhead and are treated to breakfast foods and afternoon snacks. Unlike the data collected from selected 2014-2015 partners, Indiana workshop attendees are not predominantly tech coordinators; instead, workshop attendees represent a variety of staff roles, including teachers (the majority), tech coordinators, media specialists, curriculum leads, and in a few cases, school administrators.

A key and early collaborator in Indiana' K-5 PD initiative has been the state Department of Education (DOE), which is deeply committed to CS education. From program onset, the DOE has supported CS PD opportunities across Indiana, attending Code.org workshops, providing valuable insights into related state and local programs and policies, and communicating PD schedules through various state agency distribution channels. In April of 2016, the DOE championed successful passage of K-8 standards in CS, which went into immediate effect. The standards are articulated by grade band (K-2, 3-5 and 6-8); within each grade band, knowledge mastery is listed within five core concepts: Data and Information; Computing Devices and Systems; Programs and Algorithms; Networking and Communication; and Impact and Culture. The DOE utilized deliberate and thoughtfully designed launch strategies for the CS standards: no new, CS-specific course codes were introduced, effecting a plan that both encouraged all teachers to acquire foundational computational skills and provided maximal flexibility to school districts in assigning computing roles and responsibilities; to allow for rational adoption plans and buy in, assessment to the state standards was deferred the first year; and the CS standards were released as part of (required) Science standards revisions, communicating the importance of CS education to the DOE's strategic vision.

The Indiana faculty reviewed the Code.org CSF curriculum against the new CS state standards and minor gaps were identified (such as in the Data and Information category). Supplemental CK materials were prepared to address the gaps. The CSF PD workshop agenda used across Indiana was then modified to include content that "unpacked" the state standards and provided explicit alignment between the standards and all workshop materials. Fidelity was maintained with Code.org's recommended workshop practice of a TLO activity, coding puzzle exploration and broadening participation in CS. Indiana workshops end with a reflection period after which teachers create a personalized, bulleted plan for "next steps." Attendees leave the CSF workshop with self-reported foundational competencies in CS content and method knowledge; a collection of standards-aligned online and unplugged activities that can be implemented immediately; sample lesson plans for CS integration into other subjects; and training on how to utilize the freely available Code.org online Code Studio within their classroom. Follow-up email to each participant provides additional resources, earned PD certificates, an invitation to join the DOE Community of Practice for Indiana computing teachers, and faculty contact information to answer questions or address concerns.

3.2 Snapshot: Alabama

The CSF facilitator from Alabama was trained in the early 2014 Code.org CSF cohort, which led to the first Alabama workshop offered in December 2014. Over the past three years, 57 Alabama workshops have been attended by nearly 1,575+ participants, with average attendance of 27 participants (max 43). The majority (53%) of the workshops in Alabama have been hosted at elementary schools and attended by their own district teachers, as well as those from other regional districts. Regional district administration offices hosted 34% of the workshops, with only 13% of the workshops held at an Alabama university. Most of the participants were classroom teachers, but the Alabama workshops also had representation from tech coaches, in-service specialists, and administrators. In Alabama, many of the librarians and media specialists have authority over their K-5 computer lab and frequently attended the Alabama workshops with their colleagues who were classroom teachers. The Alabama facilitator moderates a Google Group for teachers who attended past workshops, but only 18% of workshop attendees are active.

Workshop recruiting in Alabama has been pleasantly easy; almost all workshops fill and overflow to a waiting list. Advertisement for upcoming workshops typically comes by word of mouth as past attendees request workshops be brought to their districts. Additionally, 1-hour overview sessions are offered at an annual state tech conference, which has been a great venue for advertising workshop opportunities. Over the last year, university in-service centers have started to advertise the workshops on their email distribution lists.

Several Alabama politicians have offered public support for CS education in a series of very influential announcements. An increased interest in new workshops is emerging, driven by school administrators who now have an anchor for understanding the importance of CS for Alabama students. In 2017, the Alabama DOE created K-12 CS standards that were under public review in Fall 2017. The Alabama CS standards for the K-5 grade band will be mapped to the CSF curriculum.

In addition to the free materials provided by Code.org (e.g., textbook and swag bag), the Alabama facilitator provides snacks, a nice lunch, a customized t-shirt for K-5 CS teachers, and door prizes at the end of each workshop.

4. Common Evaluation Measures

This section provides a brief review of Code.org post-workshop measures, focusing on four key PD metrics: participation, confidence, CK and PK gains, and teacher changes in beliefs and values toward CS.

