Chapter 17
A Study of the Readiness of Implementing Computational Thinking in Compulsory Education in Taiwan

Ting-Chia Hsu

Abstract In recent years, Computational Thinking (CT) Education for K-12 students and undergraduates has become an important and hotly discussed issue. In Taiwan, starting from August 2019, all students in secondary schools will be required to be fostered with computational thinking competencies. This chapter discusses not only the preparation of students to learn CT but also the preparation required for teachers and principals. There will be six credits each for the compulsory education of information technology and living technology in junior high schools. Integrating CT into other courses, such as mathematics, is one of the approaches implemented at the primary school level. This chapter explores an initiative in which teachers integrated block-based programming into a mathematics course for the sixth-grade students, and further studied the self-efficacies and motivations of the students while they learn CT in the integrated course. The chapter also reports on investigations of the attitudes of educational leaders, such as the K-12 principals, towards teacher preparation for conducting CT education in their schools. The results of the study indicate that the weakest part of the object readiness (facilities) in 2017 in Taiwan was the availability of classrooms for maker activities from the perspectives of the K-12 principals. In terms of human resource readiness, instructional material resource readiness, and leadership support (management readiness), teachers and principals scored readiness degree at more than three points but less than four points on a five-point Likert scale, implying that there is still room in all these aspects to be enhanced. Many teacher training courses will need to be carried out in the next 1 to 2 years because the technological and pedagogical content knowledge of the teachers regarding CT education must continue to be strengthened.

Keywords Computational thinking · Self-efficacy · Learning motivation · Teacher education

---

T.-C. Hsu (✉)
National Taiwan Normal University, Taipei, Taiwan
e-mail: ckhsu@ntnu.edu.tw

© The Author(s) 2019

S.-C. Kong and H. Abelson (eds.), *Computational Thinking Education*,
https://doi.org/10.1007/978-981-13-6528-7_17

17.1 Introduction

Computational thinking refers to the basic concepts and processes used for solving problems in the computer science domain. The term was officially proposed in 2006 (Wing, 2006), and was later simplified into four phases for the curriculum design of CT in the United States (e.g., https://code.org/curriculum/course3/1/Teacher; http://cspathshala.org/2017/10/25/computational-thinking-curriculum/), the United Kingdom (e.g., https://www.bbc.co.uk/education/guides/zp92mp3/revision), India (e.g., https://www.nextgurukul.in/KnowledgeWorld/computer-masti/what-is-computational-thinking/), and so on (Fig. 17.1).

As shown in Fig. 17.1, the first phase of the CT process is to decompose the problem so that it can be analyzed and divided into several smaller subproblems. This is called the "problem decomposition" phase. The second phase is to identify the patterns in the data representation or data structure. In other words, if the students observe any repeated presentation of data or methods, they can identify their similarities, regularities, or commonalities. Therefore, they do not need to spend time repeating work when they write out the solution steps. The third phase is to generalize or abstract the principles or factors to become a formula or rule. The students have to try to model the patterns they found in the previous step. After testing, they

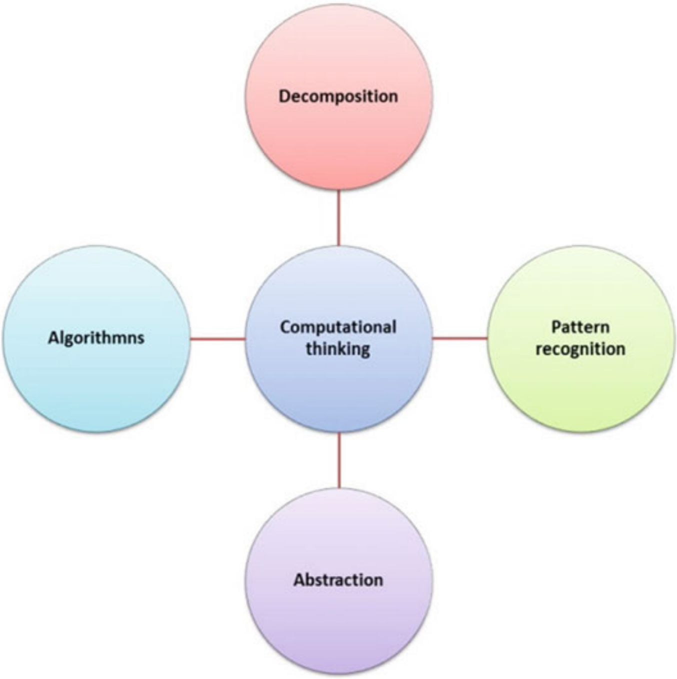

Fig. 17.1 CT process (cited from the BBC, UK)

identify and abstract the key or critical factors presenting the model for solving the problem in this step. Finally, they design the algorithm in the fourth phase, ensuring that they include all the steps for solving the problem systematically.

