Coo/Think @JC
賽馬會運算思維教育
Inspiring digital creativity 啟發數碼創意

Evaluation Midline Report

July 2019

Created and Funded By

香港賽馬會慈善信託基金
The Hong Kong Jockey Club Charities Trust
同心同步同進 RIDING HIGH TOGETHER

Co-created By

香港教育大學
The Education University of Hong Kong

Massachusetts Institute of Technology

香港城市大學
City University of Hong Kong

Research Conducted By

SRI Education™
A DIVISION OF SRI INTERNATIONAL

Authors

Linda Shear, Haiwen Wang, Yuning Xu, Carol Tate, Satabdi Basu, Daisy Rutstein

SRI Education

Suggested Citation

Shear, L., Wang, H., Xu, Y., Tate, C., Basu, S., Rutstein, D. (July, 2019). *CoolThink@JC Evaluation: Midline Report*. Menlo Park, CA: SRI International.

CoolThink@JC is created and funded by The Hong Kong Jockey Club Charities Trust, and co-created by The Education University of Hong Kong, Massachusetts Institute of Technology, and City University of Hong Kong.

Table of Contents

Executive Summary

The *CoolThink@JC* pilot[1] is, for the first time, bringing computational thinking classes to Primary 4-6 students across 32 schools in Hong Kong. While computational thinking is quickly growing as an educational priority in many countries, few have yet begun to introduce these important concepts and skills to primary-age students on this large a scale. This is the second report in a series from a rigorous evaluation by SRI International of the pilot phase of *CoolThink@JC*, and the first to offer a view of emerging outcomes from this important instructional initiative.

CoolThink@JC offers three levels of curriculum units[2] to participating schools, for a 3-year program of instruction, along with extensive professional development for participating teachers and a graduate student teaching assistant (TA) in each class to provide technical support. The curriculum lessons are built on the Scratch and MIT App Inventor programming languages. Following the framework in Appendix A, the curriculum seeks to promote three dimensions of computational thinking (CT):

- *CT Concepts:* Content knowledge related to computational thinking
- *CT Practices:* Problem-solving and logical thinking skills
- *CT Perspectives:* Interest in and motivation for computational thinking

The evaluation project that produced this report has two main components: an outcome study and an implementation study. The outcome study uses a set of rigorously-designed assessments to measure CT Concepts and CT Practices along with a student survey to measure CT Perspectives, and employs sophisticated analytic techniques to estimate the impact of *CoolThink@JC* lessons for students in 30 pilot schools[3] compared to peers in similar schools that do not participate in the pilot. The implementation study uses a teacher survey (in all 30 participating schools) and classroom observations, interviews, and focus groups (in four schools) to explore how and under what conditions those outcomes are achieved and to document the experience of *CoolThink@JC* teachers and students.

Results described in this midline report are based on data from two years of instruction in ten "cohort 1" schools which began Level 1 instruction in the 2016-17 school year, and the first year of instruction in twenty "cohort 2" schools which began in 2017-18. This provides a strong sample of over 10,500 students to look at outcomes after one year of *CoolThink@JC* lessons, while outcomes after two years are more exploratory and will be confirmed after another year of implementation.

1 Created by the Hong Kong Jockey Club Charities Trust; co-created by Education University of Hong Kong, Massachusetts Institute of Technology, and City University of Hong Kong.

2 Each level, taught in a given instructional year, includes 9 or 14 hours of planned lessons.

3 The evaluation does not include two resource schools that partnered with developers to trial the lessons prior to the larger pilot rollout.

CoolThink@JC Adoption

Based on teacher surveys and observations/interviews at four schools, the implementation study seeks to describe what *CoolThink@JC* looks like in participating classrooms and how it is experienced by teachers and students. These site visits took place in the 2017-18 school year, so the results of ongoing lesson refinement in preparation for 2018-19 were not yet in place. Key findings include:

- *The pedagogical style of CoolThink@JC* teachers varied substantially, ranging from formal to student-centered. By the end of the year the majority of teachers were making use of at least some student-centered approaches, such as facilitating guided exploration and groupwork. Some teachers described this as different from their typical teaching methods.
- *Time constraints and project complexity made it difficult to complete lessons as intended in some cases.* This was particularly true with younger students: an age group that few other computational thinking initiatives have yet reached, so approaches are still exploratory. Adaptations made by teachers as a result included simplifying tasks, using pre-coded templates, and in at least one case integrating *CoolThink@JC* lessons with assignments in other classes.
- *Teachers perceive the Teacher Development Course and associated supports to be helpful for understanding the lesson content and rationale for teaching computational thinking.* The initiative includes two 39-hour professional development courses for participating teachers, which were seen as essential enablers as teachers took up their new roles in computational thinking instruction. The participation of teaching assistants was also important to some teachers, particularly those who were still developing their confidence in the subject matter.

- *CoolThink@JC implementation offers an opportunity for professional growth that is appreciated by both teachers and principals.* Participating adults described the program as requiring extra time for preparation but also providing an opportunity for teachers to engage in new teaching practices and learn together with colleagues.

CoolThink@JC Student Outcomes

Results reported here are based on pre- and post-measures of student computational thinking knowledge, skills and beliefs after one year of instruction in 30 pilot schools, in contrast to computational thinking learning over a similar time period by students in similar schools that did not participate in *CoolThink@JC*. The full report also reflects on preliminary findings after two years of instruction. In addition, we report on relative impact[4] from *CoolThink@JC* lessons for different sub-populations of students, to see which groups of students benefit most from the lessons as they are currently implemented. Key findings include:

For CT Concepts:

- *Students in pilot schools exhibited more learning in CT Concepts than similar students in comparison schools.* This difference remained true after controlling for student, classroom and school characteristics. The difference is statistically significant, and is a notable result in the pilot phase of any initiative.
- *Boys benefited more from the pilot lessons than did girls.* Some teachers noticed that boys and girls tended to engage with the lessons in different ways, with girls more intent on following procedures and boys more likely to express their ideas and explore alternative approaches.
- *Other students who benefited more than their peers include students who showed some knowledge of CT Concepts at baseline, who had higher baseline math test scores relative to other students in their school, and students in classes with fewer special education needs (SEN) students.* These results suggest that more advanced students generally were able to engage more successfully with the version of pilot lessons that were in schools in the 2017-18 school year.

For CT Practices:

- *Students in pilot schools exhibited more learning in CT Practices than similar students in comparison schools.* These results may be somewhat compromised by the fact that many students in both pilot and comparison schools rushed through the CT Practices post-test at the end of the school year and did not give the test as much attention as they did for the pre-test early in the year, an administration issue that SRI, with the support of initiative partners, is working hard to improve in the 2019 data collection.
- *Student final projects scored by teachers in 2017-18 received approximately 76% passing marks in Level 1 and 73% passing in Level 2.* This analysis was completed by the Education University of Hong Kong, and represents another view of student learning of computational thinking practices.

4 In this report, the words "benefit" or "impact" refer to differences in learning between students in pilot and comparison schools, using statistical methods to adjust for differences in baseline performance, demographics, and other salient characteristics at the student, classroom or school levels.

For CT Perspectives:

- *Students in pilot schools exhibited greater positive changes on all CT Perspectives sub-constructs than similar students in comparison schools; however, these differences are not statistically significant.* Measured changes in student perspectives on the value of, and interest in, computational thinking were small after one year of instruction in all areas but one. Significant differences were displayed for ***self-efficacy***, which suggests that students thought they developed knowledge and skills over the course of the year.
- *Boys showed greater positive change in* ***creativity*** *and* ***self-efficacy*** *subconstructs than girls participating in the pilot, whereas girls showed greater positive change in the* ***belonging*** *subconstruct than boys.* Belonging is a measure of students' inclination to work in groups as they code.

Scaling *CoolThink@JC*

This research, combining a rigorous outcome study with a rich understanding of implementation and challenges for teachers, results in a number of recommendations for developers and stakeholders as they consider opportunities for scaling the initiative:

- *A central focus of planning for scale should be opportunities for teachers to continue to receive substantive professional learning opportunities,* which are seen as essential to the success of the initiative. The extended duration (nearly two full weeks of instruction, delivered over time) and resulting depth of the professional learning offered through *CoolThink@JC* was critical to its effectiveness, so it will be important not to sacrifice these attributes as the initiative goes to scale. Enabling methods might include train-the-trainer models and explicit facilitation of communities of practice for teachers. As the network of professional learning facilitators expands, continued monitoring and central support are essential to ensure adherence of offerings to the goals of the initiative and to leverage positive enhancements.
- *Professional learning resources should include guidance for teachers in how to adapt CoolThink@JC lessons for different students and settings.* Pilot teachers sometimes struggled with ways to manage classroom diversity and reach lower-skilled students and students with differing orientations to learning such as girls vs. boys, as well as how to work within practical constraints of available time. It will be important for professional learning coordinators to offer strategies for working with students and for adaptation of materials so that teachers can be better able to customize the lessons for the needs of their own classroom while maintaining alignment with program goals, such as a balance between structure and freedom to explore.
- *Provide differential supports for schools and teachers with different levels of capacity.* For example, some pilot teachers found teaching assistants to be a necessary aid as they adjusted to their new teaching roles, and some classrooms and schools were better equipped technically to support *CoolThink@JC* lessons. With scale comes the need to support schools and teachers with a wider range of capacity and readiness for *CoolThink@JC*; this increased diversity must be attended to in planning for growth.
- *Consider issues of systemic support for computational thinking instruction.* Computational

thinking may become a new area of academics in the Hong Kong curriculum. Pilot schools necessarily gave primacy to more established subjects in the curriculum, and could typically only allocate 9 or 14 hours per year to computational thinking instruction. It will be important for education leaders to consider ways to make time available in the school day. STEM+C (the integration of computational thinking with other school subjects) is one approach that some countries are beginning to explore, although it would require an investment in curriculum development and research to promote effectively. Events that engage parents can also be important to build community support for adding computational thinking to the school schedule.

This report demonstrates the early promise of the *CoolThink@JC* initiative, while continuing to inform the ongoing improvement of the lessons that is a primary goal of this pilot. The final report in this series, to be issued in early 2020, will reflect on the results of the multi-year trajectory of learning offered by this three-year computational thinking lesson series.

Introduction: A Growing Global Trend

Computational Thinking (CT) is increasingly being recognized globally as an essential "5th C" of 21st century skills—in addition to critical thinking, creativity, collaboration and communication—that ought to be taught to all students to help them succeed in today's world of pervasive computing in all spheres. CT refers to the thought processes and strategies involved in understanding, formulating and solving a problem in such a way that a computer can potentially carry out the solution (Wing, 2006).

Education systems around the world are recognizing the importance of students being able to problem-solve computationally and work with and create computational artifacts like models, simulations and data visualizations. While these initiatives are most prevalent in secondary school (Bocconi et al., 2016; Yadav, Good, Voogt & Fisser, 2017), nations are increasingly recognizing that students need to learn CT skills from an early age so that they can become creators, not just consumers, of the next wave of computing innovations in today's digital world. As a result, in some countries CT is beginning to emerge as part of curricular standards beginning in primary school (UK Department for Education, 2013; Bocconi et al., 2016). Hong Kong's pilot *CoolThink@JC* initiative, providing instruction in programming and CT to students in primary grades, is a part of this important global trend.

Though CT routinely features in various national standards, studies of CT skills as outcomes are limited relative to the growth of CT programs internationally. Those studies that do exist so far have generally not examined, in a rigorous manner and at scale, CT skills as outcomes in primary grades. While programming is an example of a useful and commonly used vehicle to foster CT, CT learning outcomes are much more than programming learning outcomes. Computational thinking includes the aspects of problem analysis and formulation before generating a program, as well as the use of a program to computationally problem-solve. Hence, merely assessing important programming constructs (also known as 'CT Concepts') is not a sufficient measure of CT skills, though it is the most commonly used measure.

To address this need, SRI International (SRI) created a system of assessments as part of its evaluation study of the *CoolThink@JC* pilot. These include instruments to measure CT Concepts (knowledge of computational constructs used across programming languages), CT Practices (the process of planning and efficiently engaging with the computational constructs), and CT Perspectives (attitudes towards the CT discipline and confidence in ability to create computational artifacts). Supported by these assessments, as well as the creation of student-generated computational artifacts within the program, the *CoolThink@JC* pilot presents a unique opportunity to study computational thinking for primary students at scale. An important contribution of the research is the innovative methods it uses to measure the hard-to-assess computational thinking skills that Hong Kong recognizes as a key to its future economic prosperity.

This midline report is the second in a series of reports from SRI's study of the pilot phase of Hong

Kong's *CoolThink@JC* initiative. Two years into the *CoolThink@JC* pilot initiative, this is the first opportunity to look at its emerging results. The report summarizes lesson implementation and student outcomes for *CoolThink@JC* at an early stage in the pilot program: teachers and students have been using the lessons for one year in 20 schools, and for two years in another 10 schools. Analysis of first-year outcomes uses data from all 30 pilot schools, offering a comprehensive view of student progress in the first year of the intervention. From the smaller sample of 10 schools that have been using *CoolThink@JC* lessons for two years, this report also offers a tentative preview of student progress over time that will be confirmed in future reporting.

This report will:

- Introduce the *CoolThink@JC* initiative and the methods used in this evaluation research
- Introduce the schools participating in the *CoolThink@JC* pilot
- Describe the *CoolThink@JC* classroom, and the experience of teachers and students
- Present outcomes for CT Concepts, CT Practices, and CT Perspectives
- Provide guidance based on this research for stakeholders who are considering bringing *CoolThink@JC* to scale in Hong Kong
- Summarize the contributions of the program and this research.

CoolThink@JC: Inspiring Digital Creativity

The *CoolThink@JC* initiative is a 4-year pilot program to teach computer programming and computational thinking to upper primary students in Hong Kong. The program is created by the Hong Kong Jockey Club Charities Trust (HKJC), and co-created by Education University of Hong Kong (EdUHK), Massachusetts Institute of Technology (MIT), and City University of Hong Kong (CityU).

The *CoolThink@JC* lessons are based on the visual programming languages Scratch and MIT App Inventor. There are three levels of instruction targeted at primary grades 4, 5, and 6, with each level building off of concepts and practices introduced in previous levels. The lesson sequence was developed in two versions for each level, with a 9-hour version providing the base content and a 14-hour version with extra lessons that provide opportunities for more open-ended exploration of the concepts and practices included in the 9-hour version.

CoolThink@JC also includes a host of supports for teachers. Participating teachers were offered a series of two teacher development courses, led by EdUHK and MIT, each providing 39 hours of training. A week-long workshop introduced the *CoolThink@JC* lessons and coding with Scratch and MIT App Inventor, and discussed issues of pedagogy. A second teacher development course took the form of a series of 13 3-hour lessons, approximately one per week, to provide sustained support for reflection on instruction, consideration of pedagogical approaches, and the collaborative development of a unit of instruction. In addition, one or two graduate student teaching leads (TLs) or teaching assistants (TAs) were assigned to each *CoolThink@JC* class to provide technical support and aid students. Participating schools were also given subsidies to seed the purchase or renovation of classroom equipment in order to run co-curricular activities outside of regular class hours.

The *CoolThink@JC* initiative was launched beginning in 2016, in 32 Hong Kong schools that had been successful in a competitive application process. The pilot consists of 30 schools in two cohorts: the 10 schools in Cohort 1 began teaching *CoolThink@JC* lessons in the 2016-17 school year, and another 20 schools in Cohort 2 joined in 2017-18. In addition, two resource schools are participating in lesson development and initial trials, as well as teaching the lessons to their own students.

CoolThink@JC is designed to achieve the following outcomes:

1. Increase students' content knowledge related to computational thinking (CT Concepts),
2. Improve student's problem-solving/logical thinking skills (CT Practices), and
3. Generate interest and motivation for computational thinking (CT Perspectives).

The lesson sequence is grounded in a detailed framework (shown in Appendix A), based on the work of Brennan and Resnick (2012), that elaborates on each of these categories of knowledge and skills.

This design framework also provides the grounding for the outcome study described in this report.

At the current stage of the initiative, early in calendar year 2019, resource schools and schools that began in Cohort 1 have begun their third year of teaching the *CoolThink@JC* lessons. In addition to supporting this instruction, the development partners and this evaluation project have been collecting data on progress and challenges to date. In response to these inputs, the development team is adapting the lessons, using the pilot as an opportunity to finetune their designs. In particular, a new version of the lessons was developed in summer 2018 to streamline the contents to make it better fit available instructional time, among other changes. This evaluation report is based on the version of *CoolThink@JC* that was in schools in the 2017-18 school year, prior to the implementation of these enhancements.

Overview of the Study

The *CoolThink@JC* program evaluation has two important and complementary components. It includes an outcome study to measure the impact of the lessons on target computational thinking outcomes, and an implementation study to look at how the lessons are being enacted in classrooms. This combination of methods is intended to inform stakeholders' decisions about whether *CoolThink@JC* is a strong candidate for introduction to a greater number of schools within Hong Kong, and how the program should be enacted and supported for best success at scale.

The outcome study is designed to measure student outcomes and progression over time in each of the primary goal areas of the *CoolThink@JC* lessons: CT Concepts, Practices, and Perspectives. The outcome study answers the following research questions:

1. What is the impact of the lessons on students' computational thinking concepts, practices, and perspectives?
2. How do gender, grade level, dosage, baseline performance, and other factors impact primary outcomes?
3. How does students' progression of CT learning vary across levels of the lesson sequence?

The outcome study uses a comparison design, looking at the differences in outcomes between students who are participating in the *CoolThink@JC* pilot and their peers in schools that are not enrolled in the initiative. The study includes a total of 30 schools participating in *CoolThink@JC* across the two cohorts, and 24 matched comparison schools. The research methods section below will describe the instruments we designed to assess hard-to-measure computational thinking skills, the selection of comparison schools, and the advanced analytic techniques we use to achieve the intended comparisons.