Code.org provides a post-PD survey to all attendees to capture several key metrics of CSF workshop effectiveness. As Code.org has become an increasingly significant provider of CS educational solutions, programmatic content and formal assessment practices have evolved, changing in 2017 to capture more granular data that is better aligned to traditional educational measures.

It is important to note that Code.org workshop reviews are overwhelmingly positive. In fact, a significant number of respondents ranked the workshops as the "absolute best PD" they ever attended (76%). The fact that workshop reviews (aggregated from around the USA and involving over 25 thousand participants completing workshops led by over 175 different facilitators) are so consistently high appears to affirm the selection of the TLO PD model, the positioning of the Code.org CSF curriculum, and the quality of facilitator training. In the following sections, the aggregated university-led workshop survey data from both faculty is compared to aggregated results from the same surveys administered across all K-5 Code.org CSF workshops nationally.

Compared to standard CSF workshops, university-driven workshop programs in both states contain deeper CS content, are aligned to the K12 CS Framework, and include an invitation to join local communities of practice. Interestingly, reviews from teachers in both university-led state outreach programs were very similar to each other, and reflected the national measures in all but three survey items. In the following discussion, survey results are compared and contrasted, and potential reasons for observed differences are suggested. In this experience paper, we offer these findings as directional guidance only and plan a quantitative analysis for future research publication.

4.1 Teacher Participation and Engagement

It is our experience that teacher participation and engagement during PD increases potential impact; in the case of TLO-modelled

CS PD for novice teachers, engagement and participation may be even more important. Code.org CSF workshop post-surveys include several questions related to engagement and participation, with two critical questions being: (1) "During your workshop, how often did you participate?" (Nationally aggregated score: 3.97 out of 5) and (2) "How often did you get so focused on Code.org workshop activities that you lost track of time?" (Nationally averaged participants score this question at 3.46 out of 5). Workshop participation and engagement measures in both faculty states aligned with nationally aggregated values.

4.2 Teacher Confidence

There is prolific research exploring teacher self-efficacy and technology integration, but educational research exploring the relationship between teacher self-efficacy in CS and classroom practice is still emerging. Drawing inferences from the literature in self-efficacy and science pedagogy [14], however, supports the common sense view that teacher confidence in CS is a predictor of classroom adoption. The Code.org CSF survey includes a direct question: "I feel confident to teach the Code.org CS courses." The national workshop average was 4.46 out of 5. Interestingly, our CSF workshop participants scored this survey statement lower than the nationally averaged surveys (Indiana: 4.33 out of 5 and Alabama: 4.27 out of 5). While the slightly lower self-efficacy scores were disappointing, we note there was only mild erosion of the percentage of workshop attendees intending to teach the courses. At Indiana, for example, the percentage was 80% vs the nationally averaged 82%.

4.3 Teacher CK and PK Gains

The TLO model blurs CK and PK; when workshop attendees respond to knowledge gains, it is unclear whether they are referring only to CK, PK, or both. As no survey questions explicitly tapped PK, we combine these categories for discussion purposes. Reported content knowledge gains in the university-driven workshops tracked confidence scores, with both state workshop participants reporting lower CK gains than the nationally averaged scores (Indiana scored 6.19 and Alabama scored 6.15, compared to the national average 6.51, all out of 7). It is unclear whether our lower self-efficacy and CK/PK scores were a reflection of the deeper content covered in these workshops, or some issue with the pace and coverage of the university-driven workshop agendas.

4.4 Beliefs and Values in Computer Science

Although self-reported confidence and CK/PK knowledge gains were lower in our university-driven workshops than the national averages, self-reported changes in beliefs and values toward CS were substantially higher in both states. At Alabama, for example, the self-reported post-workshop changes in beliefs and values were rated at 7.51, while the nationally numbers were only 7.04, out of 10. This experience report is limited to workshop snapshot data; follow-up classroom observation is required to better measure training impact. The potential inter-relationships between deeper CK exposure, self-efficacy, reported CK/PK gains, CS beliefs and values, and classroom behavior are worth exploration, and will be examined in future quantitative research.

5. KEY LESSONS LEARNED

In this section, we briefly overview several lessons that were learned from the shared and individual observations across the two state PD initiatives.

5.1 Shared Observations

Although there were several differences between the two state programs, most experiences were shared. In both states, key partnerships with the respective DOE brought access to additional and varied communication channels and further endorsements for school districts hesitant to support CS PD. At a more local level, school districts and schools have different processes, personnel assignments and PD models. Understanding the local culture is critical for ensuring computing initiatives will be leveraged optimally. Also important at the local level is funding. While the Code.org CSF workshops are completely free, districts incur costs when workshop attendance requires substitute backfills. Indiana workshops were offered on both weekdays and weekends, but Alabama only offered weekday workshops. Aligning with district-wide PD days is a key way to minimize district costs while avoiding weekend attendance. Not surprisingly, workshops endorsed by schools and or districts were better attended.