Although CT is not equal to programming, block-based programming languages such as Scratch, Blockly, mBlock, App Inventor, and so on, are good tools for developing the capabilities of students' CT. CT has been defined as "the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent" (Cuny, Snyder, & Wing, 2010). The current study not only employed Scratch to learn CT, but also used it to implement the solution to a problem that the students encountered in their mathematics course. Scratch or other visual programming tools are suitable to be used in different contexts such as games, science, music, and so on (Maloney, Resnick, Rusk, Silverman, & Eastmond, 2010; Armoni, Meerbaum-Salant, & Ben-Ari, 2015).

In a study by Maloney (2008), when Scratch was introduced to young students from 8 to 18 years old, the students were found to be highly motivated to write programs. Another study found that fifth and sixth graders perceived Scratch as being useful, and that they had high motivation and positive attitudes toward using it (Sáez-López, Román-González, & Vázquez-Cano, 2016). Ke (2014) applied Scratch for secondary school students to design mathematics games, and found that the integration of block-based programming and Mathematics game design could promote the potential of the students to learn Mathematics, and resulted in students' having significantly more positive attitudes toward the development of Mathematics. Furthermore, this method was beneficial for activating students' reflection on their daily-life mathematical experiences. The mathematics concepts and block-based programming were integrated when the students solved the problems or created the games. They not only took part in achieving the mathematics learning target, but also carried out CT, and transferred the reasoning process into an abstract program. It has been found that using block-based programming in computer science can promote the cognitive level and self-efficacy of students, but it does not result in high learning anxiety, and the students spend less time learning and creating new programs in comparison with line-based programming (Armoni et al., 2015).

The first study reported in this chapter integrated the block-based programming software, Scratch, into a mathematics course, and applied the four phases of CT to solve mathematics problems. The purpose of the study was to explore the correlations between self-efficacy and learning motivation, and between self-efficacy and creative tendency. From the results, the critical factor correlated with self-efficacy could be identified when the students were involved in the proposed treatments. In addition, this study also aimed to confirm whether the students made significant progress in Mathematics and in problem-solving by using block-based programming. Therefore, the research questions are as follows:

(1) Was the students' learning effectiveness of mathematics significantly promoted after the treatment?

(2) Was there a significant correlation between the performances of block-based programming with the learning effectiveness of mathematics?
(3) Was there a significant correlation between self-efficacy with creative tendency and learning motivation before and after the treatment?

Apart from exploring the effectiveness of such CT courses for K-12 students, the degree of preparedness of the teachers in teaching CT is another important issue. When CT becomes a necessary form of literacy all around the world, it will not only be a kind of expertise that, stereotypically, only computer engineers use. On the contrary, everyone should have positive attitudes toward CT in order to understand and make use of it (Wing, 2006).

Based on a survey of 17 European countries, in a previous study (Balanskat & Engelhardt, 2014), it was found that most of the countries have tried to integrate CT courses into their K-12 curricula. In addition, elementary and secondary schools in Australia have introduced CT into courses for a period of time, and have placed CT literacy in the national education curricula (Falkner, Vivian, & Falkner, 2014). Therefore, many teachers are now trying to integrate CT into various courses (Heintz, Mannila, & Färnqvist, 2016). With the current development of digital technologies and the concerns about CT literacy, how teacher education should prepare teachers to teach CT is an important question to be studied.

Recently, Orvalho (2017) indicated that teachers should follow the methodology for pre-service teachers: Before teaching students how to do CT, the teachers themselves should first acquire the knowledge and abilities related to CT. Yadav also pointed out that introducing computer science into courses for pre-service teachers can efficiently enhance teachers' understanding of CT. Therefore, the teacher can not only learn how to incorporate CT in their courses, but can also help the students cultivate their problem-solving capabilities (Yadav, Mayfield, Zhou, Hambrusch, & Korb, 2014). Mouza applied TPACK (i.e., Technology, Pedagogy, and Content Knowledge) instructional method to CT education, by having teachers designed CT courses associated with K-8 education during their teacher education. The results showed that the pre-service training not only had a positive influence on teachers, but could also help them to develop and practice instructional content embedded in CT (Mouza, Yang, Pan, Ozden, & Pollock, 2017).