While the outcome study is focused on measuring what students are learning, the implementation study seeks to describe how this learning is taking place in classrooms. By observing *CoolThink@JC* lessons and asking educators and students about their experiences with the program, the implementation study seeks to identify supports and barriers to adoption. In addition to observations, focus groups and interviews at four selected schools, the implementation study also employs a survey of all *CoolThink@JC* teachers across the 30 pilot schools for a broader understanding of teacher experiences. The implementation study investigates the following research questions:

1. To what extent are the *CoolThink@JC* lessons implemented as intended?
2. In what ways do the enacted lessons deviate from the expected models of implementation?
3. What supports and barriers do teachers encounter as they take on *CoolThink@JC*?
4. What implementation factors appear to be associated with success?

Research Design and Methods

The design of this research is noteworthy for its capture of the hard-to-assess constructs of CT Concepts, CT Practices, and CT Perspectives, for the state-of-the-art methods it uses to analyze the impact of the pilot initiative with as much rigor as possible, and for its implementation study to provide a well-rounded picture of the contexts in which students are learning. We begin with a description of assessment design.

Assessment design: Measuring computational thinking

The *CoolThink@JC* evaluation team at SRI developed three primary instruments for measuring computational thinking outcomes: a CT Concepts assessment, a CT Practices assessment, and a survey for measuring students' CT Perspectives.

The two assessments were developed using an Evidence-Centered Design approach (ECD), in order to confirm alignment between the learning goals of the *CoolThink@JC* framework and the ultimate items in the assessment. ECD is a process of design and iterative review that defines assessment goals in terms of the focal knowledge, skills and abilities (FKSAs) that we want to measure, and uses those as a basis for the development of assessment items. For more information on ECD please see Mislevy, 2007; Mislevy & Haertel, 2006; and Mislevy & Riconscente, 2006.

Another important method used in the design of these assessments is *partial matrix sampling* (see Appendix B for a detailed description of this method). The *CoolThink@JC* framework includes a large number of constructs; assessing each student on each of these would result in a heavy testing burden for students. Partial matrix sampling allows a complete set of items to be distributed across multiple forms, including some items that are common to all forms. This format reduces the time students must each spend on the test and allows scores to be compared for individual students, while still allowing measurement of the full set of constructs across the population.

Partial matrix sampling: Distribution of a complete set of test items across multiple forms, with questions that are common across each form, so individual students only have to answer a subset of the questions.

Computational Thinking Concepts

The CT Concepts suite of assessments was designed to measure students' progression in their knowledge of five computational thinking concepts that are core to the *CoolThink@JC* lessons:

- *Repetition:* Running the same sequence multiple times
- *Conditionals:* Making decisions (or branching) based on conditions
- *Parallelism and Sequencing:* Identifying a series of steps for a task and knowledge of how to make two things happen at the same time
- *Data Structures:* The basic ways data are formatted and stored
- *Procedures:* Separating out parts of the code for ease of use or reusability

In order to support automatic scoring of the assessments, the CT Concepts assessment items all use a multiple-choice format: each item contains a stem representing a computational context and three to four response options. The incorrect response options were developed based on common misconceptions that students might have. The research team piloted the items with students and selected/revised items based on results. The resulting set of items were distributed across three forms, one corresponding to each level of the lesson sequence. Item statistics and response distributions were used to determine which items were placed on which forms, to include a range of difficulty and coverage of the concepts on each form. Each form contains a core set of items that are the same across all of the forms and cover content from all 3 levels, with the majority of items focused on the content taught at the corresponding level. This format allows us to generate comparable scores to enable tracking of student progress over time.

The CT Concepts assessment uses partial matrix sampling in order to cover all required constructs without an overly long assessment. At baseline, this resulted in multiple forms of the CT Concepts Level 1 assessment that were distributed across students. Based on the results of baseline analysis and the evolution of the *CoolThink@JC* design (including the removal of the focus on the Procedures concept), the test was streamlined for midline administration so that it now consists of a single form for each level that covers all necessary concepts with fewer required items.

Computational Thinking Practices

The CT Practices assessment measures students' understanding of the four computational thinking practices that receive the strongest emphasis in the *CoolThink@JC* lessons:

- *Algorithmic Thinking:* Articulating a problem solution in well-defined rules and steps
- *Reusing and Remixing:* Making something by building on existing projects or ideas
- *Testing and Debugging:* Identifying and solving problems and errors in the problem solution when they arise
- *Abstracting and Modularizing:* Identifying patterns and exploring connections between the whole and the parts

The CT practices assessment items are a mix of multiple choice, multiple selection (multiple choice with more than one correct response option) and drag and drop format. These formats allow for the responses to be automatically scored, but also allow more accurate measurement than solely multiple-choice items would, allowing students to demonstrate CT Practices through more active engagement with the question. Each question presents a scenario that the student must address. Using a partial matrix sampling approach, CT Practices questions are distributed across three forms with some items common to all forms. The assignment of forms is randomized to students within each class to ensure adequate coverage of all items.

In contrast to the CT Concepts assessment, it was not necessary to develop separate versions of the assessment for each of the levels of the *CoolThink* lesson sequence. Instead, a single assessment serves as a pretest as well as a posttest for each of the levels.

Computational Thinking Perspectives

The CT Perspectives survey measures students' interest and motivation for computational thinking, and their perceptions of its nature and utility. Component constructs were identified through a review of literature, with Brennan and Resnick (2012) as a foundation, followed by piloting, validation, and iterative rounds of review and discussions with the project implementation partners. The constructs are:

- *Interest in programming:* Interest in programming and in thinking computationally
- *Digital self-efficacy/competence:* Confidence in ability to program and think computationally
- *Utility motivation:* Perception that computational thinking is useful, and motivation to pursue it
- *Meaningfulness/motivation to help the world:* Motivation to use computation to solve problems and benefit the world
- *Creativity:* Perception of programming as a creative endeavor
- *Engagement:* Attainment of "state of flow" level of focus when thinking computationally, including persistence in the face of programming challenges
- *Belonging:* Recognition of computing as a collaborative endeavor

The final version of the survey consists of three to five 5-point Likert-type items per construct as well as several questions about students' background information, such as prior coding experience and internet connectivity at home. In the baseline administration we distributed the items across two forms such that students randomly received items that cover three or four of the seven sub-constructs. Based on results from early administration and data on the time students required to complete the forms, all items were subsequently combined onto a single form which was provided to all students in the midline administration.

Outcome study

This research measures computational thinking outcomes for students in *CoolThink@JC* pilot schools, and compares them with their counterparts in comparison schools in order to isolate effects of the program from other factors that may affect student outcomes. If we only measured how much students learned with *CoolThink@JC*, we would not know how similar or different that is from what they might have learned without the program. The comparison design helps to answer the question: **How much more** do students learn from *CoolThink@JC*?

Outcome study data collection is taking place over a series of three years (see Table 1). For CT Concepts and Perspectives, baseline data collection was in February 2017—at the start of Level 1 instruction in Cohort 1 schools—and outcome data are being collected at the end of the 2016-17, 2017-18, and 2018-19 school years. For CT Practices, baseline data were collected at the beginning of the 2017-18 school year—at the start of Level 1 instruction in Cohort 2 schools—and outcome data are being collected at the end of the 2017-18 and 2018-19 school years.

In order to provide a fair comparison, careful selection of comparison schools is essential. Random assignment of participating schools to pilot and comparison groups was not possible, so instead we selected comparison schools that were as close as possible to pilot schools on important observable variables. The goal is to maximize the likelihood

Cohorts of schools participating in the *CoolThink@JC* pilot

- **Cohort 1:** 10 schools that began with *CoolThink@JC* in the 2016-17 school year. At the time of the midline data collection, teachers in Cohort 1 schools had been teaching *CoolThink@JC* for 2 years.
- **Cohort 2:** 20 schools that began with *CoolThink@JC* in the 2017-18 school year. At the time of the midline data collection, teachers in Cohort 2 schools had been teaching *CoolThink@JC* for 1 year.

Table 1. Outcome Study Data Collection.

Instrument	Timing	Sample
CT Concepts	February-March 2017; May-July 2017, 2018; April-July 2019	All students in Primary 4-6 at 30 pilot and 24 comparison schools*
CT Perspectives		
CT Practices	September-October 2017; May-July 2018; April-July 2019	

** In addition, students in Primary 3 were tested in the spring of 2017 and 2018 to serve as baseline for the following year*

that any differences in outcomes between pilot and comparison students would be due to *CoolThink@JC* and its implementation, rather than to prior differences between pilot and comparison schools.

Comparison schools were selected from schools that had applied for the program but were not selected. We were able to match the 30 pilot schools participating in *CoolThink@JC* with 24 comparison schools, based on the experience these schools had with prior coding instruction, percent of students using financial aid, percent of students with special education needs, as well as a "paper vetting score." Paper vetting score was assigned by the *CoolThink@JC* team during the selection process to capture the motivation or willingness of the schools to engage in different aspects of the *CoolThink@JC* program based on the applications they submitted to enter the program.

The outcome study compares students' performance on CT Concepts, Practices, and Perspectives assessments over time and between pilot and comparison schools. We also conducted analyses to understand whether the students in different subgroups are more or less successful in *CoolThink@JC* based on demographic and other factors, such as gender, special education status, economic status, family background, grade level, baseline achievement, and baseline computational skills and experiences.

In school-based research, an important decision point is level of analysis: the school, classroom, or student. Because the intervention is conducted at the school level (the whole school adopts *CoolThink@JC* or not), and because school-level characteristics tend to be an important source of variation in students' educational outcomes, we apply in our analyses *hierarchical linear models (HLM)* that account for nesting of students within schools. Because students stayed in the same classroom during the first year of the pilot program, we also include a classroom level for year 1 impact analysis to adjust for classroom effect. The analyses also control for differences in student, classroom and school background characteristics and student baseline scores on the *CoolThink@JC* research assessments. Please see Appendix B for detailed descriptions of our analytic methods.

Because students in pilot schools take different levels of the lessons in sequence (first level 1, then level 2, and in some cases finally level 3), we estimate the cumulative impact as students proceed through the three levels. As of 2017-18, when the data in this report were collected, students in all participating schools (cohorts 1 and 2) had taken the first year of the pilot lessons. However, a much smaller number (only some students in the 10 Cohort 1 schools) had experienced two years of lessons so far. Therefore, this report focuses primarily on year 1 results. The year 2 impact results presented in this report are preliminary, and will be confirmed in the coming year when more students have participated in Level 2.

Students' CT Concepts, Perspectives and Practices scores are calculated based on *Item Response Theory (IRT)*. IRT is a modeling method that is commonly used with important large-scale standardized assessments (e.g., Trends in International Mathematics and Science Study (TIMMS), Program for International Student Assessment (PISA), and US National Assessment for Educational Progress (NAEP)). In IRT, scores on assessment items are used to place student ability and item difficulty on a single continuum of a given construct. This method of analysis allows us to create an overall estimate of computational thinking ability, and to look at the progression of an individual student or cohort of students along that computational thinking continuum. IRT is tuned to handle missing data, which is important in partial matrix sampling because individual students will each only respond to a subset of the total pool of test items or constructs on a given assessment. IRT takes item characteristics such as difficulty into account in student ability score estimation. As a result, learning gains can be tracked across forms and over multiple years using a subset of common items, despite the fact that students receive different items within and across administrations.

Implementation study

The implementation study is designed to enhance stakeholders' understanding of impact study results by illuminating how differences related to school contexts, teachers, and students may explain differences in student outcomes. Research in implementation science, in education as well as other sectors from healthcare to manufacturing, tells us that not all recipients of the same intervention will benefit in the same way (Kelly and Perkins, 2012). Many factors can influence both the way in which a program is enacted and the way in which it is received by the target group (in this case, primary students). In a curricular intervention such as *CoolThink@JC* many variables exist at every level—school, teacher, and student—which might influence implementation and thus student outcomes. For example, school variables range from the way furniture is arranged in the classroom to the professional culture among teachers and school leaders. At the teacher level, such things as teacher experience, primary subject area, and pedagogical commitments are important influences on instruction.

The implementation study cannot provide a definitive answer to why a program works or doesn't. Instead, it offers a picture of what the program looks like in the real world, and how closely that picture resembles the designers' intentions. It also offers possible explanations for variations in program outcomes, and insights into how the program may be affecting individual participants, or groups of participants in different ways.

The implementation study uses primarily qualitative methods (classroom observations and interviews/focus groups) to look deeply into the experience of teachers and students in four schools. Instruments for data collection were designed by SRI, in consultation with project partners, and data were collected by Ipsos, a global research organization with an office in Hong Kong. In addition to this in-depth inquiry at the four focal schools, an educator survey of all pilot teachers, designed and administered by Ipsos in consultation with SRI, provides a broader view of teacher characteristics such as years of experience and primary subject area, as well as allowing us to track their attitudes and perceptions of *CoolThink@JC*. The survey also gathers information about adaptations teachers are making to the lessons, including how, why, and how much teachers are modifying the materials and activities.

The implementation study data will be collected in three waves, as outlined in Table 2. This midline report is based on analysis of data from early and late in the 2017-18 school year, or the first two waves of data collection.

The sample of case study focal schools is designed to be representative of the wide range of Hong Kong primary school contexts. In choosing schools, we sought variation in funding sources, student backgrounds, medium of instruction (Chinese and English), and religious affiliation. Two of our four focal schools have an affiliated secondary school.

The study design outlined above seeks to gather actionable information that can inform project partners' efforts to support adoption and scaling of *CoolThink@JC* in a variety of settings. We looked for features of the instructional settings that appear to facilitate or hinder lesson enactment, and how different implementation profiles may be associated with stronger student learning outcomes. The goals of the implementation study are to discern potential pitfalls as well as available levers to support successful implementation as the *CoolThink@JC* program scales to more Hong Kong schools.

Table 2. Implementation Study Data Collection.

<table>
<tr><th>Method</th><th>Sample</th><th>Timing</th></tr>
<tr><td>Classroom Observations</td><td>12 (3 per school)</td><td rowspan="5">Wave 1: November-December 2017
Wave 2: May-July 2018
Wave 3: March-June 2019</td></tr>
<tr><td>Teacher Interviews</td><td>12 (3 per school)</td></tr>
<tr><td>Student Focus Groups</td><td>8 (2 per school)</td></tr>
<tr><td>Principal Interviews</td><td>4 (1 per school)</td></tr>
<tr><td>Educator Survey</td><td>All CoolThink@JC teachers</td></tr>
</table>

About the *CoolThink@JC* Pilot Schools

30 schools and more than 10,500 students in Primary 4-6 are participating in the *CoolThink@JC* pilot. The schools selected for the pilot represent a broad cross-section of the school types in Hong Kong. Characteristics of these schools as of the beginning of the pilot are summarized in Table 3. The majority of pilot schools are aided schools that conduct instruction in Chinese. Half of them are located in New Territories. The schools serve students from a variety of religious backgrounds. On average, the pilot schools enroll 12% students with special needs and 38% students with financial aid. Pilot school enrollments are similar across Primary 4-6.

As summarized in Table 4, the Educator Survey revealed that the majority of teachers in the pilot schools are quite experienced, with an average of 12 years of teaching experience, and 7 years of experience teaching ICT/computing. More than half (59%) of them have taught over 10 years and 38% percent have taught ICT/computing for 10 years or more. However, the majority of teachers trained by the *CoolThink@JC* initiative received their pre-service training in a subject other than Information Technology (IT) or related subjects. According to data collected by the initiative, only 33% of those 112 teachers had previously obtained a bachelor degree or above majoring in IT or related subjects. Almost three quarters (74%) of teachers report teaching classes of less than 30 students. Forty-six percent of survey respondents say they teach *CoolThink@JC* in a single-lesson format (35 minutes), while 54% teach a double lesson (70 minutes).

To illustrate the diversity of schools in the pilot, the four schools visited for the implementation study are described in the box.

Table 3. *CoolThink@JC* Pilot School Enrollment by Grade and School Characteristics.

	Number of Schools	Student Enrollment	% Students
Total	**30**	**10,513**	**100%**
By grade			
Primary 4	30	3,418	33%
Primary 5	30	3,475	33%
Primary 6	30	3,620	34%
By school type			
Government	2	802	8%
Aided	26	8,894	85%
Direct subsidy scheme	2	817	8%
By region			
Hong Kong Island	4	1,243	12%
Kowloon	11	4,133	39%
New Territories	15	5,137	49%
By religious affiliation			
No affiliation	11	3,707	35%
Catholicism	9	3,212	31%
Christianity, Non-Catholic	7	2,759	26%
Other	3	835	8%
By instructional language			
Chinese instruction	26	9,160	87%
English instruction	4	1,353	13%

Source: 2016-17 school rosters and CoolThink@JC applications.

Table 4. Characteristics of Pilot Teachers.

Teacher Characteristics	Average
Average Years of Teaching	12
% Teaching Longer Than 10 Years	59%
Average Years of Teaching ICT	7
% With a Bachelor Degree or Above in IT or Related Subject	33%

Source: Educator survey, 2018; 119 teachers (78% response rate)

School One is a traditional Catholic girls' school using English as the medium of instruction. This school is well-known for its academic excellence as well as its focus on well-rounded personal development. Music, art, sports and other extra-curricular activities are given importance. The student body is quite homogenous in terms of family background (coming from more well-off families) and academic ability. The parents are mostly well-educated and are highly engaged in school activities and developments. School One has an affiliated secondary school.

School Two is an established neighborhood-type school using Chinese as the medium of instruction. The student body is heterogenous in terms of social background and academic ability. There are SEN (Special Educational Needs) students, South Asians and Chinese Immigrants, and students who reside in mainland China and cross the border to attend school. Classes include approximately 25% non-native Cantonese-speaking students.

School Three is a traditional Catholic School using Chinese as the medium of instruction. Students are from low-income families living nearby (mostly in public housing). The school supports the development of the whole student with afternoon sessions reserved for less academically intense classes such as physical education, music, and computers. This school has less homework when compared with most subsidy schools in the same district. The school relies heavily on government subsidy for its resources; it has less flexibility to allocate additional funding to computer studies.