Facilitators in both states received significant feedback from teachers that the deeper content knowledge they received was valuable. Only 1 respondent (Indiana) out of a combined total of over 3,000 attendees across both states indicated that "some of the workshop content was over my head"; most workshop feedback that was from open survey responses included statements such as, "learned things I always wondered about," "made connections I didn't know existed," "would love to learn even more about this." Many attendees in both states asked for advanced CK workshops as follow-up training. As an example of additional interest, registration for an Indiana hosted state K-8 CS Teacher Conference filled within a few weeks. Although teachers asked for follow-up workshops, they rarely participated in the online communities of practice offered by facilitators in both states. Community participation picked up slightly in Indiana after standards were passed.

We also received numerous requests for implementation guidance. Specific requests for recommendations included how to bring up a pipeline (e.g., the entire school at once, an initial focus on lower grades and moving those trained students up through the grade levels); what infrastructure responsibilities were optimal for student success (e.g., could unplugged activities be taught by classroom teachers while online activities were managed by tech specialists); how much class time needs to be allotted for computational work, with most teachers indicating that less than 20 hours could be devoted to singularly purposed computing instruction throughout a school year; and how should teachers assess computing work. In both states, teachers voiced a desire for guidance on how to integrate CS topics into multiple K-5 subjects.

All school personnel, especially classroom teachers, found the wealth of Code.org CSF resources reassuring (e.g., the brief online videos that are available as summaries of each activity in the textbook, and the detailed lesson plans). Facilitators in both states agree that workshop time devoted to viewing the Code.org CSF teacher resources has been well-spent, even though that time carried an opportunity cost for other workshop content that would have been beneficial to include.

In addition to the CSF resources, we believe peer discussion and the TLO philosophy helped us to build self-efficacy and confidence among the teachers on coding puzzles and curriculum activities. Several teachers mentioned in the open response section of the post-survey that they felt anxiety at the start of training, but left the workshop with a much better feeling of their ability to lead CSF activities in their own classroom.

5.2 Observations from Indiana

To date, Code.org CSF surveys show the PD outreach to be consistently well-received, with marked affect changes in self-confidence and self-reported gains in CK and PK. Three key lessons emerged for Indiana PD: (1) the workshop framework must address the reality that additional curriculum content means "yet one more thing" that teachers have to do. Attending participants are provided with a spreadsheet containing the state standards, with an implementation column indicating content already being taught as is, content already being taught without a computing vocabulary (such as "algorithms"), and existing curriculum gaps (such as programming); (2) the ubiquity and creativity of CS lends itself well to integration across multiple subjects. By explicitly helping teachers to discover this during TLOs, teachers reported that the task of teaching CS is not quite as daunting as they originally feared; (3) it seemed helpful for teachers to identify and share with the workshop group at least one specific next step they plan to take toward teaching computational thinking in their classrooms.

5.3 Observations from Alabama

The Alabama facilitator challenges all participants to complete at least 20 coding puzzles before the workshop conclusion, which has been realized by nearly all of the 1,575+ participants. This suggests that teachers with very little CS background are able to increase their CK and self-efficacy even in just a one-day workshop. A very interesting aspect from observing participants during their exploration of coding puzzles was seen in the demonstration of a growth mindset [15]. Participants are often heard voicing their frustration in not completing a puzzle correctly, but in a way that suggests a level of perseverance or grit (e.g., a teacher who is frustrated by being taunted by one of the Pigs at the unsuccessful execution of an Angry Birds puzzle is motivated to push forward and use problem solving strategies introduced in the training to debug their program). We have seen the same exhibition of growth mindset when observing K-5 students on the same coding puzzles.

A particularly beneficial workshop strategy has been to complete a detail-driven unplugged activity before beginning programming, so that attendees understand the importance of precision in computer science. This "precision" perspective is illustrated in activities such as the popular peanut butter and jelly sandwich building exercise, or in unplugged grid programming, both illustrating the importance of analyzing a puzzle and designing a solution before ever moving a block of code.

Challenges faced by the Alabama facilitator have been related mainly to managing workshops with participants from all K-5 grades. In a room full of teachers who are simultaneously solving coding puzzles across multiple courses, it can be hard to make sure all teachers are sampling as many types of concepts and constructs as possible, while not getting stuck on a single type of puzzle. Although the TLO sessions are often split across different breakout rooms according to grade levels, it is still difficult to visit all rooms and play the role of "Observer" to make sure that the concepts are correctly understood in the presented lessons.