As CT is applied to the training of not only teachers but also principals, they will all know what CT is, and how to integrate it into their courses, as well as the requirements of those courses. Many counties in Taiwan have asked newly appointed principals to enroll in training courses related to technology and leadership since 2011. It is expected that the principals know the requirements of the facilities and faculty in their schools for carrying out technology-related instruction, administration, and service. In the recent 2 years, they were made aware of the associated issues related to CT. Israel, Pearson, Tapia, Wherfel, and Reese (2015) applied CT to teacher education to overcome obstacles for teachers to achieve expertise in an Introduction to Computer Science course. K-12 faculty would realize what difficulties the students with deficient resources may encounter. Through the teacher education for pre-service teachers or newly appointed principals, they would benefit greatly and could know how to provide support and assistance to their teachers for enhancing CT education (Israel, Pearson, Tapia, Wherfel, & Reese, 2015).

In addition, when it comes to CT education, visual programming is a critical enabler. When the teachers design CT-related courses, they mostly use block-based programming tools for the basic level. Cetin (2016) considered CT to be the foundation, and applied Scratch to pre-service teachers' training. The results indicated that this did indeed help the teachers in arranging beginner courses, and the visual programming environment could help teachers better understand CT (Cetin, 2016).

The second study reported in this chapter applied the same approach in study one (i.e., visual programming for the mathematical learning unit) in the teacher training for newly appointed K-12 principals. After they experienced the demonstrations and training, we then investigated the readiness of their schools according to four dimensions: technology readiness, teacher readiness, instructional resource readiness, and leadership support. We also investigated the technology, pedagogy, and content knowledge (TPACK) and the overall TPACK related to CT education based on the real conditions the principals perceived. Therefore, the research questions for the second study are as follows:

(4) Concerning the principals who had experienced this course during their professional development training, how did they perceive the present readiness of their school for conducting such CT courses?
(5) Concerning the principals who had experienced this course during their professional development training, how did they perceive the present TPACK of the teachers in their school?

Overall, study one aimed to confirm the feasibility and effectiveness of conducting CT education in K-12 courses. Study two explored the readiness of the leadership in K-12 schools to implement and support CT education.

17.2 Method of Study One

As mentioned above, two studies are integrated into this chapter. The following section illustrates the research method including participant samples, measuring tools, and the experimental process, as well as the research results for study one.

17.2.1 Participants

For research questions one to three in study one, the subjects included one class of sixth graders of an elementary school in Taiwan. A total of 20 students participated in the study. They were taught by the same instructor who had taught that mathematics course and Scratch for more than 10 years. The average age of the students was 12.

17.2.2 Measuring Tools

The learning performance of CT includes three aspects which are concepts, perspectives, and practices (Brennan & Resnick, 2012). In study one, the research tools included the pre-test and post-test of the mathematics learning achievements, the post-test of Scratch Programming implementation, and the questionnaire for measuring the students' learning motivation, creative tendency, and self-efficacy.

The mathematics test sheets were developed by two experienced teachers. The pre-test consisted of 10 calculation questions about the prior knowledge of the course unit "equality axiom," with a perfect score of 100. The post-test consisted of 10 calculation questions for assessing the students' knowledge of the equality axiom unit, with a perfect score of 100. For instance, the following is an example for the elementary school students to practice mathematics and programming at the same time.

> On Sandy's birthday, her father, mother, and brother go to a theme park with her. They participate in a competition in which they have to guess the size of the facilities, which constitute a triangle. The host asks them to estimate the area of the triangle to get points by applying the block tools on a computer. If you were Sandy, how would you solve the problem using block-based programming to make automatic calculations?

Table 17.1 shows the answer of one student for the abovementioned problem to reveal an example of block-based programming and the CT process.

In the post-test of programming performance, there was a total of five situated problems for the students to solve using block-based programming according to the four phases of CT. Each programming problem was scored as 20 points, including five points for assessing whether the students employed proper blocks, five points for checking the usage of variances, five points for evaluating the formula transferred from the meaning of the problem by the students in the program, and five points for confirming if the output was correct or not. Consequently, the five programming problems were worth a total of 100 points.

The questionnaire of learning motivation was modified from the measure published by Hwang, Yang, and Wang (2013). It consisted of seven items (e.g., "It is

Table 17.1 Demonstration of the block-based programming and CT process

Example of block-based programming	Algorithmic thinking	CT process
	Starting the sequential steps	Define problem
		Pattern recognition
	Read the base from input	
	Read the height from the input	
	Use a formula to calculate	Abstraction
	Output the area of the triangle	

important for me to learn what is being taught in this class") with a 5-point rating scheme. The Cronbach's alpha value of the questionnaire was 0.823.