School Four is a Direct Subsidy School (DSS) using English as the medium of instruction. As a DSS, the school serves tuition-paying families but also receives funding from the government. The school has high flexibility in using this funding as it sees fit. The school has 100% control in the selection of its students, who typically come from middle class families. This school uses an activity-based and subject-specified teaching approach across the curriculum, in which teachers are only responsible for teaching one specialized subject and they are allowed to design their own curriculum. Like School One, this school has an affiliated secondary school.

Characteristics of pilot and comparison schools

In order to estimate whether students learned more about computational thinking with *CoolThink@JC* than they might have if they had not participated in the pilot, we identified a set of comparison schools whose school and student characteristics are as similar as possible to the pilot schools and students. As described earlier, the sampling process attempted to find close matches between pilot and comparison schools on available school-level variables, but it was hindered by the purposeful selection of pilot schools based on some of these same characteristics. In particular, because paper vetting score[5] was a primary selection criterion for the pilot schools, on average this score is predictably higher for pilot schools than for the comparison schools that were not selected into the pilot initiative (see Table 5), which may mean that as a set they have somewhat more capacity for teaching *CoolThink@JC* than their comparisons. Other differences between the pilot and comparison groups are not significant, although somewhat more students in comparison schools receive financial aid than in pilot schools. These differences between the two groups of schools are taken into account in analyses through statistical controls.

We now turn to a description of the implementation and outcomes of the *CoolThink@JC* initiative, as of the end of its second year in Cohort 1 schools and first year in Cohort 2 schools. Before we describe student outcomes, we offer a picture of what teaching and learning looks like in *CoolThink@JC* classrooms and the experiences of educators and students.

Table 5. Characteristics of *CoolThink@JC* Pilot and Comparison Schools.

Teacher Characteristics	Pilot School Average	Comparison School Average
School Characteristics	(n=30)	(n=24)
% Special Needs	12.0	13.6
% Financial Aid	37.5	44.0
Prior Coding Experience Score	4.4	4.0
Total Paper Vetting Score	29.7	22.5

5 Paper vetting score is a measure of a school's expected capacity for *CoolThink@JC*, based on information collected during the application process.

CoolThink@JC Adoption

As mentioned in the research design section, we do implementation research on a particular intervention in order to understand the contexts of implementation—the people involved, who is benefitting from the intervention, who isn't, and what kinds of variations in fidelity of implementation exist. This helps us interpret the outcomes we see in the impact study. When we analyze our data, we draw on research to guide our analysis. While the focus on teaching computational thinking in primary grades is relatively new, and the research base is still developing, research on the science of implementation in education (e.g. Fixsen, et al., 2010), and the scholarship of school change more generally (e.g. Elmore, 1996; City et al., 2009; Fullan, 2012), informs our inquiry.

David Cohen and Deborah Ball (2007), who did extensive research on the implementation of standards-based math reform in the United States, have pointed out that, while the design of the curriculum is extremely important, it is also essential to focus on how lessons are understood and experienced by teachers and students. This relationship between curriculum, students, and teachers is what Harvard education scholar Richard Elmore (1996) calls the 'instructional core'. In Elmore's view, it is the failure of good educational ideas to penetrate this core that makes it very difficult for reforms to scale. Success depends on three things: high-level content, strong teacher knowledge, and student engagement. Intervening on one aspect of this core without intervening on the others, Elmore argues, will not yield positive results. In other words, no matter how strong the design of an instructional intervention, it is only what students are actually doing in the classroom that will make the difference. This explains how two teachers working with the same materials can have very different student outcomes. It is with these principles in mind that we look closely at the *CoolThink@JC* program in action.

It is important to note that the surveys and observations that inform this section took place late in the 2017-18 school year. Some challenges described here, such as a lack of sufficient time to complete all content, have since been addressed with significant revisions made by the *CoolThink@JC* development team. The responses discussed here do not reflect teachers' experiences with the 2018-19 version of *CoolThink@JC* teaching materials.

What does *CoolThink@JC* instruction look like?

The instrument we used to guide classroom observations was designed to elucidate several dimensions of lesson enactment that might influence student outcomes. We looked at classroom arrangement (e.g., Is the room set up for lectures, with desks in rows, or are students working together at tables?), the pedagogical style of the teacher (e.g., How much of the lesson is the teacher talking? What kinds of questions is the teacher asking?), and the level of engagement among students

CoolThink@JC Implementation: Key Takeaways

- The pedagogical style of *CoolThink@JC* teachers ranged from formal to student-centered, and became more student-centered over the course of the school year.
- Time constraints and project complexity made it difficult to complete lessons as intended in some cases. Teachers made adaptations to address this challenge.
- Teachers perceive the Teacher Development Course and associated supports to be helpful for understanding the lesson content and rationale for teaching computational thinking.
- Implementing *CoolThink@JC* offers an opportunity for professional growth that is appreciated by both teachers and principals.

(Do students appear interested in the activities?). Observers also paid attention to the amount and quality of communication in the classroom; both teacher-student and student-student interactions were noted.

Classrooms at the four focal schools exhibited a range of pedagogical styles and physical environments. While some lessons exemplified the student-centered pedagogy intended by the *CoolThink@JC* developers—with students engaging in guided exploration, working in groups and discussing their ideas with each other as the teacher circulated through the room to offer support—others demonstrated a more formal, teacher-centered pedagogy, characterized by students primarily sitting still, listening to instructions, and working out tasks according to specific steps with little room for creative input.

Although one teacher was observed to spend an estimated 70% of class time lecturing, this was not the norm. The data indicate a broad range of pedagogical styles, from extremely formal to student-centered, with most classes observed to be more on the student-centered end of the spectrum. Some teachers report that this teaching style represents a major departure from their usual practice, while other teachers at one focal school say that this "activities-based approach" is in line with their school's philosophy and is already being employed across the curriculum.

In the second wave of observations, conducted near the end of the school year, 10 of the 11 teachers were characterized by the observer as primarily a 'facilitator' or 'coach', and only one as a 'transmitter

In one classroom, the teacher locks students' screens often to make sure they are paying attention to the lecture. Following the explanation, students work independently to complete the steps on the handout. Quiet and discipline are enforced by the teacher.

In another classroom, students work in groups of two or three to complete a project. They are cooperating and fully engaged in the work. The teacher acts as a coach, offering hints and asking thought-provoking questions as needed.

of knowledge.' This marked a shift in practice from the first half of the year, when nearly half of the teachers observed (5 of 12) were characterized as primarily a 'transmitter of knowledge'. One classroom did not appear to have strong discipline, but many ideas were exchanged and most students exhibited intense interest in their projects. In nine of the 11 lessons observed near the end of the school year, teachers spent at least half of the class period circulating and answering questions as students worked on projects; in one case 95 percent of class time was spent in this way.

Teachers described several ways that *CoolThink@JC* classes differed from their usual practice. When asked about assessing student progress, for example, one teacher said, "We don't rely on examinations and tests" and "We need to interact more, observe how they are doing." They also point to differences in the nature of the learning targets, pointing out that, "Other subjects are more about memorization." Teachers tended to agree that *CoolThink@JC* is demanding for them ("Teaching CoolThink takes more energy!") One teacher expressed that he saw *CoolThink@JC* as requiring a teaching approach similar to that of his main subject area of visual arts, in that it is focused more on inspiring the students and less on results. A principal from a different school concurred, explaining, "Overall the way of teaching CoolThink is more like carrying out extra-curricular activities, which means teachers need to give minimal instruction, but to allow students to work with different alternatives - it is also common in other non-academic subjects like visual art."

In interviews and focus groups, principals, teachers and students at the focal schools were all asked what was different about *CoolThink@JC* classes.

"It's a chance for teachers to acquire new knowledge and inspire them for more innovative approaches in teaching computer." (principal)

"Students have more control …teachers play more of a facilitator role instead of the traditional teacher role." (teacher interview)

"CoolThink is different … we can have free time for discussion, to work on our assignment. It is not necessary to follow the instructions as the teacher always encourages us to do something different from others." (student)

On the Educator Survey, all teachers had the opportunity to share their impressions of how *CoolThink@JC* is different from other subjects they teach.

"Less time is spent on teaching; more time is allocated for the student to explore."

"It focuses on developing the creativity and logical thinking of the students."

"We provide a greater learning autonomy to the students."

Modifications

On the Educator Survey, over half of the teachers of *CoolThink@JC* report making modifications to the lessons and a quarter of them report using additional materials beyond what is specified in the provided materials. Teachers' input on modifications they made were used by the *CoolThink@JC* team to inform lesson improvements with an eye to scalability. The most common reason given for lesson modification was insufficient instructional time to cover all required content, leading teachers to seek more efficiency in the flow of the class. In the second wave of site visits, three main ways of coping with this challenge were evident: reducing tasks, using pre-coded templates, and integrating *CoolThink@JC* lessons with assignments in other classes.

Some schools decided to cope with the time crunch by reducing the number of tasks in each unit so that students would only need to focus on the key concepts. This was the most commonly cited modification in the Educator Survey. In interviews, teachers said that time constraints and project complexity made it difficult to complete lessons as intended in some cases, so they limited the required activities. Activities that were cut were sometimes turned into optional tasks for the more capable students to attempt.

Some teachers reported relying on pre-coded templates to ensure they could get through the lesson activities in the time allotted. This finding highlights the tension between pedagogical goals and practical constraints. In some cases, the pre-coded templates were observed to save time, but they also tended to encourage a less student-centered approach to instruction, as teachers rushed students through pre-determined steps rather than letting them experiment and solve problems on their own. Some of the more capable students were observed to lose motivation because they were just following instructions. In a student focus group, one said, "We just follow notes. Without the notes we can do nothing."

Teacher descriptions of how they adapted the lessons

"I simplified the unit because teaching time was not enough for students to explore."

"If we follow the original design, the students could not finish the tasks in the lessons. Therefore, I provided the programming code to help them."

In one school, teachers worked together to find a creative solution to the time challenge. The final mini-project was modified so that instead of asking students to design an app that could solve a problem in their daily life, as the assignment was presented in the *CoolThink@JC* materials, they were tasked to redesign the Hong Kong Tour app (from a prior lesson) by using the topic of 'One Belt One Road', which is a unit taught in General Studies. The advantage of this modification was that it enabled class time from another discipline to be leveraged. Students could do the necessary research for the app content in their general studies class, and work on the app development in their *CoolThink@JC* class. Both classes benefitted from the extra time.

Another common adaptation is the use of additional materials, which included instructional YouTube videos, PowerPoint presentations or other reference materials. A quarter of the teachers on the Wave 2 survey report using additional materials to teach *CoolThink@JC*.

Two of the focal schools reported incorporating small robots to extend students' use of their new coding skills and, in some cases, to apply them in other classes. In these schools, students are using mBot, which has an interface similar to Scratch, allowing students to use their coding skills to control the robot. These schools are also using micro:bit robots in a similar way, to enhance interest and enthusiasm for MIT App Inventor lessons. Interestingly, students in focus groups in these schools that were applying coding knowledge to robotics showed greater enthusiasm for *CoolThink@JC* than students in the other two focal schools.

Engagement

Approximately 80% of teachers reported high engagement among students on the Educator Surveys, responding either 'agree' or 'strongly agree' with the statement, "Students demonstrate enthusiasm and effort in completing the assigned tasks." Interviews with teachers and student focus groups suggest that engagement varies to some degree across contexts and grade levels. Several students expressed high interest in the learning to make apps and talked about how different it was from other classes. A teacher concurred, saying, "Students find learning coding is more interesting than traditional computer studies that are more focused on hard skills. It allows room for creativity." However, younger students sometimes found lessons hard to follow and teachers reported difficulty in keeping them engaged. For example, one P4 student lamented, "It's very complicated, I don't understand but I don't want to ask questions - I don't want to look stupid and let others know I cannot follow." And another P4 student commented, "The maze game is very difficult, I failed to make the cat pass the maze and I don't know how to fix it. I decided to give up that assignment." Comments such as these underscore the challenge of providing coding challenges to young students that encourage creative problem solving without overwhelming them to the point that they lose motivation. Lesson revisions have sought to address these challenges, streamlining tasks and providing increased scaffolding in the form of screenshots and more detailed directions.

The second wave of data collection showed a slight drop in perceptions of student enthusiasm and effort among teachers (from 83% to 79% agreement with the statement, "Students demonstrate enthusiasm and effort in completing assigned tasks."). Interviews and focus groups showed that this might be attributable to students finding the MIT App Inventor lessons in the later part of the course less accessible than the Scratch lessons that they started out with. One principal reported reservations on using MIT App Inventor, saying that it is good

Students appreciate the creative nature of *CoolThink@JC* activities.

"CoolThink class is my favorite lesson. It's the only time I can express my own ideas during class and through my assignments." (P6 student)

"Activities of CoolThink class are interesting, it isn't like Mathematics or Chinese that need to memorize so many formulas or content." (P4 student)

"I need to 'use my brain' during CoolThink class, it makes me feel I am less robotic." (P6 student)

"I do really bad in most of the subjects, CoolThink gives me a chance to discover my real strength, I have been selected to join inter-school competition of coding." (P6 student)

to teach CT concepts, but it's a bit complicated to write the blocks when compared with other coding tools (e.g. Micro:bit).

In one focus group of P4 students, who had experienced both Scratch and MIT App Inventor lessons, all agreed they are more interested in Scratch and find it easier to learn. They reported that they lost their interest in *CoolThink@JC* in the second semester because they found MIT App Inventor too difficult. In a second focus group from a different school, students reported feeling frustrated when working on MIT App Inventor, saying there are too many blocks and they find it difficult to locate the one they need to use. One student commented, "Some units in Scratch are quite interesting, but App Inventor is not fun at all."

While engagement in *CoolThink@JC* lessons is strong overall, developers are using this feedback from students and teachers to fine-tune their designs with the goal of making the full range of lessons more accessible and fun for students.

Collaboration

A key element of the intended *CoolThink@JC* pedagogy is for students to collaborate on coding projects. On the first wave Educator Survey, 83% of respondents agreed with the statement, "Students work together to build their skills and knowledge through the assigned tasks." In the second wave, that number had dropped to 66%. This may be attributable to the above-mentioned issues with MIT App Inventor vs. Scratch. In site visits, we found a wide variation in the level and types of collaboration occurring in classrooms, as well as a range of opinions concerning collaborative learning activities among students.

In one focus group, one student expressed appreciation for peer supports but other students revealed a limited understanding of collaboration, characterizing it as dividing up tasks rather than working together. Some students reported preferring to work alone than with a less capable partner. This sentiment may possibly be traced to teachers' reported strategy of pairing capable students with less capable ones. As one student complained, "I like to work independently, I don't want to work with stupid partner."

Different classroom models of "collaboration"

In one class, the whole class is engaged in intense discussion and sharing. Lots of ideas are being exchanged, not only within project groups, but also among the whole class. The atmosphere is a bit noisy, but students are clearly focused on the projects.

In another class, stronger students have been paired with students who are struggling with the concepts. Collaboration in this room consists of students taking turns to use the computer rather than working together on projects.

What supports and barriers did *CoolThink@JC* educators experience?

Educators identified several supports and barriers to successful implementation. Interviews reveal that one of the most crucial elements of the program is the training and ongoing support provided by partner universities EdUHK and MIT, which received very positive reviews from both teachers and principals.

The support of Teaching Assistants (TAs) is also highlighted by several teachers and principals. This support appears to have been more crucial at the beginning of the school year, when many teachers were less familiar with the lessons. TAs performed a range of functions in the classrooms, from solving technical issues to answering student questions to leading lessons. In one school visited by researchers, some teachers had been assigned their *CoolThink@JC* teaching roles relatively late and received only a brief school-based introduction rather than the more extensive training provided by *CoolThink@JC*. These teachers expressed that they would not have been able to teach the lessons without the TA, who acted as teacher much of the time. In another school, which has a strong technical support staff, teachers expressed that the TA was used mostly for administrative tasks.

One set of barriers cited by teachers was the type and arrangement of classroom resources. Fixed chairs impeded group collaboration, large monitors blocked views, moving furniture took precious class time, and hardware problems caused delays in some cases.

How do educators see the value of *CoolThink@JC*?

The challenges outlined above are typical of a new innovation in the pilot stage of real-world implementation. Despite important challenges concerning time constraints and meeting the needs of a range of students, the overall reaction to *CoolThink@JC* has been extremely positive. Most teachers (80%) agree that *CoolThink@JC* is fun for students, and 78% say that it succeeds in equipping students with basic programming capabilities. Three quarters of teachers surveyed also said that they think the course motivates students to learn computational thinking. Eighty-one percent of teachers surveyed said they would recommend *CoolThink@JC* to others.

Training and support materials are appreciated by teachers and principals

"Supports from the program are very comprehensive, especially the high-quality training that caters to the needs of different teachers, including those who do not have experience in teaching computer or coding." (teacher)

"Systematic and resourceful supports help the school speed up and expand CT learning to more students. The ecosystem of CoolThink allows the school to give feedback and there are professional working teams from MIT and EdUHK to adjust the lessons, which is more effective than doing this through a school-based developed program. (principal)

"The comprehensiveness of the program and lesson design [is a plus]. It's less seen in Hong Kong to have coding curriculum that is specially designed for primary students, especially to P4." (teacher)

In interviews, teachers spoke of the extra work that teaching *CoolThink@JC* requires, but they also expressed appreciation for the sense of professional growth that has come with teaching computational thinking. Principals also noted this effect. As one said, "Enrolling into CoolThink motivates teachers to self-learn and inspires them to improve their teaching methods to cope with the rapidly changing environment that students are facing."

Classroom Layout Matters

"The computer lab is using desktops but the monitor in front always blocks the sight of students. It would be more ideal to change the desktops to laptop with touchscreen function (like those currently using in classroom) as kids find more interesting and easier to navigate with touch screen." (teacher)

CoolThink@JC Student Outcomes

With this understanding of teachers' and students' experiences of *CoolThink@JC*, we now turn to measured student outcomes in computational thinking. We will use results from the CT Concepts and CT Practices assessments and CT Perspectives survey to describe how much students learned from participating in *CoolThink@JC*, as compared to students in comparison schools. After two years of data collection, we are now in a position to see whether the program is beginning to have an impact on student CT outcomes, and if so, how much more students in pilot schools learn in comparison to their peers in similar schools who did not experience the *CoolThink@JC* lessons.