5.4 Benefits of University Led PD

The challenge of creating national K-12 computing pipelines is demanding enough that multiple contributions should be considered, including those contributed by university CS faculty. By utilizing high quality, third-party provider curricula and PD frameworks, CS university faculty can efficiently engage in the stewardship of CS education. CS university faculty bring deep CS content knowledge, have access to college and graduate students who can be trained to support classroom activities, can incubate before and after school clubs, and can mentor students. Perhaps most importantly, the involvement of university faculty increases the likelihood that CS K-5 educational research will accelerate, with critical and foundational work remaining to be done in many areas.

ACKNOWLEDGMENTS

We would like to thank all the K-5 teachers who participated in Code.org workshops, the Alabama and Indiana Departments of Education, Code.org, Baker Franke, and the Computer Science Departments at Alabama, IUPUI and Indiana University.

6. CONCLUSION

In this experience report, two CS university faculty selected Code.org's CSF curriculum and PD to conduct approximately 100 CS workshops over the past three years, reaching over 3,000 teachers in their combined states. While maintaining fidelity with the Code.org CSF PD paradigm, deeper CS context was provided and ongoing support was offered. Post-workshop survey data shows that while attendees reported confidence and CK gains lower than nationally aggregated workshop scores, beliefs and attitudes toward CS changed considerably more. Several future research thrusts were identified, including the apparent inter-relationship between self-reported CK, self-efficacy and belief and value changes; classroom follow-up visits to assess implementation; and the impact of explicit guidance in inter-disciplinary integration of CS. Finally, CS university faculty are encouraged to consider the efficiency in utilizing high quality, third-party CS education providers (such as Code.org) to support the stewardship of CS education in their states.

REFERENCES

[1] Katie Apone, Marina Umaschi Bers, Karen Brennan, Diana Franklin, Maya Israel, and Pat Yongpradit, "Bringing Grades K-5 to the Mainstream of Computer Science Education," In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education (SIGCSE)*, March 2015, Kansas City, MO, pp. 671-672.

[2] Code.org, "What Most School's Don't Teach," YouTube, http://bit.ly/code-wmsdt

[3] *A Model Curriculum for K-12 Computer Science: Final Report of the ACM K--12 Task Force Curriculum Committee.* Technical Report, ACM, New York, NY, 2003.

[4] Dale Parsons and Patricia Haden, "Parson's Programming Puzzles: A Fun and Effective Learning Tool for First Programming Courses," In *Proceedings of the 8th Australasian Conference on Computing Education (ACE)*, Vol. 52, 2006, Darlinghurst, Australia, pp. 157-163.

[5] Jeannette M. Wing, "Computational Thinking," *Communications of the ACM*, 49(3), 2006, pp. 33-35.

[6] George Pólya, *How to Solve It*, Princeton University Press, 1945.

[7] Tim Bell, Frances A. Rosamond, and Nancy Casey, "Computer Science Unplugged and Related Projects in Math and Computer Science Popularization," *The Multivariate Algorithmic Revolution and Beyond*, 2012, pp. 398-456.

[8] David Bau, Jeff Gray, Caitlin Kelleher, Josh Sheldon, and Lyn Turbak, "Learnable Programming: Blocks and Beyond," *Communications of the ACM*, 60(6), 2017, pp. 72-80.

[9] Kiki Prottsman, "Conditionals with Cards," YouTube Video, http://bit.ly/cond-cards

[10] Susan Loucks-Horsley, Katherine Stiles, Susan Mundry, Nancy Love, and Peter Hewson, *Designing Professional Development for Teachers of Science and Mathematics.* Corwin Press, 2010.

[11] Joanna Goode, Jane Margolis, and Gail Chapman, "Curriculum is not Enough: The Educational Theory and Research Foundation of the Exploring Computer Science Professional Development Model," In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education (SIGCSE)*, March 2014, Atlanta, GA, pp. 493-498.

[12] Outlier Research and Evaluation, *Evaluation of Code.org Computer Science Education Programs (2014-2015)*, http://outlier.uchicago.edu/evaluation_codeorg/

[13] Jane Margolis, Rachel Estrella, Joanna Goode, J., Jennifer Holme, and Kim Nao, *Stuck in the Shallow End: Education, Race, and Computing*, MIT Press, 2008.

[14] Wynne Harlen and Colin Holroyd, Primary teachers' understanding of concepts of science: Impact on confidence and teaching, *International Journal of Science Education*, vol. 19, no. 1, 1997, pp. 93-105.

[15] Carol Dwek, *Mindset: The New Psychology of Success*, Ballantine Books, 2007.