The self-efficacy questionnaire originates from the questionnaire developed by Pintrich, Smith, Garcia, and McKeachie (1991). It consists of eight items (e.g., "I'm confident I can understand the basic concepts taught in this course") with a five-point Likert rating scheme. The Cronbach's alpha value was 0.894.

The Creativity Assessment Packet (CAP) was revised from Williams (1991), and included the scales of imagination, curiosity, and so on. It consisted of 50 items (e.g., "I have a vivid imagination") with a 5-point rating scheme for the scales of overall creativity, curiosity, imagination, complexity, and risk taking.

17.2.3 Experimental Procedure

Before the experiment, the students were given time to get used to the block-based programming environment. Figure 17.2 shows the flow chart of the experiment. Each period in the mathematics class is 40 min in elementary school. At the beginning, the instructor spent 8 weeks (i.e., one period a week, and totally eight periods) teaching the students to become familiar with the block-based programming environment.

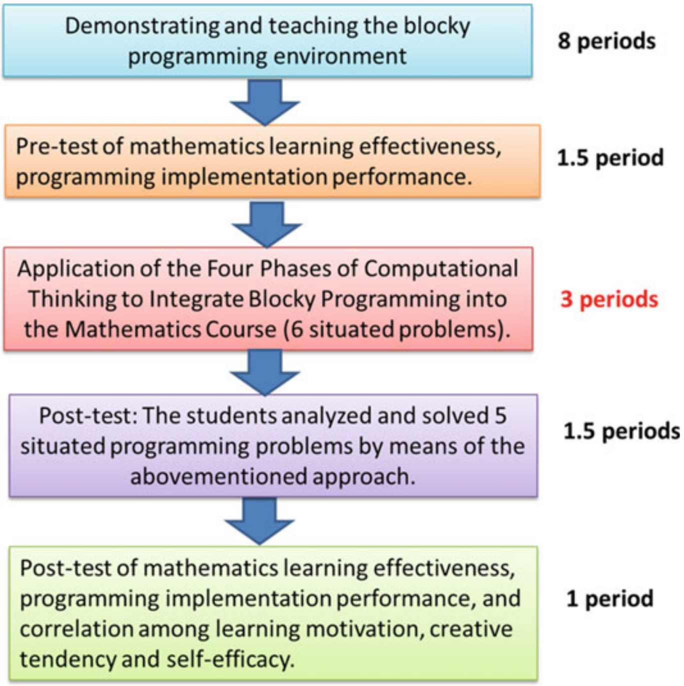

Fig. 17.2 Experimental procedure in study one

Before the learning activity of systematically applying the four phases of CT, the students completed the Creativity Assessment Packet measure, took the pre-test, and completed the learning motivation and self-efficacy questionnaires.

Thereafter, 3 weeks (i.e., one period per week, and totally three periods) was spent on the enhancement of applying the four phases of CT and the integration of block-based programming in a sixth-grade Mathematics course. After the effectiveness of involving the CT process with mathematics was confirmed in study one, this part (three periods) was later demonstrated in the teacher training course for the newly appointed principals to experience and observe the common sense of involving CT processes in learning. The students practiced this method six times, each time taking half a period. Therefore, there were totally six situated examples implemented during the three periods of the mathematics course.

At the same time, the students learned mathematics from solving the block-based programming problems through the four CT phases. After the learning activity, there were totally five programming problems for evaluating the students' block-based programming performance, with the four CT phases involved in both the block-based programming and the mathematics problems. The test took 1.5 periods.

Finally, they also spent one period on the post-test of the pen-and-paper-based mathematics test for measuring their learning achievements. There were totally 15 periods spent on the experiment, which lasted for a total of around three-fourth of a semester (i.e., 15 weeks). The experimental treatment after the pre-test was 5 weeks.

17.2.4 Data Analysis

In study one, the pre- and post-test were compared via a paired-sample *t* test to assess whether or not the students had made progress in the mathematics unit.

Their block-based programming performance was also assessed. Correlation analysis was performed to identify the relationship between the students' block-based programming performance and their post-test results.

Correlation analysis was utilized for checking the correlation among the students' learning motivation, self-efficacy, and creative tendency after the students learned mathematics from the application of the four phases of CT to integrate visual programming into the mathematics course.

Based on the above-mentioned data analysis methods, study one in this chapter reports the cognition and perspectives of the students involving CT processes in their mathematical learning.