CT Concepts: Key Takeaways

- Students in pilot schools exhibited more learning in CT Concepts than similar students in comparison schools.
- Boys benefited more from the pilot lessons than did girls.
- Other students who benefited more than their peers include students who showed some knowledge of CT Concepts at baseline, who had higher baseline math test scores relative to other students in their school, and who were in classes with fewer special education needs (SEN) students.

CT Concepts

As described previously, the CT Concepts assessment measures students' knowledge of computational thinking concepts that are core to the *CoolThink@JC* lessons, such as repetition, conditionals, parallelism and sequencing, and data structures. We focus on CT Concepts results from the first year of *CoolThink@JC* lessons (Level 1), for which we have data from all students in grades 4 through 6 in all study schools. We also include initial exploratory results from a smaller group of students in schools that have had two years of participation and their comparison schools. For year 1 analyses, we also report on whether and how student learning varied by important subgroups, such as gender, grade level, baseline achievement, and baseline computational skills and experiences.

How much did students learn from *CoolThink@JC* lessons in their first year?

In this report, we look at student learning in two ways. First we look at gain, or improvement in test scores, as measured by the difference between scores at baseline and the end of year one, in pilot and comparison schools. This shows what students in each condition learned in their first year. Next we look at impact, which is the estimate of how much benefit students receive from participating in *CoolThink@JC* (i.e. outcomes they would have been unlikely to achieve without the program). Impact analyses also look at student learning in both pilot and comparison schools, but this time we use hierarchical linear modeling (HLM) to take into account a number of factors that help us to make a fair comparison of similar students in similar circumstances. These factors include the nesting of students within classrooms and schools, and important characteristics at the student, classroom, and school levels.

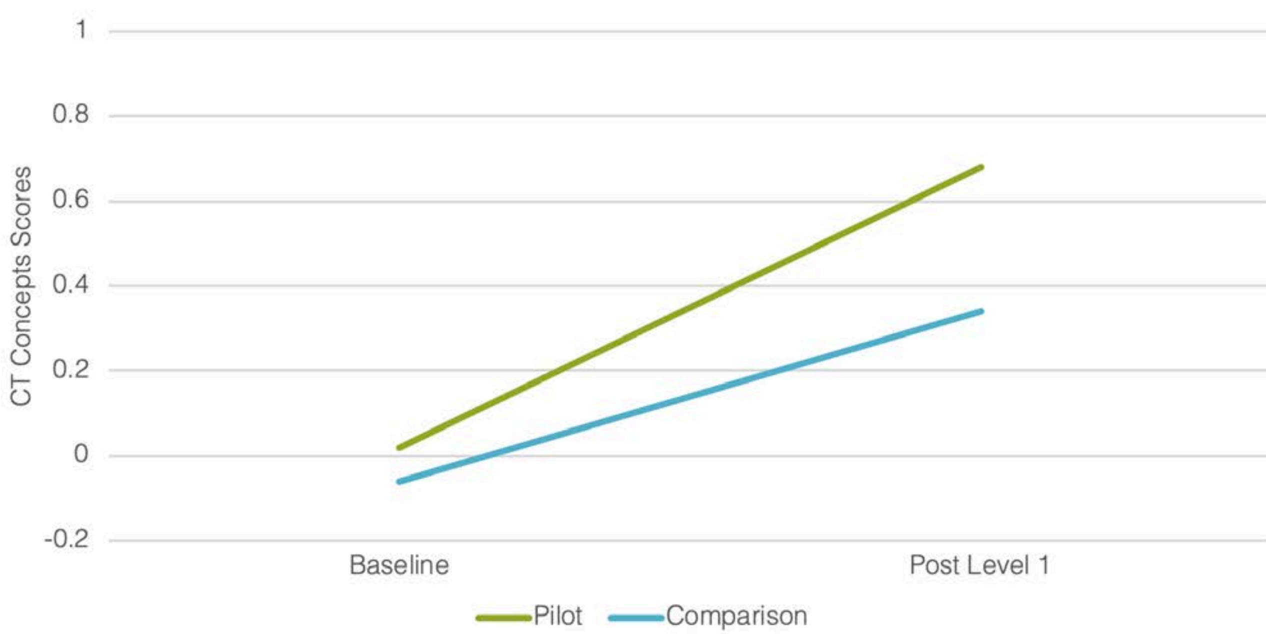
Gain: the difference in score between baseline and outcome, indicating how much students learned.

Impact: the difference between student gains in pilot versus comparison schools, adjusting for differences in student, classroom and school characteristics, indicating how much students benefitted from *CoolThink@JC*.

These analyses include all students who have taken the Level 1 lessons to date, combining students in Cohort 1 and Cohort 2 pilot schools in order to improve the reliability of the estimates.

Figure 1 illustrates what students in pilot and comparison schools learned in CT Concepts in their first year.

Figure 1. CT Concepts Scores for Students in Pilot and Comparison Schools, Baseline to End of Year 1.



Students in pilot schools gained more in CT Concepts than students in comparison schools in the first year of *CoolThink@JC* lessons. CT concepts score increased from 0.02 to 0.68 for students in pilot schools, for a gain of 0.66 points. Students in comparison schools still increased in their CT Concepts knowledge over the course of the year, but they learned less: their score increased from -0.06 to 0.34, for a gain of 0.40 points.

We interviewed teachers about what they see students gaining in terms of computational thinking and problem-solving abilities. Several teachers emphasized that the students are young, and that the value of the lessons is more in the exposure to computational thinking than in the acquisition of specific skills. They see the lessons as planting a seed that may take a while to bear fruit.

Over three quarters of pilot teachers surveyed agreed that learning computational thinking develops students problem-solving, communication, and collaboration skills, and that it helps students to learn and perform better across all disciplines.

We then looked at impact, or the benefit that students receive from participating in *CoolThink@JC* when compared with similar students who did not participate. As described above, this analysis uses a 3-level HLM model that adjusts for students' baseline CT Concepts scores and student, classroom and school characteristics.

The *CT Concepts Score* places students on the continuum of ability in computational thinking concepts based on performance on items of carefully-designed varying difficulty. On this scale, zero approximates the mean CT Concepts score at baseline. The vast majority of students score between -2 and 2 on this scale.

Figure 2 illustrates the estimated gains in pilot and comparison schools separately; the difference in gain between pilot and comparison schools represents *CoolThink@JC* impact one year after the intervention. The asterisks indicate whether the difference in the gain is statistically significant. A 0.05 significance level indicates that if there is no true difference between gains for pilot and comparison students, we'd have a less than 5% chance in obtaining the observed difference; a 0.01 significance level indicates that chance to be less than 1%.

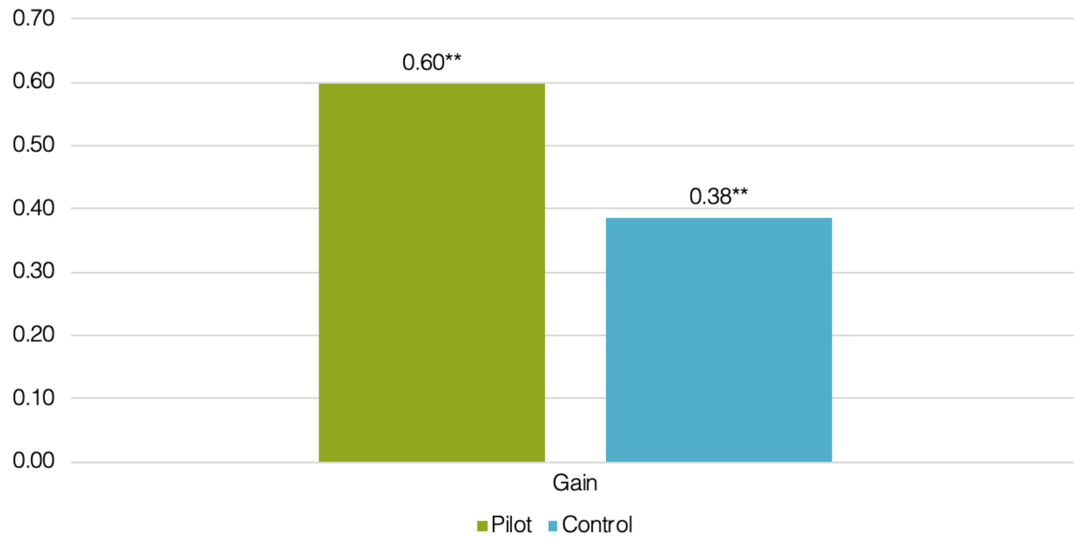
Students in pilot schools exhibited significantly more learning (higher gain) in CT Concepts score than similar students in comparison schools. The estimated impact of 0.21 points, calculated from the difference in gain scores between pilot and comparison schools, is statistically significant at the 0.01 level (detailed

What teachers say about students' CT learning

"It is too soon to judge if they really understand the CT concepts, but allowing students to have basic ideas about CT is good enough for primary students."

In an assignment, one teacher values the process: "whether students ask questions, whether they put in new ideas, whether they work out the things reasonably—not necessarily correctly."

Figure 2. Estimated Gain in CT Concepts, Baseline to Year 1.



*Note. **indicates pilot schools significantly differ from the comparison schools at the 0.01 significance level after adjusting for the baseline CT concepts, and student- and school-level characteristics.*

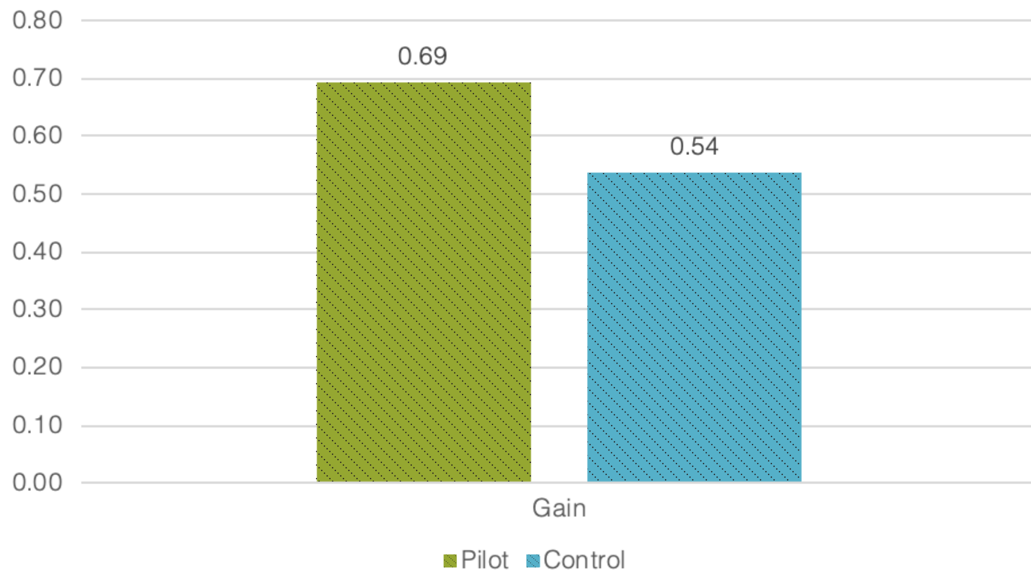
estimates in Appendix C, Exhibit C2). This estimated impact in CT concepts score translates to an effect size of 0.20. Effect size is a common measure of the magnitude of an impact, and 0.20 is considered moderate in education research. Given that these results are from a pilot program, which are rarely the targets of rigorous research because their outcomes are not yet stable, it is notable that a moderate impact was achieved.

Impact through year 2

We conducted exploratory year 2 impact analysis comparing CT Concepts learning after *CoolThink@JC* Level 1 and Level 2 pilot lessons in 10 Cohort 1 pilot schools and their 12 comparison schools.

As shown in Figure 3, it appears that Cohort 1 pilot schools exhibited higher gain in CT Concepts score than similar students in comparison schools. However, the difference is not statistically significant. Due to the small number of schools in this analysis, the results are not conclusive. Students' learning progression over two years will be confirmed with additional data from all 30 schools in the coming year.

Figure 3. Estimated Gain in CT Concepts Score for Students in Cohort 1 Pilot and Comparison Schools



How much did different subgroups benefit from *CoolThink@JC* in their first year?

CoolThink@JC has goals of promoting equity by giving all students the opportunity to learn computational thinking. For this reason, we look at the Level 1 data in 30 schools to see whether all students had similar outcomes from their first year of participation in the pilot, or if some types of students benefitted more.

In this analysis, we looked at student CT concepts score by gender, grade level, baseline concepts score, baseline math achievement, special education needs (SEN), student income, and relevant indicators from CT perspectives assessment such as computer use at home, internet access at home, and background in programming, as well as school prior programing experience and paper vetting score. We also looked at hours of *CoolThink@JC* instruction a student had received by the time of the CT Concepts assessment. Because we do not have individual information for student SEN and family income, we examined whether students in classrooms with a higher percentage of SEN students or students under government financial assistance scheme benefit more, or less, from the pilot lessons than students in classrooms with a lower percentage of such students.

Below are findings on the gains in CT concepts scores relative to similar students in comparison schools, from baseline to end of year 1, for different groups of students (detailed estimates in Appendix C, Exhibit C2). As a result of participating in the *CoolThink@JC* pilot:

- Boys gained more than girls.
- Students with higher baseline CT Concepts scores (suggesting that they already had some knowledge of CT Concepts) gained more than students with lower scores.
- Students with higher baseline mathematics scores gained more than students in the same school whose math scores were lower.

- Students in classes with fewer SEN students gained more than students in classes with more SEN students.

These analyses do not detect a significant effect of other tested factors — including grade level, computer use at home, internet access at home, prior programming experience, and classroom percentage of students receiving financial aid, percentage of non-Chinese-speaking students, and hours of *CoolThink@JC* instruction — on the benefit students receive from *CoolThink@JC*.

The data gathered from site visits offers some insights into factors that may contribute to some of these observed differences in student performance.

What teachers say about gender

When asked about differences between the performance of boys and girls, one teacher noted that girls may be less interested in *CoolThink@JC* because they are more focused on excelling in core subjects. Said another, "Girls in general are less eager to learn coding, they prefer to spend more time on language studies." Another teacher noted that the girls in his class "tend to be more passive," indicating that their approach may not be as well-suited to getting the most out of *CoolThink@JC* lessons. Indeed, in one class, boys were observed to ask more questions and express more ideas than girls, to the point that the teacher in this class encouraged the girls to speak up.

It is important to note that these comments were representative of about half of the small group of teachers interviewed. The other teachers did not note any differences between boys' and girls' performance. One said, "The abilities of the students are very even. There are no material differences between the skillsets of male and female students."

Gender in Class (a composite of observations and teacher descriptions):

In this mixed gender class, both boys and girls are working on their *CoolThink@JC* projects. Most girls appear focused on completing the procedures correctly, and are noticeably reluctant to ask the teacher questions or express their ideas, while many boys are asking a lot of questions and enthusiastically throwing out ideas for the apps. In fact, the boys get so boisterous that the teacher asks them to settle down. While creating apps, the girls tend to focus on the story sequence, and on designing attractive backgrounds, while the boys are concerned with adding functionality to their apps, such as add-ons, motion and audio features.

This vignette illustrates how the same pilot lessons, enacted by the same teacher, can result in very different learning experiences for different students. Although this discussion is based on the impressions of a small number of teachers, and should be viewed as such, their comments may offer clues about how teachers might better support girls' achievement in *CoolThink@JC*. A recommendation for future teacher development would be to focus on developing teachers' understanding of these potential differences and of which features of *CoolThink@JC* lessons best support the intended outcomes.

The research will continue to track relative gains by gender in the final year of the study, to see whether the CT learning trends cited here persist over time, or whether girls begin to benefit more with two or three years of CT study.

What teachers say about mixed ability levels

In the Educator Survey, teachers cited the range of ability in their classes as one of the major challenges of teaching *CoolThink@JC*. Classroom observations revealed some variation in the strategies that teachers employ to cope with this range of ability, as well as some qualitative differences in the type of learning activities that students of different sub-groups are engaged in.

Special educational needs (SEN) students require special care, one teacher noted, and the lessons in *CoolThink@JC*, although well-prepared in his view, don't cater to SEN students or those with lower academic ability. A few teachers said that SEN students tend to be less engaged in lesson activities, but one teacher expressed surprise that some SEN students could tell the teacher their own ideas for modifying the plan for creating a particular app.

In some cases, students with higher ability are encouraged to elaborate more on projects, complete optional challenge tasks, or design modifications to their apps. In other classes, though, they were observed to be assigned to help those who were struggling. The different strategies create different learning opportunities for high ability students; this also highlights situations when students of lower academic ability are engaged more in the procedural work of following instructions, while those of higher ability are, in some contexts, experiencing more creative problem-solving challenges.

CT Practices

While the CT Concepts assessment focuses on students' knowledge, the CT Practices assessment measures students' understanding of computational thinking skills that are core to *CoolThink@JC*, such as algorithmic thinking, reusing and remixing, testing and debugging, and abstracting and modularizing. The CT practices typify the kind of thinking and activities essential to applying and using knowledge of 'CT concepts' for computational problem-solving.

CT Practices: Key Takeaways

Students in pilot schools exhibited more learning in CT Practices than similar students in comparison schools.

How much did students gain in CT Practices from *CoolThink@JC* lessons in their first year?

This section describes changes in students' understanding of CT Practices after *CoolThink@JC* Level 1 pilot lessons. As a contrast, we also looked at the changes for comparison school students during the same period of time. This analysis only includes the 20 Cohort 2 schools, as students in Cohort 1 schools were not assessed at baseline due to the timing of data collection for CT Practices.

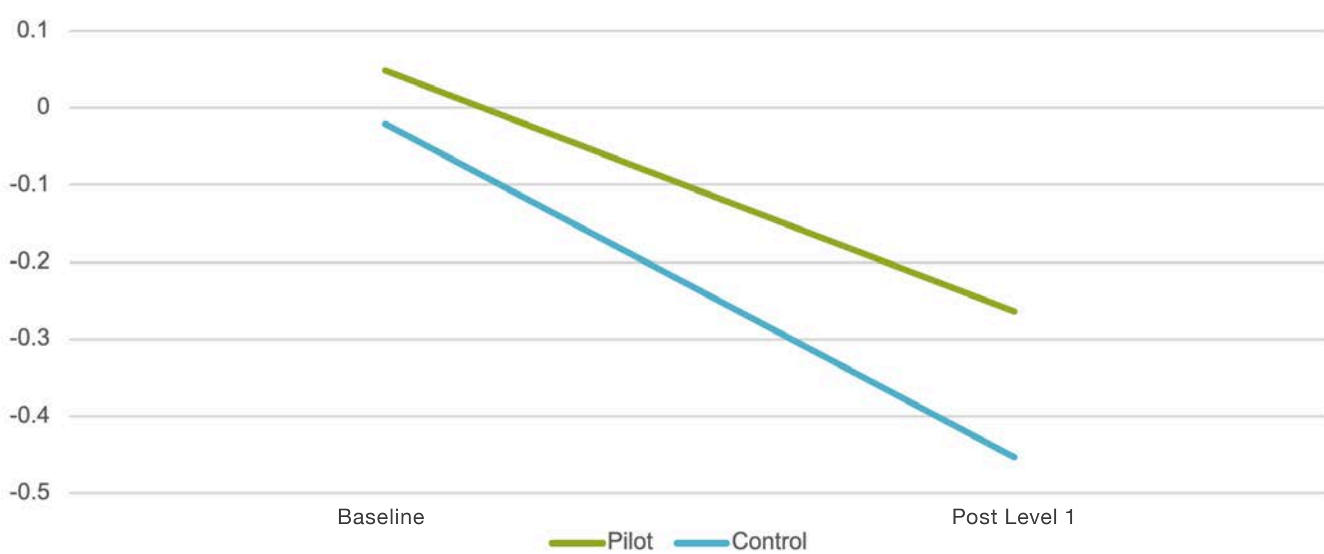
It is important to note that the CT Practices scores in this midline report appear to have been strongly affected by the timing of test-taking. For CT Practices, the baseline was taken in September, near the beginning of the school year. The follow-up was taken in June, often after compulsory testing for the year was over. Students spent far less time working through the assessment in June than they had in September[6]. This speed of administration seems to have had an impact on scores. Students' CT Practice scores dropped over the course of the year in both pilot and comparison schools. Nevertheless, because the issue of administration timing affected pilot and comparison students in similar ways, any differential gain would still reflect the impact of the pilot program.

Figure 4 illustrates one-year progress in CT Practices between pilot and comparison schools.

Although students' CT Practices scores dropped on average for both groups, pilot students dropped less than comparison students in CT Practices score after one year of intervention. CT Practices score decreased from 0.05 to -0.26 for students in pilot schools, for a loss of 0.31 points. The score dropped from -0.02 to -0.45 for students in comparison schools, for a loss of 0.43 points.

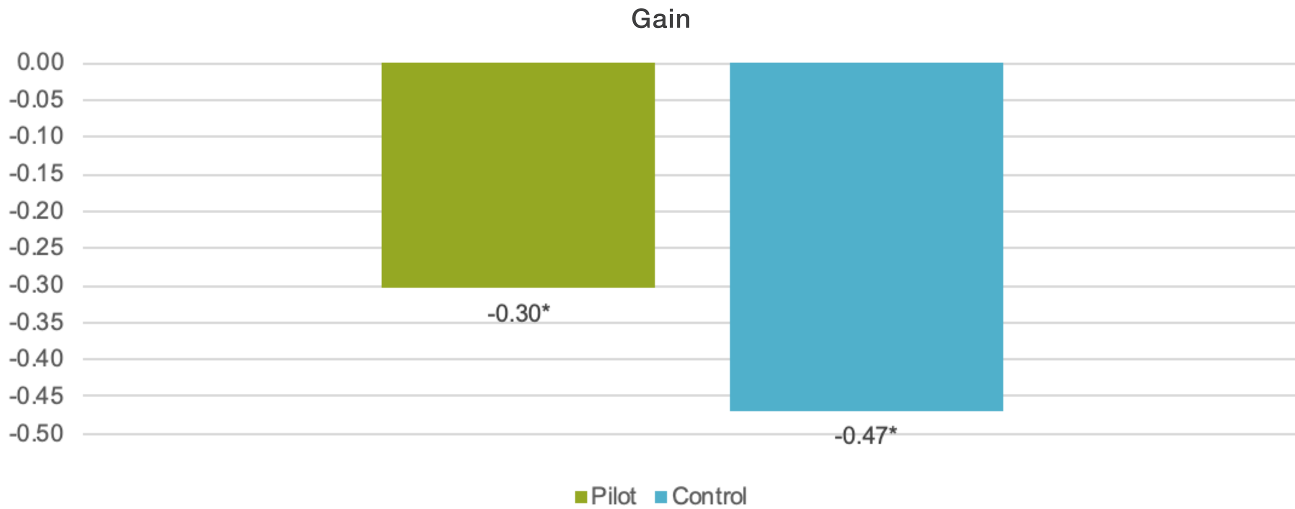
Students participating in the pilot exhibited lower loss in CT Practices score than similar students in comparison schools, as illustrated in Figure 5. The estimated impact of 0.17 points on the IRT score, calculated from the difference in gain scores between pilot and comparison schools, is statistically significant at 0.05 significance level (detailed estimates in Appendix C, Exhibit C6). This impact in CT Practices score translates to an effect size of 0.23, which is considered moderate in education research. SRI, with the support of initiative partners, is working to ensure improved administration in 2019, and will have the opportunity to confirm CT Practices gains more accurately in the coming year.

Figure 4. CT Practices Scores for Students in Pilot and Comparison Schools, Baseline to End of Year 1.



6 Students spent an average of about 13 minutes on CT practices in the September 2018 assessment, but only about 7 minutes in the June 2019 assessment.

Figure 5. Estimated Gain in CT Practices Score for Students in Pilot and Comparison Schools.



*Note. *indicates pilot schools significantly differ from the comparison schools at the 0.05 significance level after adjusting for the baseline CT practices, and student- and school-level characteristics.*

Analysis of Student Projects

The CT Practices assessment described above looks at students' knowledge of the core activities of programming. Another lens on CT Practices is a more practical one: are students able to use these computational thinking practices to develop a computer program?

This section summarizes findings of the Education University of Hong Kong (EdUHK) in their analysis of teachers' scores of the programming projects that students completed as part of their *CoolThink@JC* lessons. For more information on this work, please see Education University of Hong Kong (2018).

Students complete required thematic programming projects in each level of the *CoolThink@JC* lesson sequence to integrate and demonstrate their understanding. In Level 1 students complete one Scratch and one MIT App Inventor project on the theme of family; in Level 2 they complete a final MIT App Inventor project about travel. In many classes (representing approximately 59% of Level 1 projects and 77% of Level 2 projects) these activities were completed by students working in pairs; in other classes they worked individually or in small groups.

After several hours of training on the scoring process, teachers used 5-point rubrics to grade the student projects on five dimensions: implementing a solution, using appropriate programming logic, designing the screen in a logical and aesthetically pleasing way, experimenting and creating something novel, and connecting or synthesising ideas. The resulting scores were combined for a total of 80% of a student's grade (with "implement a solution" and "using appropriate programming logic" receiving the highest weights in that computation), and an additional 10% was awarded for the use of particular CT concepts or features. With this possible 90% total, researchers deemed 45% or above as a passing score.

Student projects have been collected and scored for the past two years, with the ten Cohort 1 schools participating in 2016-17 and all thirty pilot schools in 2017-18. In the 2017-18 school year a total of 9377 student projects from Level 1 received scores (of which 4724 projects were in Scratch and 4653 in MIT App Inventor), as did 1274 MIT App Inventor projects from Level 2.

EdUHK's analysis found that in 2017-18, passing scores were given to approximately 76% of Level 1 projects and 73% of Level 2 projects. Scores for the individual rubrics are summarized in Table 6 below. As seen in the table, students tended to do better on implementing a solution and creating a screen design, but had more challenges with displaying innovative thinking (on the "experimenting and creating something novel" rubric) and connecting or synthesizing ideas.

The analysis also found that Level 1 students who had the 14-hour version of the lesson sequence had higher pass rates on their MIT App Inventor projects than those who had the 9-hour version (80.5% vs 67.8%). Results suggest that the extra lessons in the 14-hour version, which allowed students more opportunity to explore the concepts they had learned, were important to their overall success.

Overall, while some students still struggled, the data show that the majority of students are able to grasp the programming concepts and apply them when creating their own programming project.

Table 6: Percent of projects that score a 3 or above across the rubrics for the 2017-18 school year.

	Implement a Solution: % of projects that score 3 or above	Programming Logic: % of projects that score 3 or above	Screen Design: % of projects that score 3 or above	Innovative Thinking: % of projects that score 3 or above	Connecting, Synthesizing, Transforming: % of projects that score 3 or above
Level 1 Scratch	71.9% (out of 4609)	74.2% (out of 4610)	75.5% (out of 4625)	69.7% (out of 4624)	68.2% (out of 4617)
Level 1 MIT App Inventor	75.4% (out of 4539)	73.1% (out of 4532)	79.0% (out of 4574)	67.2% (out of 4553)	65.4% (out of 4555)
Level 2 MIT App Inventor	73.6% (out of 1248)	67.7% (out of 1252)	75.4% (out of 1253)	60.8% (out of 1246)	65.7% (out of 1246)
Total	73.6% (out of 10,396)	72.9% (out of 10,394)	77.0% (out of 10,452)	67.6% (out of 10,421)	66.7% (out of 10,418)

Source: data from the Education University of Hong Kong

CT Perspectives

In contrast to CT Concepts and Practices, which are assessments of students' knowledge and abilities in computational thinking, CT Perspectives is a survey that asks students about their interests, beliefs, and motivations related to computational thinking. It is important to measure these perspectives because they describe students' shift in their evolving understanding of themselves, their relationships to others, and the technological world around them as they participate in *CoolThink@JC* activities. CT Perspectives represent a unique dimension of the computational thinking framework that is not captured by CT Concepts and Practices.

CT Perspectives Sub-construct Definitions

(a) Belonging: Recognition of computing as a collaborative endeavor

(b) Interest in programming: Interest in programming and in thinking computationally

(c) Engagement: Attainment of "state of flow" level of focus when thinking computationally, including persistence in the face of programming challenges

(d) Meaningfulness/motivation to help the world: Motivation to use computation to solve problems and benefit the world

(e) Creativity: Perception of programming as a creative endeavor

(f) Digital self-efficacy/competence: Confidence in ability to program and think computationally

(g) Utility motivation: Perception that computational thinking is useful, and motivation to pursue it

The CT Perspectives survey is designed to measure seven sub-constructs, as shown in the box above. A given student's perspectives may vary substantially across these categories—for example, a student may be very motivated to study computational thinking without viewing it as a collaborative activity. As a result, we report the sub-constructs separately rather than combining them into an overall "CT Perspectives score."

How much did students' CT Perspectives change from the first year of *CoolThink@JC* lessons?

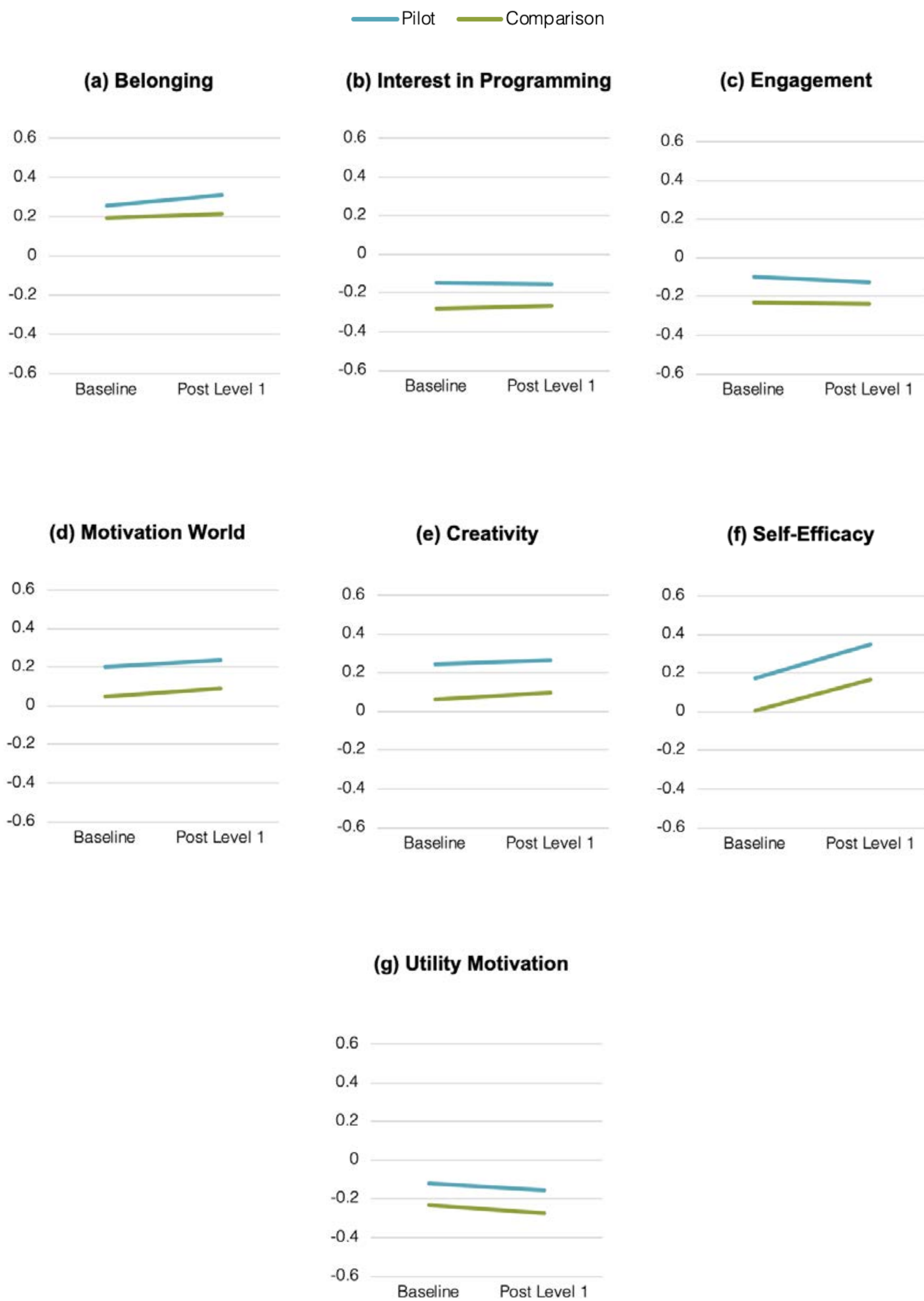
This section describes how students' computational thinking perspectives changed after *CoolThink@JC* Level 1 pilot lessons, in contrast with comparison students over the same period of time. We used HLM to estimate the benefit of *CoolThink@JC* with respect to CT perspectives. Similar to the previous analyses in this report, these analyses look across all Level 1 outcomes, combining students in Cohort 1 and Cohort 2 pilot schools to provide more reliable estimates of the impact of the *CoolThink@JC* Level 1 pilot lessons.

Students in both pilot and comparison schools showed minimum changes in most aspects of CT Perspectives after the first year of CoolThink lessons. The exception is self-efficacy, which showed increases for both types of schools. Figure 6 illustrates one-year changes in CT Perspective sub-constructs for pilot and comparison schools. Scores for *self-efficacy* (panel (f)), which is related to students' confidence in their ability to program, showed strong and comparable increases in pilot and comparison schools. This is not surprising, given that CT Concepts results demonstrate positive student learning from the pilot. For the other six sub-constructs, the absolute

CT Perspectives: Key Takeaways

- Students in pilot schools exhibited greater positive changes on all CT Perspectives sub-constructs than similar students in comparison schools; however, these differences are not statistically significant.
- Boys showed greater positive change in *creativity* and *self-efficacy* sub-constructs than girls participating in the pilot; whereas girls showed greater positive change in the *belonging* sub-construct than boys.

Figure 6. CT Perspective Scores for Students in Pilot and Comparison Schools, Baseline to End of Year 1.



changes after year 1 lessons are all very small for both pilot and comparison schools.

Across all sub-constructs, pilot schools had higher scores than comparison schools at both baseline and end of year 1. Similar increasing/decreasing patterns were observed for pilot and comparison schools on all sub-constructs except for interest in programming (panel (b)). For interest in programming, though changes for pilot and comparison schools were in opposite directions, these changes were too small to be considered as meaningful in both types of schools.

Impact analysis was conducted to test whether there were differential changes in CT perspectives between students in pilot and comparison schools, adjusting for students' prior coding experience and perceptions as well as student, classroom, and school characteristics.

Pilot school students exhibited greater positive changes or smaller negative changes than similar students in comparison schools on all CT Perspective sub-constructs, although these changes were not statistically significant. We estimated the impact of intervention by calculating the difference in the change scores between pilot and comparison schools. The estimated impact for creativity (.14) and utility motivation (.11) is nearly statistically significant (p=.08 and .07 respectively). Compared to similar students in comparison schools, students in pilot schools agreed somewhat more that programming is a creative endeavor and that computational thinking is useful, and they were more motivated to pursue it after attending the pilot lessons for one year. Detailed results are presented in Appendix C, Exhibits C8 - C14.

Impact Through Year 2

Exploratory analysis was conducted to examine the effects of *CoolThink@JC* Level 1 and Level 2 pilot lessons on CT Perspectives changes. In this midline year, these analyses could only be performed using the relatively small dataset of the 10 Cohort 1 pilot schools and their 12 comparison schools.