Table 17.2 Paired sample *t* test on the pre- and post-test

	N	Mean	SD	t
Post-test	20	86.35	17.60	2.72^{*}
Pre-test	20	80.75	16.66	

$^{*}p < 0.05$

Table 17.3 Correlation between programming performance and the post-test (N = 20)

Pearson correlation coefficient	Block-based programming	Post-test
Block-based programming performance	1	0.673^{***}
Post-test	0.673^{***}	1

$^{***}p < 0.001$

17.3 Results of Study One

17.3.1 Paired-Sample t-Test Analysis of the Mathematics Pre- and Post-test

The research design hypothesized that the students would make progress in the learning objectives of the mathematics unit. Therefore, a paired-sample *t* test was performed on the pre-test and post-test in the mathematics unit.

The students did not use conventional instruction to learn mathematics; rather, the four phases of CT were applied to integrate block-based programming into the mathematics course. Table 17.2 reveals that this approach did indeed contribute to the students' learning effectiveness. They made significant progress in the mathematics equality axiom unit after the experimental treatment ($t = 2.72$, $p < 0.05$).

17.3.2 The Correlation Between the Block-Based Programming Performance and the Mathematics Post-test

In this study, we attempted to verify the correlation between the performance of block-based programming and the mathematics post-test. The results showed that they did have a significantly positive correlation (Pearson = 0.673, $p < 0.01$), as shown in Table 17.3. When the students had better performance on applying the four phases of CT to write a blocky program which solved the situated problems of the equality axiom mathematics unit, they also had better learning outcomes on the post-test of the conventional pen-and-paper-based mathematics test.

Table 17.4 Correlations between self-efficacy, creative tendency and learning motivation (N = 20)

Spearman correlation coefficient	Motivation	Self-efficacy	Creative tendency
Motivation	1	0.623^{**}	0.189
Self-efficacy	0.623^{**}	1	0.232
Creative tendency	0.189	0.232	1

$^{**}p < 0.01$

17.3.3 Correlations Between Students' Self-efficacy, Creative Tendency, and Learning Motivation

The self-efficacy of the students applying the four phases of CT to integrate block-based programming into the mathematics course was significantly correlated with their learning motivation (Spearman correlation value = 0.623, $p < 0.01$), but was not noticeably related to their creative tendency (Spearman correlation value = 0.232, $p > 0.05$), as shown in Table 17.4. In sum, the learning motivation was positively correlated with the students' self-efficacy regarding CT processes in their learning.

17.4 Method of Study Two

As mentioned above, two studies are integrated into this chapter. The following sections illustrate the participant samples, measuring tools, and the experimental process for study two.

17.4.1 Participants

For study two, there were 24 newly appointed principals who participated in the teacher training course. They were taught by the same instructor as study one in the teacher training workshop.

17.4.2 Measuring Tools

In study two, the participants had to answer the questionnaire of readiness for CT education at their school, and express the situation they perceived in the TPACK (i.e., technological, pedagogical, and content knowledge) of their teachers. There are eight scales in the questionnaire. The first four were revised from the readiness questionnaire of mobile learning (Yu, Liu, & Huang, 2016) which referred to an eclectic e-learning readiness including object readiness, software readiness, and leadership support (Darab & Montazer, 2011), and referred to the higher education m-learning readiness model based on the theory of planned behavior (TPB; Cheon, Lee, Crooks, & Song, 2012). Accordingly, object readiness, instructor readiness, instructional resource readiness, and leadership support are important scales for evaluating the readiness for putting something into practice at school, such as e-learning, mobile learning, or CT. Therefore, this study employed the readiness questionnaire, and the Cronbach's reliability coefficient for each scale in the revised questionnaire was .701 for object readiness, .673 for instructor readiness, .646 for instructional resource readiness, and .835 for leadership support.

The relationship between teachers' technological, pedagogical, and content knowledge (TPACK) is clearly pointed out in the framework of the TPACK model (Mishra & Koehler, 2006). Numerous studies have therefore adopted this model to assess teachers' professionality or the effectiveness of teacher education (Chai, Koh, & Tsai, 2010; Koehler, Mishra, & Yahya, 2007). This model has also been introduced in another study for teachers to perform self-assessment (Schmidt et al., 2009). The current study also employed the TPACK model (Chai et al., 2010) for the principals to describe the school teachers in the technology domain. The Cronbach's reliability coefficient for each scale in the revised questionnaire was .840 for the knowledge of technology, .884 for the knowledge of pedagogy, .943 for the knowledge of content, and .908 for the overall TPACK.

17.4.3 Experimental Procedure

After study one, the same instructor taught the CT course in the teacher training workshop for the newly appointed principals. The teacher training workshop consisted of 18 h. There were 9 h spent experiencing the CT process integrated with the mathematics unit through the tool of visual programming. During the remaining 9 h, they had to visit a school or institute where the infrastructure has been well established, and attend the training course introducing the requirements for conducting the 12-year compulsory education in the technology domain.