The initial analysis of sub-construct score patterns indicates increases in *belonging*, decreases in *interest*, *engagement*, and *utility motivation*, and a mixed direction of increases and decreases in *motivation world*, *creativity*, and *self-efficacy* for pilot and comparison schools. Impact analysis shows that the changes after two years of *CoolThink@JC* lessons were comparable for pilot and comparison school students with similar characteristics and school context on all CT Perspective sub-constructs except for *interest in programming*. For *interest in programming*, Cohort 1 pilot school students exhibited a significantly larger drop after two years as compared to similar students in comparison schools. This may be a result of students' stronger engagement in lessons that used Scratch than MIT App Inventor, as reported earlier in this report. However, this finding is based on a relatively small amount of data and hence is inconclusive. In the coming year we will be able to confirm these results by evaluating the impact of two years of *CoolThink@JC* lessons on CT Perspectives in all 30 schools.

How much did perspective changes from one year of *CoolThink@JC* vary for different subgroups?

Similar with CT Concepts, we conducted subgroup analyses to examine potential different changes in learning perspectives or perceptions as a result of participating in the pilot by gender, grade level, SEN, student income, relevant indicators from CT Perspective surveys, school prior programing experience, paper vetting score, and hours of *CoolThink@JC* instruction a student had received by the time of taking the CT Perspectives survey. The analyses found that

- Boys had significantly greater positive changes in *creativity* and *self-efficacy* from participating in the pilot than girls: they felt they learned more and experienced computing more strongly as a creative enterprise.
- Girls showed greater positive changes in *belonging* from the pilot than boys: they found more benefits from working with peers as they program.

Other factors, including grade level; prior programming experience and use of computers and the internet at home; classroom percentages of SEN students, students receiving financial aid, and non-Chinese-speaking students; and hours of instruction did not result in significant differences in CT Perspectives.

Scaling *CoolThink@JC*

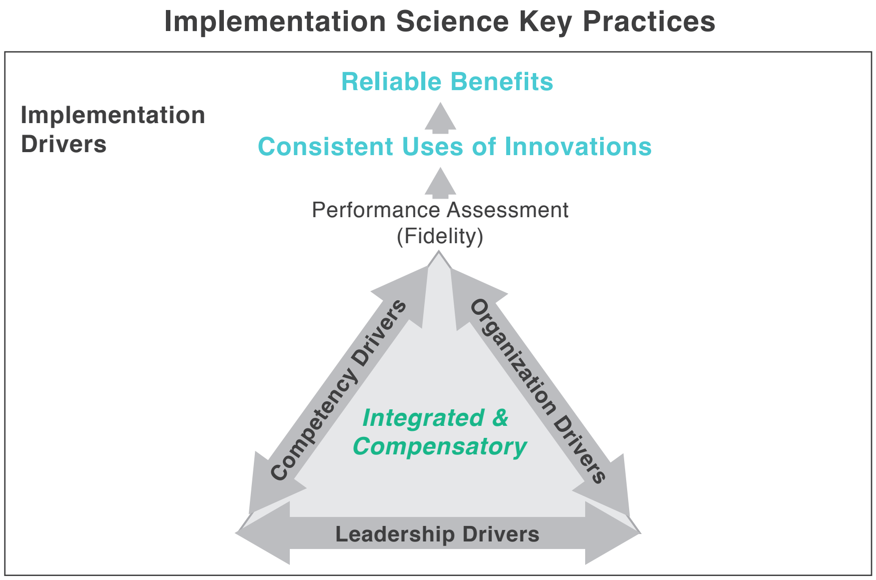
In the pilot phase of *CoolThink@JC*, the focus has been on the design and refinement of the *CoolThink@JC* lessons, and there have already been significant revisions to the program based on feedback from teachers. As project partners consider scaling this initiative into more Hong Kong primary schools, the focus will shift to how they can support widespread implementation with fidelity. In this work, the multidisciplinary field of implementation science offers guidance. In areas from healthcare to social work to education, there has been growing recognition that some factors influencing implementation are universal to any human service sector. The relationship between the **design** of any innovation, the **implementation** of that innovation, and the **context** in which it is being implemented, is an interdependent one. If any of these three factors is weak, the impact of the innovation, no matter how well crafted, will be diminished.

It is the unavoidable nature of implementation that there will be opposing forces at work: inevitably, an innovation will be changed to fit its context and, at the same time, the context will be altered to accommodate the innovation. An example of this is that over half of teachers say they modified *CoolThink@JC* lessons to fit them into the allotted time. Some examples of adjustments to context include integrating *CoolThink@JC* lessons with content from other subject areas and schedule accommodations made to allow more time to work on *CoolThink@JC* projects. The range of pedagogical styles discussed in the adoption section above also speaks to this tension. Whereas in one school, the principal sees the *CoolThink@JC* pedagogy as fitting into, and perhaps furthering, a schoolwide commitment to student-centered teaching, in another, the prevalence of a more formal teaching culture results in *CoolThink@JC* lessons being delivered in a more traditional instructional fashion.

Experts in the dynamics of implementation advise paying attention to three kinds of drivers of success (Fixsen and Blase, 2008): individual competencies (e.g. the professional development of *CoolThink@JC* instructors); necessary organizational supports (e.g. technology and adequate lesson time); and engaged leadership (e.g. school leaders supporting adoption).

Based on the research in this report, we offer considerations for scaling in each of these important areas.

Figure 7. Implementation Drivers, based on the work of Fixsen and Blase.



Source: https://nirn.fpg.unc.edu

Individual competencies

- *Continue to provide substantive professional development opportunities at scale.* The professional learning courses provided to pilot instructors received very positive feedback from teachers who participated, in part because they were much more in-depth than it is possible to accomplish in programs of a more typical duration (providing a week-long course and a ongoing series of 13 3-hour lessons rather than the more common dosage of two or three 3-hour lessons). Teachers who didn't participate, by contrast, tended to struggle with teaching the lessons, often asked TAs to teach their classes when they didn't feel prepared, and typically had less understanding of the purposes of the program. The train-the-trainer model currently under consideration holds promise, if the development team supports and monitors those professional learning experiences to ensure they maintain the intended dosage, approach and standard.
- *Encourage communities of practice.* A team approach to implementation has often been seen to improve accountability and sustainability. Common planning time is also an important enabler.
- *Provide guidance to teachers on support of different populations in CoolThink@JC classes.* One of the most common challenges teachers faced was how to work productively in mixed-ability classrooms, or to support less academically advanced students. Results from this research also suggest that teachers need guidance in engaging girls fully in programming activities and exploration, and that students in classrooms with higher percentages of special needs students learned less overall. While these differential outcomes are commonly seen across programming initiatives (e.g., Basu 2019), the situation suggests the need for increased teacher awareness of productive strategies to bridge the gap.
- *Provide guidance to teachers on how to adapt materials for contextual constraints.* This need

most commonly surfaced with regard to limited available time in a class period. It will be important for the lesson development team to consider alternative approaches to reducing the density of content and accessibility of materials without resorting to entirely structured teaching models that don't allow time for exploration and creativity.

Necessary supports

- *Continue to provide TAs to teachers, with a frequency tailored to need.* The pilot teachers used TAs in a range of ways depending on their own capacity and experience, but in general they were more frequently experienced as a necessary support early in a teacher's first year teaching *CoolThink@JC*, and in schools with less technical support resources. In future phases of *CoolThink@JC*, TAs could become an adaptive resource that are available to teachers who need it.
- *Attend to physical classroom setup.* Elements such as support of teamwork and visibility of materials can make an important difference to student engagement. Again, supports in this area may need to be differential for schools with different levels of technical capacity.

Engaged leadership

- *Consider ways to allow sufficient allotment of time to computational thinking instruction.* Some of the schools in the *CoolThink@JC* pilot were only able to set aside nine hours per year for computer lessons. If increasing students' capacities for computational thinking is recognized as a goal in Hong Kong, this allocation is not sufficient to enable deep engagement in the topic. This is a challenge for leadership at all levels in the education system. Integrating programming with projects in other areas of the curriculum is an approach that at least one pilot school started to explore.
- *Attend to other issues of structure.* In addition to instruction, it's important for leaders to consider schedules that allow for common planning time and ways to support teacher availability for professional development.
- *Make support for computational thinking education visible in the school community.* School leaders can establish clubs and events to engage and educate parents and students about the importance of computational thinking and celebrate the accomplishments of *CoolThink@JC* students.

Conclusions: Pushing the Frontier in Understanding Computational Thinking

The *CoolThink@JC* pilot extends the global trend that is bringing computational thinking instruction to primary and secondary schools around the world, positioning students to be critical problem-solvers and creative developers of technology. Introducing students to programming and computational thinking from Primary 4 grades and empowering them to apply these skills to create their own mobile applications and stories has the potential to plant the seeds of interest and engagement in leveraging technology toward whatever career choices students pursue in the future. It is indeed encouraging to hear young students remark that they enjoyed the *CoolThink@JC* classes and felt like they were in control of their learning and had to 'use their brain' instead of memorizing facts like in some other classes.

The *CoolThink@JC* pilot takes place in a cutting-edge field for both instruction and research, particularly for the primary school level. As such, rigorous large-scale studies of programs fostering CT skills in young children are not yet commonly available. This research is poised to make a substantial contribution to both the theory and practice of computational thinking instruction and assessment at scale. In light of the current lack of validated CT assessments, the principled design, development and validation of assessments for measuring CT concepts, CT practices and CT perspectives for evaluating the pilot study is an important step forward in the field of CS assessment research.

In many ways, the *CoolThink@JC* pilot can serve as a model for other nations striving to introduce CT into their curricular standards at the primary school level. While the program is still navigating some challenges for teachers and focusing on how to scale up to more widespread implementation with high fidelity, it has the opportunity to contribute important lessons and examples to the global community on how to develop, implement and sustain a large scale computational thinking curriculum at the primary school level; systematically assess computational thinking outcomes through a system of assessments spanning CT concepts, practices and perspectives; and iteratively refine the CT lessons and their implementation based on teacher and school feedback, implementation outcomes and student learning outcomes. The experience of this initiative can bring insights on the critical success factors for implementing new computational thinking curricula and for launching new educational initiatives with primary school children.

An important example of one of these critical success factors is the professional learning that is offered to teachers. *CoolThink@JC* teachers participated in two professional learning courses: one was a week-long in-depth workshop, and the other was offered as a series of 13 3-hour lessons, distributed over time to allow practice, reflection, and collaborative lesson development. This is in line with the recommendations of an extensive body of research, which points to a strong relationship between overall dosage, designs that offer ongoing structured engagement, and effectiveness in terms of teacher knowledge, skills, and pedagogy; this research also points out that more typical computer science teacher education programs are briefer and often less effective (Menekse, 2015; Yadav et al., 2013). A challenge to Hong Kong and to the *CoolThink@JC* team will be to design comparably rich, ongoing learning experiences for teachers that can affordably be delivered at scale.

It is laudable that a large-scale pilot program that is still undergoing modifications has demonstrated moderate effect sizes in gains in CT concepts and practices outcomes. Teachers and principals appreciated the program, and teachers acknowledged the comprehensive training and teaching support they received from the program. Most students and teachers agreed that the *CoolThink@JC* pilot classes were very different from that of other regular classes, and felt that the pilot helped steer the classrooms towards a more student-centered learning experience. In addition, some teachers felt comfortable in exploring the integration of *CoolThink@JC* lessons with other subject areas. Recognizing the synergies between computational thinking and other subject areas is an important aspect of being computationally literate, and the integration of CT lessons with other subject areas represents an important area the *CoolThink@JC* pilot could systematically explore in the future.

While still early in the progress of the pilot, after just one year of implementation for the majority of schools, this evaluation research is beginning to demonstrate the potential of *CoolThink@JC* for offering students an opportunity to learn computational thinking and offering teachers an opportunity to explore new models of instruction. The design team has already begun to address some of the challenges raised in this report, which is a goal of the pilot phase of any program. In a year, the final report from this study will seek to confirm the findings reported here, and provide more insight into the progression of learning over the course of the multi-year pilot.

References

Basu, S. (2019). Using rubrics integrating design and coding to assess middle school students' open-ended block-based programming projects. To appear in *Proceedings of 50th ACM Technical Symposium on Computer Science Education* (SIGCSE '19), February 27–March 2, 2019, Minneapolis, MN, USA.

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016). *Developing Computational Thinking in Compulsory Education – Implications for Policy and Practice.* European Commission, Joint Research Centre. EUR 28295EN, doi:10.2791/792158.

Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association,* Vancouver, Canada (Vol. 1, p. 25).

City, E., Elmore, R., Fairman, S., & Teitel, L. (2009). *Learning to see, unlearning to judge. In Instructional rounds in education: A network approach to improving teaching and learning* (pp. 83-98), Harvard Education Press, Cambridge, MA.

Cohen, D. K. and Ball, D. (2007). Educational innovation and the problem of scale. In Schneider, B., & McDonald, S. K. (Eds.) *Scale-up in education: Ideas in principle* (Vol. 1) (pp. 19-36.) Rowman & Littlefield Publishers.

Department for Education. (2013). *National Curriculum in England: Computing Programmes of Study.* https://www.gov.uk/government/ publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study

Education University of Hong Kong (2018). *CoolThink@JC:* Report on project assessment by teachers in 2016-17 and 2017-18. Hong Kong: The Education University of Hong Kong.

Elmore, R. (1996). *Getting to scale with good educational practice.* Harvard Educational Review, 66(1), 1- 27.

Embretson, S. E., & Reise, S. P. (2013). *Item response theory*. Psychology Press.

Fixsen, D. L., & Blase, K. A. (2008). *Drivers framework.* Chapel Hill, NC: University of North Carolina, The National Implementation Research Network, Frank Porter Graham Child Development Institute.

Fixsen, D. L., Blase, K. A., Duda, M., Naoom, S. F., & Van Dyke, M. (2010). Sustainability of evidence-based programs in education.*Journal of Evidence-Based Practices for Schools,11*(1), 30-46.

Fullan, M. (2012). *Change forces: Probing the depths of educational reform.* Routledge.

Kelly, B., & Perkins, D. F. (Eds.). (2012). *Handbook of implementation science for psychology in education.* Cambridge University Press.

Menekse, M. (2015) Computer science teacher professional development in the United States: A review of studies published between 2004 and 2014. *Computer Science Education, 25*:4, 325-350, DOI: 10.1080/08993408.2015.1111645

Mislevy, R. J., & Haertel, G. D. (2006). Implications of evidence-centered design for educational testing. *Educational Measurement: Issues and Practice, 25*(4), 6–20.

Mislevy, R. J., & Riconscente, M. M. (2006). *Evidence-centered assessment design: Layers, concepts, and terminology.* In S. Downing & T. Haladyna (Eds.), Handbook of test development (pp. 61-90). Mahwah, NJ: Lawrence Erlbaum.

Mislevy, R.J. (2007). Validity by design. *Educational Researcher, 36*(8), 463-469.

Parker, M. C., & DeLyser, L. A. (2017, March). Concepts and Practices: Designing and Developing A Modern K-12 CS Framework. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (pp. 453-458). ACM.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–36.

Yadav, A., Good, J., Voogt, J., & Fisser, P. (2017). Computational thinking as an emerging competence domain. In *Competence-based vocational and professional education* (pp. 1051-1067). Springer, Cham.

Yadav, A., Hambrusch, S., Korb, T., & Gretter, S. (2013). Professional development for CS teachers: A framework and its implementation. *Future directions in computing education summit.*

Appendices

The report will include the following appendices for more technical readers:

A. *CoolThink@JC* framework

B. Elaboration of our analytical methods

C. Model estimates from the impact analysis

D. Report on our ongoing validation analysis

E. Supporting details for figures included the narrative

Appendix A: *CoolThink@JC* Framework

01 CT Concepts: Fundamental Coding Concepts

< Sequences / > : identify a series of steps for a task

< Events / > : one thing causing another thing to happen

< Repetition / > : run the same sequence multiple times

< Conditionals / > : make decisions based on conditions

< Parallelism / > : make things happen at the same time

< Naming / > : name variables and functions descriptively to make them distinguishable from each other

< Operators / > : support for mathematical and logical expressions

< Manipulation of data and elementary data structures / > : basic ways data are formatted and stored as store, retrieve and update values

02 CT Practices: Problem-solving Skills

< Algorithmic thinking / > : articulate a problem's solution in well-defined rules and steps

< Abstracting and modularizing / > : explore connections between the whole and the parts

< Testing and debugging / > : make sure things work — and find and solve problems when they arise

< Being incremental and iterative / > : develop a little bit, then try it out, then develop more

< Reusing and remixing / > : make something by building on existing projects or ideas

03 CT Perspectives: Identity and Motivation

< Expressing / > : create and express ideas through this new medium

< Questioning / > : feel empowered to ask questions about and with technology

< Connecting / > : appreciate that others are engaged with – and appreciate – one's creations

< Digital empowerment / > : develop the ability to see problems in the world as solvable through code

< Computational identity / > : see oneself as being able to enhance the world through coding

Source: https://www.coolthink.hk/en/ct/

Appendix B: Analytical Methods

For a more technical audience, this appendix introduces several of the analytic methods used in this research.

Partial Matrix Sampling

A matrix sampling approach to assessment design involves distributing sets of items across multiple forms, and randomly assigning the forms to students. This process allows data to be collected on more items than can be administered to one student at a time. This approach is commonly used when the goal is to determine how cohorts of students are performing rather than to obtain individual scores for students. For example, this approach is used in the National Assessment for Educational Progress, an assessment that measures student achievement across the United States.

While this method does not allow for direct comparison of students, as different students will receive different items which may cover a different set of concepts, it is well tuned for comparisons of the performance of groups of students. A version of matrix sampling, referred to as *partial matrix sampling*, allows for some individual student comparison. In in this version there is a set of common items, or items that all students receive, and the rest of the items are split up across forms. Using this approach along with an item response theory analysis (described below) allows student scores to be generated that are comparable across students even if they do not take all of the same set of items. Specifically, student ability is estimated using item response models based the items they take. The items that they have in common are used to anchor their ability estimates so that the estimates are on the same scale, and thus comparable. We used the partial matrix sampling approach to develop the CT Concepts and CT Practices assessments. The benefit of this approach is that it allows for measurement of the cohort of students on all items while reducing the testing time for individual students.