After they experienced the CT course and completed the teacher training, the principals filled out the questionnaires to assess their schools. One questionnaire was revised from the readiness for mobile learning, and the other one was the TPACK (i.e., Technological, Pedagogical, and Content Knowledge) model.

Table 17.5 Descriptive information for the first four scales: readiness

Scale name	Description	Sample item
Object readiness	For the current situation of equipment in the school, please answer the following questions	There is enough information equipment such as computers for learning in the school, providing resources for technological courses
Human resources readiness (instructor readiness)	For the current situation of teachers in your school, please answer the following questions	There are full-time information technology teachers in my school
Instructional resource readiness	For the arrangement of teaching materials for the technology domain, please answer the following questions	The teachers in my school have the capabilities to employ the official textbooks in the information technology courses
Leadership support (management readiness)	For the attitude of school management, please answer the following questions	School management proposes visions, policies, or plans that support and encourage the teaching as well as learning in the technological domain

17.4.4 *Data Analysis*

In study two, the descriptive information for the first four scales of readiness is shown in Table 17.5, which is abstracted from the first questionnaire (i.e., readiness).

The reliability data suggest that the refined version of each scale for readiness and TPACK has acceptable internal consistency. The investigation results show the descriptive statistics for each item, including the mean scores and the standard deviation.

17.5 Results of Study Two

17.5.1 *The Present Readiness for CT Education as Perceived by the Principals*

From the investigation results in Table 17.6, the average scores of object readiness are quite low. From each item shown in the questionnaire in Appendix 17.1, it could be found that the information equipment such as computers for learning in the school has been available for a period of time (Mean = 4.17, SD = 0.87). Therefore, the prin-

Table 17.6 School readiness for conducting compulsory education in the technology domain for each dimension (N = 24)

Scale	Mean	SD
Object readiness	3.00	0.82
Human resources readiness (instructor readiness)	3.28	0.93
Instructional material resource readiness	3.18	0.80
Leadership support (management readiness)	3.77	0.74

cipals expressed higher scores for the computer hardware in the first item. However, if the instruction requires equipment for hands-on activities, the maker classrooms are relatively lacking at the present time (Mean = 1.88; SD = 1.36). This is the main reason why the object readiness was reduced. In sum, the overall technology hardware and software for conducting compulsory education in the technology domain has not yet been well prepared as it is still 2 years before compulsory education in the technology domain begins.

As for instructor readiness, there are not enough full-time faculty in the technology domain according to the results of the human resources readiness scale, as shown in Table 17.6. The teacher education institutes must speed up the cultivation of new teachers, and the K-12 schools should open recruitment for teachers in the technology domain as their top priority.

In terms of instructional material resource readiness, the teachers tended not to take part in the teaching plan competitions. Therefore, holding teaching plan contests may not be the best strategy to produce adequate instructional material in the technology domain.

Finally, the leadership support was also taken into consideration for the readiness of conducting compulsory education in the technology domain. The scale of leadership support has the highest mean score among the four scales (Mean = 3.77; SD = 0.74). There is a strong tendency for the school leaders to put greater emphasis on students participating in DIY activities. In other words, school leadership tends to accept that teachers can design problem-solving tasks integrating different disciplines and hands-on activities in the future.

17.5.2 *The Teachers' TPACK for CT Education as Perceived by the Principals*

From the survey of TPACK for CT teachers, as shown in Table 17.7, it was found that the teachers are partially prepared at present (see also Appendix 17.2). The expertise of the present teachers was acceptable in terms of their knowledge of technology from the principals' point of view. However, the pedagogy of CT still has room for improvement. From the investigation results, the teacher education institutes have to put more effort into pedagogical research and training.

Table 17.7 TPACK for CT teachers for each scale (N = 24)

Scales	Mean	SD
Knowledge of technology	3.95	0.69
Knowledge of pedagogy	3.83	0.87
Knowledge of content	3.74	0.99
TPACK	3.58	0.97

17.6 Discussion and Conclusion

Study one not only put the four phases of CT into practice, but also applied them to solve mathematics problems with the block-based programming language. The results indicate that the implementation of programming activities was effective; in addition, the students' learning effectiveness, and their results in the mathematics concepts post-test both improved remarkably in comparison with the pre-test of the same mathematics unit. The programming implementation had a significantly positive correlation with the learning effectiveness of mathematics, implying that the students who had better block-based programming scores outperformed the other students in the mathematics concepts post-test.