Item Response Theory

The students' CT Concepts, Practices, and Perspectives scores were calculated based on Item Response Theory (IRT). IRT is a latent variable modeling approach by which scores on assessment items are used to place items on a scale indicating their difficulty, as well as to place students on the same scale indicating their ability. This method of analysis allows us to create an overall measure of computational thinking ability, and to look at the progression of an individual student or cohort of students along that continuum. IRT is tuned to handle missing data, which is important in matrix sampling because individual students will only respond to a subset of the total pool of test items or constructs on a given assessment. Student ability is estimated based on the student's available responses. The responses that are missing by design will not contribute to ability estimation. This allows us to generate an overall estimate of computational thinking ability for each student at each administration point.

With this design we can compare individual students' progress along the full continuum of computational thinking ability. Because matrix sampling randomly distributes items that measure individual constructs across a large number of students, we can also compare cohorts of students on each construct. It is important to recognize that this is different than designs that track individual students' learning of each specific construct over time.

In order to maintain comparability in estimates of item difficulty and student abilities across different assessments, we include common items (often referred to in IRT as anchor items) that help define the scale. Therefore, for the CT Concepts assessments we not only have common items across forms for the assessments administered at the same level, we also include items that are the same across levels. Using these anchor items, we are able to calibrate item characteristics such as difficulty and discrimination with different student samples, and to link assessments across forms and years. We can then see the variation in the ability estimates of the students based on one common continuum of the construct of interest. For CT Concepts with dichotomous items, we used a one-parameter logistic (1PL) model as we were concerned with the difficulty of the items. For CT Practices, items are either dichotomously scored or polytomously scored up to 5 points. We used a partial credit model (PCM) to account for the items with multiple scoring categories. For CT Perspectives where items were on a Likert-type scale, we used a rating scale model to account for the multiple response categories of the items. Additional details on these models and their uses can be found in Embretson & Reise (2013).

Hierarchical Linear Models for Impact Analysis

The year 1 impact estimates were derived from a three-level hierarchical model with student, classroom and school levels that controlled for covariates at their own levels. The model is shown below:

$$y_{cij} = \beta_0 + \beta_1(CoolThink_j) + \beta_m(m^{th} student\ covariate_{cij}) + \beta_k(k^{th} classroom\ covariate_{ij}) + \beta_l(l^{th} school\ covariate_j) + e_{cij} + r_{ij} + u_j$$

where c is students, i is classrooms, j is schools; Ycij is a student outcome; CoolThinkij equals 1 for CoolThink schools and 0 for comparison schools; and ecij and rij and uj are student, classroom and school random effects. β_1 is the estimated impact of *CoolThink@JC* on student outcomes. Statistical significance at a $p<0.05$ level would indicate a significant impact of *CoolThink@JC* on student outcomes.

For year 2 onward, because students would change classrooms along the way, we posited two-level HLM models with student and school levels for impact. The model is shown below:

$$y_{ij} = \beta_0 + \beta_1(CoolThink_{ij}) + \beta_k(k^{th} student\ covariate_{ij}) + \beta_l(l^{th} school\ covariate_j) + e_{ij} + r_j$$

where i is students, j is schools; Yij is a student outcome; CoolThinkij equals 1 for *CoolThink@JC* pilot schools and 0 for comparison schools; eij and rj are student and school random effects. β_k represents a vector of teacher covariates; β_1 is the estimated impact of *CoolThink@JC* on the student outcome.

Exhibit B1 lists the variables we adjusted at each level.

Exhibit B1. Control Variables.

Level	Control Variables
School	% of students using financial aid[a] % of students with especial needs[a] % of non-Chinese speakers[a] School prior coding instruction experience Paper vetting score Cohort indicator in combined cohorts analysis
Classroom	% of students using financial aid % of students with especial needs % of non-Chinese speakers
Student	Baseline outcome measure Grade level (4-6)

[a] *We have the same demographic variables at both classroom and school level. School demographic variables are only used in two-level analyses without the classroom level.*

Appendix C: Model Estimates

This appendix shows the model estimates, standard errors and p-values for the impact analysis.

CT Concepts

We examined the difference in student outcome scores between CoolThink@JC pilot and comparison schools at baseline to check whether the two groups were equivalent in student outcomes before the start of the intervention. Exhibit C1 shows the baseline concept scores and sample sizes for pilot and comparison schools respectively. The difference between the two groups is within 0.25 standard deviation, so this analysis achieved baseline equivalence.

Exhibit C2 shows the impact and subgroup differential impact estimates from a three-level HLM model with student, classroom and school levels. These results were discussed in the main text.

Exhibit C1. Baseline Student Concept Scores, Year 1 Impact Analysis.

	Mean	SD	Students	Classrooms	Schools
Pilot	0.020	0.657	4971	354	30
Comparison	-0.063	0.617	6592	340	24

Exhibit C2. Model Estimates for Student Concept Scores, Year 1 Impact Analysis.

	Estimate		SE	P-value
Overall impact				
CoolThink@JC pilot	0.213	**	0.080	0.008
Differential impact on subgroups				
Girls versus boys	-0.070	*	0.033	0.040
Grade 5 versus grade 4	-0.004		0.075	0.960
Grade 6 versus grade 4	-0.043		0.078	0.580
Baseline concept score	0.075	**	0.027	0.005
Standardized baseline math test score	0.067	***	0.019	0.000
Class % students with special needs	-1.004	*	0.436	0.021
Class % students using financial aid	-0.212		0.254	0.405
Class % non-Chinese speakers	0.013		0.251	0.959
Ever done programming	-0.031		0.038	0.412
Ever done programming at home	-0.020		0.041	0.619
Have internet at home	0.035		0.043	0.409
Computer use at home	0.008		0.010	0.431
Hours of *CoolThink@JC* instruction	-0.001		0.010	0.950
School existing coding experience	-0.014		0.014	0.316

$^*p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$

Exhibit C3 shows the baseline concept scores and sample sizes for pilot and comparison schools respectively. The difference between the two groups are within 0.25 standard deviation, therefore this analysis achieved baseline equivalence.

Exhibit C4 shows the impact estimates for Cohort 1 year 2 analysis from a two-level HLM model with student and school levels. This result was discussed in the main text.

Exhibit C3. Baseline Student Concept Scores, Cohort 1 Year 2 Impact Analysis.

	Mean	SD	Students	Schools
Pilot	-0.003	0.595	1160	10
Comparison	-0.060	0.572	1278	11

Exhibit C4. Model Estimates for Student Concept Scores, Cohort 1 Year 2 Impact.

	Estimate	SE	P-value
Overall impact			
CoolThink@JC pilot	0.156	0.201	0.400

CT Practices

For CT practices, we only included Cohort 2 schools because students were first assessed in September 2018, when Cohort 1 pilot schools had completed level 1, we therefore do not have baseline measure in CT Practices for Cohort 1 pilot schools.

Exhibit C5 shows baseline practices scores and sample sizes for pilot and comparison schools respectively. The difference between the two groups is within 0.25 standard deviation, so this analysis achieved baseline equivalence.

Exhibit C5. Baseline Student Practice Scores, Cohort 2 Year 1 Impact Analysis.

	Mean	SD	Students	Classrooms	Schools
Pilot	0.049	0.681	5279	260	20
Comparison	-0.022	0.639	3613	224	21

Exhibit C6 shows the impact estimates for Cohort 1 year 2 analysis from a two-level HLM model with student and school levels. This result was discussed in the main text.

Exhibit C6. Model Estimates for Student Practice Scores, Cohort 2 Year 1 Impact Analysis.

	Estimate	SE	P-value
Overall impact			
CoolThink@JC pilot	0.166	0.068	0.015

CT Perspectives

We examined the differences in student perspectives sub-construct scores between *CoolThink@JC* pilot and comparison schools at baseline to see whether the two groups were equivalent in student outcomes before the start of the intervention. Exhibit C7 presents the descriptive statistics for pilot and comparison groups respectively. Across all perspective sub-constructs, the mean differences between the two groups are smaller than .20 of the respective standard deviations. We deemed the pilot and comparison schools achieved equivalence at baseline.

Exhibit C7. Baseline Student Perspectives Scores, Year 1 Impact Analysis ($N_{school – pilot}$ = 30, $N_{class – pilot}$ = 372, $N_{school – comparison}$ = 24, $N_{class – comparison}$ = 356).

		Mean	SD	Students
Belong	Pilot	.26	.95	5760
	Comparison	.19	.90	4264
Interest in programming	Pilot	-.15	.87	5993
	Comparison	-.28	.84	4532
Engagement	Pilot	-.10	.93	5986
	Comparison	-.23	.92	4531
Motivation to help the world	Pilot	.20	1.24	5964
	Comparison	.05	1.19	4573
Creativity	Pilot	.24	1.16	5960
	Comparison	.06	1.14	4570
Self-efficacy	Pilot	.17	1.08	5955
	Comparison	.00	1.07	4568
Utility motivation	Pilot	-.12	1.03	5937
	Comparison	-.24	1.02	4560

Exhibits C8-C14 show the impact and subgroup differential impact estimates from a three-level HLM model with student, classroom, and school levels. The results were discussed in the main text.

Exhibit C8. Model Estimates for Student Perspectives Scores - Belonging, Year 1 Impact Analysis.

	Estimate		SE	P-value
Overall impact				
CoolThink@JC pilot	.06		.07	.342
Differential impact on subgroups				
Girls versus boys	.08	*	.04	.034
Grade 5 versus grade 4	-.05		.06	.384
Grade 6 versus grade 4	-.04		.06	.518
Baseline belonging score	-.01		.02	.514
Class % students with special needs	-.08		.38	.829
Class % students using financial aid	.01		.21	.955
Class % non-Chinese speakers	-.12		.21	.575
Ever done programming	.04		.04	.304
Ever done programming at home	.01		.05	.835
Have internet at home	.02		.05	.670
Computer use at home	.01		.01	.555
Hours of *CoolThink@JC* instruction	-.01		.01	.247
School existing coding experience	-.01		.01	.328

$^*p < .05$

Exhibit C9. Model Estimates for Student Perspectives Scores – Interest in Programming, Year 1 Impact Analysis.

	Estimate	SE	P-value
Overall impact			
CoolThink@JC pilot	.05	.05	.308
Differential impact on subgroups			
Girls versus boys	-.05	.03	.094
Grade 5 versus grade 4	-.03	.05	.523
Grade 6 versus grade 4	-.05	.05	.338
Baseline interest score	.02	.02	.213
Class % students with special needs	-.27	.31	.380
Class % students using financial aid	-.11	.17	.538
Class % non-Chinese speakers	-.07	.17	.670
Ever done programming	-.01	.03	.673
Ever done programming at home	-.02	.04	.592
Have internet at home	.04	.04	.273
Computer use at home	.00	.01	.854
Hours of *CoolThink@JC* instruction	-.01	.01	.109
School existing coding experience	.00	.01	.874

Exhibit C10. Model Estimates for Student Perspectives Scores – Engagement, Year 1 Impact Analysis.

	Estimate	SE	P-value
Overall impact			
CoolThink@JC pilot	.05	.06	.354
Differential impact on subgroups			
Girls versus boys	-.05	.03	.139
Grade 5 versus grade 4	-.03	.05	.591
Grade 6 versus grade 4	-.03	.05	.543
Baseline engagement score	.01	.02	.654
Class % students with special needs	-.02	.33	.942
Class % students using financial aid	-.16	.18	.390
Class % non-Chinese speakers	-.15	.18	.414
Ever done programming	-.03	.04	.366
Ever done programming at home	-.05	.04	.252
Have internet at home	.01	.04	.844
Computer use at home	-.01	.01	.361
Hours of *CoolThink@JC* instruction	-.01	.01	.054
School existing coding experience	-.01	.01	.311

Exhibit C11. Model Estimates for Student Perspectives Scores – Motivation to Help the World, Year 1 Impact Analysis.

	Estimate	SE	P-value
Overall impact			
CoolThink@JC pilot	.11	.07	.132
Differential impact on subgroups			
Girls versus boys	-.04	.05	.468
Grade 5 versus grade 4	.01	.07	.933
Grade 6 versus grade 4	-.06	.07	.391
Baseline motivation world score	.01	.02	.566
Class % students with special needs	.30	.46	.517
Class % students using financial aid	-.21	.25	.411
Class % non-Chinese speakers	-.19	.25	.448
Ever done programming	-.05	.05	.349
Ever done programming at home	-.04	.06	.488
Have internet at home	-.04	.06	.468
Computer use at home	.00	.01	.893
Hours of *CoolThink@JC* instruction	-.02	.01	.078
School existing coding experience	-.01	.01	.472

Exhibit C12. Model Estimates for Student Perspectives Scores – Creativity, Year 1 Impact Analysis.

	Estimate		SE	P-value
Overall impact				
CoolThink@JC pilot	.14		.08	.079
Differential impact on subgroups				
Girls versus boys	-.10	*	.05	.023
Grade 5 versus grade 4	-.07		.07	.296
Grade 6 versus grade 4	-.09		.07	.182
Baseline creativity score	.02		.02	.388
Class % students with special needs	-.11		.46	.806
Class % students using financial aid	-.35		.25	.166
Class % non-Chinese speakers	-.04		.25	.875
Ever done programming	.00		.05	.995
Ever done programming at home	.00		.05	.941
Have internet at home	-.03		.06	.615
Computer use at home	.00		.01	.739
Hours of *CoolThink@JC* instruction	-.02		.01	.079
School existing coding experience	-.01		.01	.443

*$p < .05$

Exhibit C13. Model Estimates for Student Perspectives Scores – Self-Efficacy, Year 1 Impact Analysis.

	Estimate		SE	P-value
Overall impact				
CoolThink@JC pilot	.10		.08	.173
Differential impact on subgroups				
Girls versus boys	-.11	*	.04	.010
Grade 5 versus grade 4	-.07		.06	.258
Grade 6 versus grade 4	-.05		.06	.463
Baseline creativity score	.02		.02	.257
Class % students with special needs	.23		.42	.588
Class % students using financial aid	-.27		.24	.250
Class % non-Chinese speakers	-.03		.24	.896
Ever done programming	-.01		.05	.811
Ever done programming at home	-.03		.05	.540
Have internet at home	.04		.05	.374
Computer use at home	.02		.01	.062
Hours of *CoolThink@JC* instruction	.00		.01	.602
School existing coding experience	-.02		.01	.103

*$p < .05$

Exhibit C14. Model Estimates for Student Perspectives Scores – Utility Motivation, Year 1 Impact Analysis.

	Estimate	SE	P-value
Overall impact			
CoolThink@JC pilot	.11	.06	.072
Differential impact on subgroups			
Girls versus boys	-.07	.04	.055
Grade 5 versus grade 4	-.06	.06	.301
Grade 6 versus grade 4	-.04	.06	.442
Baseline creativity score	.03	.02	.081
Class % students with special needs	.48	.36	.183
Class % students using financial aid	-.14	.20	.480
Class % non-Chinese speakers	-.04	.19	.849
Ever done programming	-.03	.04	.501
Ever done programming at home	.00	.04	.929
Have internet at home	.04	.05	.368
Computer use at home	.00	.01	.803
Hours of *CoolThink@JC* instruction	.00	.01	.555
School existing coding experience	-.01	.01	.292

Exhibit C15 presents the descriptive statistics of perspectives sub-constructs for pilot and comparison schools at baseline for cohort 1 year 2 analysis. Pilot schools have substantially higher baseline scores than comparison schools on five of the seven sub-constructs (*interest in programming, engagement, creativity, self-efficacy,* and *utility motivation*). The differences between the two groups range from .25 to .45 of the respective standard deviations. The two exceptions are belonging and motivation to help the world. For these two sub-constructs, the differences between pilot and comparison schools are within 20% of the standard deviations. For the sub-constructs with substantial baseline differences, we did not find differential impacts for students with various baseline levels of perspective scores as shown in Exhibits C8-C14.

Exhibit C15. Baseline Student Perspectives Scores, Cohort 1 Year 2 Impact Analysis ($N_{school-pilot} = 10$, $N_{school-comparison} = 11$).

		Mean	SD	Students
Belong	Pilot	-.01	.74	656
	Comparison	.03	.84	623
Interest in programming	Pilot	.18	.85	677
	Comparison	-.06	.93	662
Engagement	Pilot	.15	.89	676
	Comparison	-.08	.91	666
Motivation to help the world	Pilot	.10	.84	676
	Comparison	-.06	.87	697
Creativity	Pilot	.13	.83	676
	Comparison	-.12	.87	694
Self-efficacy	Pilot	.27	.92	674
	Comparison	-.12	.93	692
Utility motivation	Pilot	.14	.91	672
	Comparison	-.13	.95	693

Exhibit C16 presents the impact estimates for Cohort 1 year 2 analysis from a two-level HLM model with student and school levels. The results were discussed in the main text.

Exhibit C16. Model Estimates for Student Perspectives Scores, Cohort 1 Year 2 Impact Analysis.

	Estimate		SE	P-value
Overall impact (*CoolThink@JC* pilot)				
Belonging	-.10		.15	.511
Interest in programming	-.35	*	.16	.023
Engagement	-.30		.16	.068
Motivation to help the world	.00		.25	.990
Creativity	-.23		.23	.328
Self-efficacy	-.23		.18	.200
Utility motivation	-.18		.15	.253

**$p < .05$*

Appendix D: Validation Analysis

Test validity refers to the degree to which evidence and theory support the interpretations of test scores for the proposed uses. Contemporary validation analysis is considered as an ongoing process that is initiated at the beginning of assessment design and continues throughout development and implementation. This is particularly important in the case of the CT instruments being used in the impact portion of the evaluation. The types of uses and interpretations of the testing results we want to and can make can be different at the different phases of the evaluation (i.e., baseline, midline, and endline), as students experience a range of potential impact of the *CoolThink@JC* lessons at these various timepoints.

In the baseline report, we provided an overview of the preliminary types of validity evidence that support baseline interpretations of the CT Concepts Level 1, CT Practices, and CT Perspectives scores. The types of validity evidence included aspects of test content and internal structure. We described how well the tests represent the domain of interest using the Evidence-Centered Design (ECD) approach to support test content validity of all three CT instruments. We discussed the internal structure aspect of validity for CT Concepts and CT Perspectives by examining test reliabilities and factor structures. In this appendix, we present validity evidence collected from the second baseline administration and the midline administration and compare the results to evidence from the first baseline administration in order to build coherent validity arguments. The types of validity evidence presented include aspects of test content, internal structure, item validity, and external criterion.