In addition to the CT concepts and practices, the perspectives of the students were also assessed. Few studies have explored the relationships among self-efficacy, learning motivation, and creative tendency. The results of study one found that the students' self-efficacy was correlated with their learning motivation, but not with their creative tendency. In other words, the students who had higher learning motivation possessed higher self-efficacy. This result was similar to that of a previous study which pointed out that information literacy self-efficacy is associated with both intrinsic and extrinsic motivation (Ross, Perkins, & Bodey, 2016). Another study indicated that the creative tendency, such as curiosity as well as imagination, and domain-specific knowledge are critical for students' creative science problem-finding ability (Liu, Hu, Adey, Cheng, & Zhang, 2013). An earlier study reported that self-efficacy was closely related to creativity with intrinsic motivation completely mediating this relationship (Prabhu, Sutton, & Sauser, 2008). From study one, it has been confirmed that the students could learn CT and mathematics at the same time. In future studies, to promote the motivations of the students, the teachers could try to ask the students to design mathematics game programs with the block-based programming tools so that the students can also learn CT and mathematics at the same time. Future studies could also integrate CT into different subjects so that students can learn CT, programming, and subject knowledge (e.g., physics, mathematics) at the same time.

After confirming the feasibility and benefits of conducting CT in study one, study two further explored the readiness of the K-12 schools. Based on the results of this investigation on the K-12 principals in study two, some suggestions to enhance the preparation for involving CT education in the 12-year compulsory education are given as follows.

It appears that object readiness, which refers to the educational hardware of the technology domain at school, is the easiest part if the government is willing to devote sufficient resources to the K-12 schools. However, the teachers have to be trained so they know how to operate the new equipment, regardless of whether it is the maker environment or computer technology products; otherwise, the money spent on the hardware will be wasted. The perfect environment which is expected to be constructed within the next 2 years will not work without professional teachers. Therefore, future studies could further analyze the regression between the readiness of the teachers in the technology domain and the readiness of the hardware, and find direct evidence for this inference.

Unfortunately, the participants perceived that the leadership and management levels have not provided enough support for conducting CT education. In other words, many people agree that CT education is important; nevertheless, people in leadership roles have not put enough emphasis on it. We inferred that the reason for this unexpected situation is that the literacy of CT is not included as part of the senior high school or college entrance examinations. It is important that schools should not just pay attention to the subjects related to senior high school or college entrance examinations; liberal education should also be encouraged.

Study one was conducted in 2016, and study two was carried out in 2017, while the compulsory education of CT will be put into practice in August 2019. Accordingly, in the next 2 years, the related institutes have a large amount of work to do. The report of this chapter provides some findings, references, and suggestions for both K-12 school faculty and the Ministry of Education. The most important aspect is teacher education. The teachers in the technology domain should be trained in the requirements of instruction in the technology domain.

Acknowledgements This study is supported in part by the Ministry of Science and Technology in Taiwan under contract number: MOST 105-2628-S-003-002-MY3 and MOST 107-2511-H-003-031.

Appendix 17.1

School readiness for conducting compulsory education in the technology domain (N = 24)

Scale	Item	Mean	SD
Object readiness	1. There is enough information equipment such as computers for learning in the school, providing resources for technological courses	4.17	0.87
	2. Living Technology classrooms are set up, enabling complete living technology tutoring activities	2.46	1.38
	3. Maker classrooms are set up, giving students a proper environment to do maker activities	1.88	1.36
	4. There is availability of managers familiar with hardware and software, either full-time or outsourcing service	3.22	1.38
	5. I have confidence that the infrastructure in my school could put the 12-year compulsory education for the technological domain into practice	3.33	0.92
Human resources readiness (instructor readiness)	6. There are full-time information technology teachers in my school	3.29	1.71
	7. There are full-time living technology teachers in my school	2.54	1.56
	8. Our school provides workshops for teachers in the domain of technology to enhance their specialty	3.67	1.24
	9. The second specialty classes are available for other specialist teachers who want to be teachers in the technology domain	3.08	1.44
	10. The school is willing to arrange intramural and interschool activities for technology teachers to improve their capability	3.79	0.98
Instructional material resource readiness	11. The teachers in my school have the capabilities to employ the official textbooks in the information technology courses	3.29	1.33
	12. The teachers in my school have the capabilities to employ the official textbooks in the living technology courses	3.17	1.30
	13. The teachers in my school can develop instructional materials for school-based curricula on their own	3.17	1.17
	14. The teachers in my school are willing to take part in contests of making teaching plans for curricula	2.96	1.27

(continued)

(continued)