CT Concepts

We piloted CT Concepts Level 1 and Level 2 tests with 146 students and 135 students respectively and performed statistical analysis to examine item quality. We selected items that cover the constructs of interest and with a range of difficulty levels to form the Level 1 and Level 2 tests. Between baseline and midline the lessons were modified to reduce the number of learning goals at each level. In doing so, the focus on the Procedures concept was minimized and it was decided to remove this concept from the assessment. With the reduction in learning goals covered on each level it was deemed that a smaller number of items would provide coverage of the core concepts. Therefore, the forms were reduced so that every child would receive the same set of items at each level of the lesson sequence, with a core set of items between repeated on the different assessments for the different levels.

The internal structure of CT Concepts Level 1 and Level 2 forms were examined for the baseline and midline administrations. We conducted reliability analysis and confirmatory factor analysis (CFA) to see if the items in a test form measure the one CT Concept construct stably and consistently. While there are 4-5 CT concepts being measured in the CT Concepts test, the number of items for each concept was small and our belief is that these concepts are related enough that we can consider this test as measuring just one construct (students' knowledge of CT concepts). Exhibit D1 presents coefficient alpha for the reliability analysis and fit indices for a one-factor CFA model for CT Concepts. The coefficient alpha for CT Concepts forms ranges from .40 to .63 across time and levels of the lesson sequence. The relatively low reliabilities at baseline were expected because students had not yet started, or had just started, learning the content. Since students were not expected to have a good grasp of the concepts, we expected that they would be doing a large amount of guessing, which can reduce the measured reliability. In addition, the test is intentionally designed to include a range of item difficulties, with some items containing content students would not have seen until a higher level of the lesson sequence. While this range can help us differentiate students, it also can reduce the reliability, as the way students interact with the items is not expected to be similar across all items. Therefore we deemed that the reliability is acceptable for this situation. This floor effect, due to a lack of student knowledge prior to *CoolThink@JC*, also led to the lack of meaningful covariance among items for CT Concepts at baseline. As learning progressed, coefficient alpha increased significantly at midline, despite the smaller numbers of items in tests.

Results from factor analysis show that the hypothesis of items measuring one CT Concepts construct is better supported by the midline data than at baseline, possibly due to the same floor effect. For both level 1 and level 2 at midline, the one-factor model shows acceptable fit to data as indicated by the fit indices. Examination of the common items across time shows better student performance and greater covariances among these items from baseline to midline, which we would expect as more students are able to answer some of the questions correctly.

We examined item validity by studying item difficulty, discrimination, and missing rate. In each form of CT Concepts, items exhibit a distribution of difficulty levels as intended. Items with more difficult features (for example, those that represented Level 2 content rather than Level 1, or items that contain more intricate code) were found to have a lower item correctness rate. The average correctness rate for CT Concept items increased from baseline (33% and 32%) to midline (50% and 44%), indicating the tests were set at appropriate difficulty levels. Exhibit D2 presents the percent of correct for the common items across the different test forms for CT Concepts. Except for one item (item q121), students had increased or stable performance from baseline to midline on all items. Item q121 is a particularly challenging item in that it requires the knowledge of two sub-constructs of repetition and data structures and algorithm. Further analysis of the item is needed

Exhibit D1. Summary of Reliability Analysis and Factor Analysis for CT Concepts Forms for Baseline and Midline Administrations.

CT Concepts Form	# of Students	# of Items	Coefficient Alpha*	Fit for a One-Factor Model**			
				Chi-Square	df	RMSEA	CFI
2017 Feb Level 1	*289*	*31*	*.40*	601.296***	434	.037	.519
2017 Jun/Sep Level 1	*439*	*31*	*.41*	684.076***	434	.036	.649
2018 Jun Level 1	*16594*	*16*	*.56*	7069.713***	104	.064	.890
2018 Jun Level 2	*2355*	*21*	*.63*	1627.836***	189	.057	.816

**One common accepted cutoff for good reliability is .70. However, this cutoff is greatly affected by test length, content domain, the number of constructs measured, and the intended use of a test.*

***Common cutoffs for CFA model fit indices: equal to or greater than .90 and .95 for CFI is deemed as good and excellent fit; equal to or smaller than 0.01, 0.05, and 0.08 for RMSEA is deemed as excellent, good, and mediocre fit.*

****p < .001*

to understand why it still had a low correctness rate. Lack of increase on items Q321 and Q312b also warrants further exploration, which will be supported by the ongoing calibration of items across tests that is performed after each administration.

Across the forms, the overall test discrimination increased from 0.33 and 0.34 (baseline) to 0.62 and 0.64 (midline) as estimated by a 1PL IRT model. The results indicated that the tests were gaining greater power in differentiating students with different ability levels as they participated in the pilot lessons.

Exhibit D2. Percent Correct for Common Items Across CT Concepts Forms, in Four Validation Schools at Baseline.

Sub-Contract	Level	Item	2017 Feb Level 1 (N=289, J=31	2017 Jun/Sep Level 1 (N=439, J=31)	2018 Jun Level 1 (N=16,594, J=16)	2018 Jun Level 2 (N=2,355, J=21)
Repetition	1	q111	37.50%	37.27%	48.97%	51.31%
	2	q121	17.58%	14.42%	-	15.78%
	2	q122	46.64%	44.55%	48.64%	-
Conditionals	1	q212	29.33%	28.17%	35.54%	39.91%
	1	q213	15.11%	14.82%	22.24%	-
	2	q221	26.24%	27.31%	35.32%	31.59%
Parallelism and Sequencing	1	q312a	37.94%	31.24%	40.06%	-
	1	q312b	59.71%	53.38%	54.55%	-
	2	q321	36.07%	38.55%	37.01%	38.31%
	2	q322	33.94%	31.85%	37.76%	38.02%
Data Structure and Algorithm	1	q411a	45.71%	49.88%	74.47%	62.45%
	1	q411b	44.93%	44.37%	67.68%	56.63%
	1	q411c	35.53%	36.38%	45.78%	43.30%

CT Practices

The design and development of CT Practices follows the ECD approach. Additionally, a set of items was also reviewed by having eight students talk out loud as they worked on the items. The students' comments were recorded and reviewed to determine if students are approaching the items as expected and to identify any difficulties interpreting the items. Revisions were made to clarify the items based on the feedback.

We conducted reliability analysis for CT Practices and the estimated coefficient alpha is .61. A one-factor model and a four-factor model were fit to data to examine the internal structure of the test. Exhibit D3 shows the factor analysis results for these two models. The one-factor model fit adequately to data, according to the commonly adopted fit criteria.
Fit of the four-factor model is comparable with the one-factor model. Correlations between the sub-constructs are high, ranging from .62 to 84. Such

Exhibit D3. Summary of Factor Analysis for CT Practices Validation Form for Midline Administration (*N*=4946, *J*=26).

Factor Model	Chi-Square	df	RMSEA	CFI
One-factor model	1211.147***	299	.025	.928
Four-factor model	1002.592***	293	.022	.944

$^{***}p < .001$

results indicate that the use of a one-factor model is appropriate for CT Practices and that the use of one score for the test is appropriate. Detailed examination of the reliability and the factor analysis revealed some individual items with low covariance with the rest of the items. Further examination deemed that there were constructs included in these items that were not included in other tasks which may explain this finding. We will continue to monitor the performance of these tasks.

Two alternative forms were purposely designed for the validation of CT Practices because of the large number of items included in this test. Students were randomly assigned to one of the two forms – form F and form G – which had the exact same set of items but differ in the item order. Exhibit D4 presents the percent of max possible scores for CT Practices items in different validation forms. The analysis found that item correctness rate and item missing rate (not shown) are related to the item's position in the test: the same item tended to have a lower correctness rate and a high missing rate if it showed up later in the test. This order effect led to the overall better performance on Form F as well as better reliability estimate and higher discrimination power than Form G. This finding is consistent with the earlier finding that students were speeding through the test, which would necessarily result in poor measures of correctness. Modifications to administration are being implemented in order to improve student motivation as they take this test. Item difficulties will be re-visited after the next administration to determine if these changes were effective.

Exhibit D4. Percent of Max Possible Scores for CT Practices Validation Forms for Midline Administration.

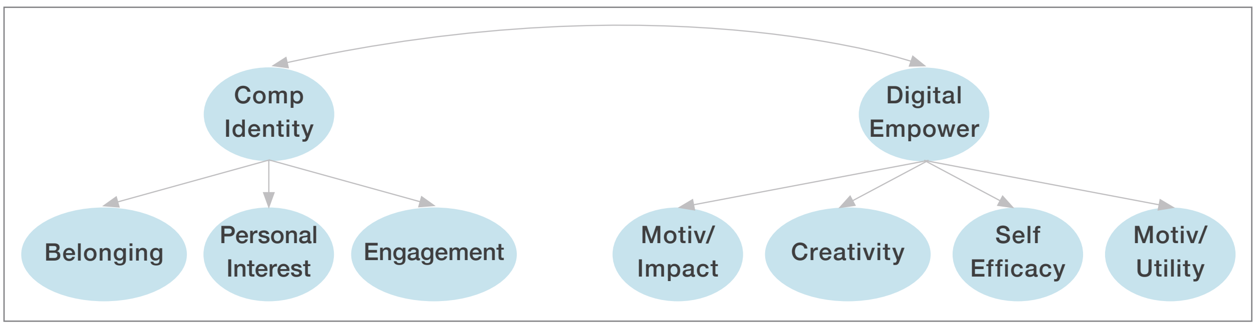
Sub-construct	Item		Max Possible Score	All Forms (N=4946, J=26)	Form F (N=2456, J=26)		Form G (N=2490, J=26)	
					Order	% Max	Order	% Max
Algorithmic Thinking	q11a	Robot moving with blocks	1	43.98%	1	49.21%	23	38.27%
	q11b		1	35.24%	2	42.17%	24	27.85%
	q11c		1	38.55%	3	44.07%	25	32.63%
	q11d		1	33.99%	4	36.53%	26	31.25%
	q12aa	Analyzing the maze	1	36.93%	16	37.08%	7	36.79%
	q12ab		1	27.79%	17	28.03%	8	27.56%
	q12ba		1	23.08%	18	23.96%	9	22.24%
	q12bb		1	24.86%	19	24.59%	10	25.10%
	q13	Analyzing the maze part 2	1	41.78%	20	40.23%	11	43.25%
Reusing and remixing	q215	Remaking a picture	1	40.81%	22	22.01%	1	44.20%
	q217		1	33.71%	23	37.01%	2	37.60%
	q21_mod		2	22.32%	24	29.32%	3	22.61%
	q22a	Making shapes	1	20.87%	6	20.03%	19	21.77%
	q22b		1	9.47%	7	12.40%	20	6.35%
	q22c		1	10.21%	8	9.24%	21	11.24%
Testing and Debugging	q32a	Painting a picture	1	12.43%	9	14.99%	17	9.76%
	q32b		4	15.11%	10	18.33%	18	11.82%
	q34	Calculator testing	1	21.10%	15	20.61%	12	21.56%
Abstraction and Modularizing	q42	Picking a picture	1	14.44%	5	14.85%	22	14.01%
	q43a	Wizard/Mandy story	2	51.46%	13	53.37%	13	49.52%
	q43b		2	22.79%	14	22.84%	14	22.75%
	q44a	Average of scores	1	20.85%	25	19.33%	4	22.20%
	q44b		1	7.70%	26	5.64%	5	9.53%
Testing and Debugging	q51	Robot making cakes	2	37.48%	21	41.03%	6	33.94%
Abstraction and Modularizing	q52a	Playground design	2	58.19%	11	58.07%	15	58.31%
Algorithmic Thinking	q52b	Playground design	5	75.49%	12	76.50%	16	74.47%

CT Perspectives

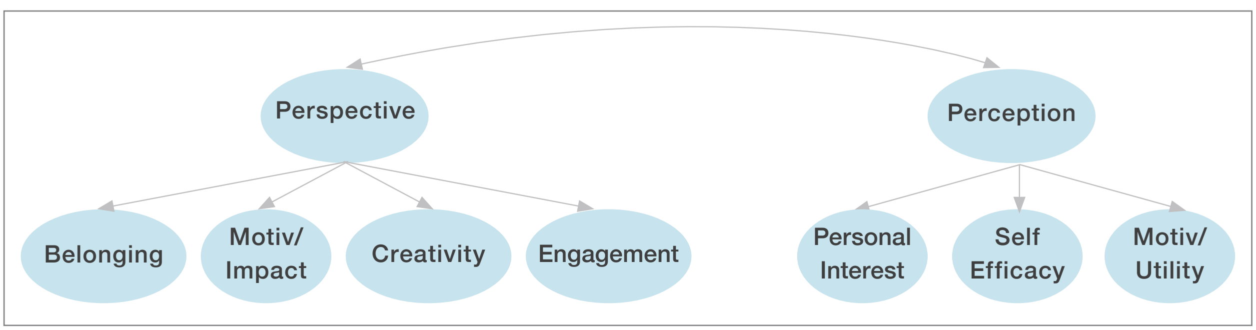
The development of CT Perspectives followed the steps of reviewing relevant literature, defining and developing sub-constructs, and creating and piloting items. The forms were piloted with 72 students and analysis was performed on these responses.

Different from CT Concepts and Practices, the evaluation of the internal structure for CT Perspectives aimed at examining whether items appropriately measure the seven sub-constructs and whether the relationship among the sub-constructs supports any of the proposed conceptual framework for CT Perspectives. Exhibit D5 presents the three different frameworks for conceptualizing the CT Perspective constructs.

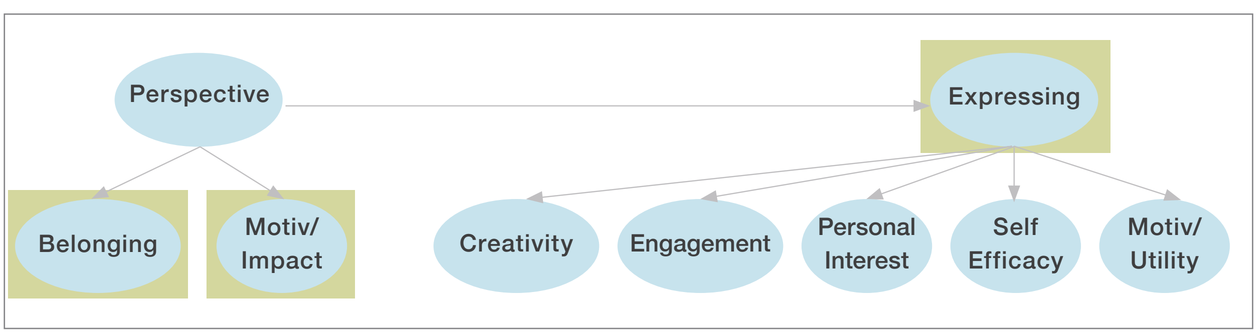
Exhibit D5. Conceptual Frameworks at the Higher-order of CT Perspectives Sub-constructs.



(a) Framework 1: Computational Identity versus Digital Empowerment



(b) Framework 2.1: CT Perspective versus Perception



(c) Framework 2.2: CT Perspective with Questioning, Connecting, and Expressing

We conducted a series of CFA models to address the hypotheses regarding the internal structure of CT Perspectives. These models include a seven-factor measurement model with all factors allowed to be freely correlated and three CFA models depicting the higher-order relationship among the sub-constructs following the frameworks 1, 2.1, and 2.2. Results of fitting these factor models are presented in Exhibit D6. At both baseline and midline, most of the items reliably measure their intended subconstructs. The exceptions were items that were reversely worded on purpose. Based on the model fit information, however, none of the three conceptual frameworks fit significantly better than the other one or better than the measurement model. Across time, all sub-constructs are highly correlated with each other, except for belonging. Exhibit D7 presents the correlations between the sub-constructs at midline as an illustration.

Exhibit D6. Summary of factor analysis for CT Perspectives for baseline and midline administrations.

Data	# of Students	Factor Model	Chi-Square	df	RMSEA	CFI	SRMR*
2017 Feb	323	Measurement model	624.152***	303	.057	.948	.049
		Conceptual framework 1	664.951***	316	.058	.943	.054
		Conceptual framework 2.1	703.599***	316	.062	.937	.053
		Conceptual framework 2.2	703.529***	316	.062	.937	.053
2018 Jun	19868	Measurement model	16092.576***	329	.049	.965	.042
		Conceptual framework 1	17768.168***	342	.051	.961	.045
		Conceptual framework 2.1	19936.148***	342	.054	.957	.044
		Conceptual framework 2.2	19954.205***	342	.054	.957	.044

** For SRMR, equal to or smaller than .08 is typically deemed as good model fit.*
****p < .001*

Exhibit D7. Correlations between sub-constructs of CT Perspectives for midline administration (*N*=19868).

	Belonging	Interest	Engagement	Motivation World	Creativity	Self-Efficacy
Interest	.47***					
Engagement	.47***	.92***				
Motivation World	.46***	.76***	.80***			
Creativity	.49***	.84***	.86***	.86***		
Self-Efficacy	.42***	.80***	.80***	.73***	.79***	
Utility Motivation	.45***	.81***	.83***	.83***	.85***	.84***

****p* < .001

SRI Education™

SRI Education, a division of SRI International, is tackling the most complex issues in education to identify trends, understand outcomes, and guide policy and practice. We work with federal and state agencies, school districts, foundations, nonprofit organizations, and businesses to provide research-based solutions to challenges posed by rapid social, technological and economic change. SRI International is a nonprofit research institute whose innovations have created new industries, extraordinary marketplace value, and lasting benefits to society.

Silicon Valley
(SRI International headquarters)
333 Ravenswood Avenue
Menlo Park, CA 94025
+1.650.859.2000
education@sri.com

Washington, D.C.
1100 Wilson Boulevard, Suite 2800
Arlington, VA 22209
+1.703.524.2053

www.sri.com/education

Stay Connected