Scale	Item	Mean	SD
	15. Currently, the school teachers do not have to worry about the teaching materials for the technology domain	3.29	1.08
Leadership support (management readiness)	16. School management proposes visions, policies, or plans that support and encourage the teaching as well as learning in the technological domain	3.71	0.81
	17. The school has established a reward system for those who have outstanding teaching performance in technology	3.58	0.88
	18. School management is gradually putting greater emphasis on students participating in DIY activities and contests	3.92	0.83
	19. School management will encourage teachers and students to engage in a robot or programming competition if there is one	3.79	1.10
	20. School management will encourage teachers and students to engage in a living technology contest if there is one	3.83	1.09

Appendix 17.2

TPACK for Computational thinking teachers (N = 24)

Scales	Questionnaire items	Mean	SD
Knowledge of technology	TK1-Our teachers know how to solve their own technical problems	4.13	0.90
	TK2-Our teachers can learn new technology easily	4.04	0.81
	TK3-Our teachers have the technical skills and use the technologies appropriately	3.92	0.78
	TK4-Our teachers are able to use computational thinking tools or software to do problem-solving	3.71	0.86

(continued)

(continued)

Scales	Questionnaire items	Mean	SD
Knowledge of pedagogy	PK1-Our teachers can adapt their teaching style to different learners	3.83	1.09
	PK2-Our teachers can adapt their teaching based upon what students currently do or do not understand	3.79	1.02
	PK3-Our teachers can use a wide range of teaching approaches in a classroom setting (collaborative learning, direct instruction, inquiry learning, problem/project based learning, etc.)	3.75	1.03
	PK4-Our teachers know how to assess student performance in a classroom	3.96	0.91
Knowledge of content	CK1-Our teachers have various ways and strategies of developing their understanding of computational thinking	3.67	1.05
	CK2-Our teachers can think about the subject matter like an expert who specializes in computational thinking	3.79	1.06
	CK3-Our teachers have sufficient knowledge of computational thinking	3.75	1.03
TPACK	TPACK1-Our teachers can teach lessons that appropriately combine computational thinking, technologies, and teaching approaches	3.75	1.11
	TPACK2-Our teachers can use strategies that combine content, technologies, and teaching approaches	3.46	1.14
	TPACK3-Our teachers can select technologies to use in the classroom that enhance what they teach, how they teach, and what students learn	3.58	1.18
	TPACK4-Our teachers can provide leadership in helping others to coordinate the use of content, technologies, and teaching approaches at my school	3.54	0.93

References

Armoni, M., Meerbaum-Salant, O., & Ben-Ari, M. (2015). From scratch to “real” programming. *ACM Transactions on Computing Education (TOCE), 14*(4), 25.

Balanskat, A., & Engelhardt, K. (2014). *Computing our future: Computer programming and coding-Priorities, school curricula and initiatives across Europe*: European Schoolnet.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In: *Proceedings of the 2012 Annual Meeting of the American Educational Research Association* (pp. 1–25). Vancouver, Canada.

Cetin, I. (2016). Preservice teachers' introduction to computing: exploring utilization of scratch. *Journal of Educational Computing Research, 54*(7), 997–1021.

Chai, C. S., Koh, J. H. L., & Tsai, C. C. (2010). Facilitating preservice teachers' development of technological, pedagogical, and content knowledge (TPACK). *Educational Technology & Society, 13*(4), 63–73.

Cheon, J., Lee, S., Crooks, S. M., & Song, J. (2012). An investigation of mobile learning readiness in higher education based on the theory of planned behavior. *Computers & Education, 59*(3), 1054–1064. https://doi.org/10.1016/j.compedu.2012.04.015.

Cuny, J., Snyder, L., & Wing, J. M. (2010). *Demystifying computational thinking for noncomputer scientists*. Unpublished manuscript in progress, referenced in http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf.

Darab, B., & Montazer, G. (2011). An eclectic model for assessing e-learning readiness in the Iranian universities. *Computers & Education, 56*(3), 900–910. https://doi.org/10.1016/j.compedu.2010.11.002.

Falkner, K., Vivian, R., & Falkner, N. (2014). The Australian digital technologies curriculum: Challenge and opportunity. In: *Paper presented at the Proceedings of the Sixteenth Australasian Computing Education Conference* (Vol. 148).

Heintz, F., Mannila, L., & Färnqvist, T. (2016). A review of models for introducing computational thinking, computer science and computing in K-12 education. In: *Paper Presented at the Frontiers in Education Conference (FIE)*: IEEE.

Hwang, G. J., Yang, L. H., & Wang, S. Y. (2013). A concept map-embedded educational computer game for improving students' learning performance in natural science courses. *Computers & Education, 69,* 121–130.

Israel, M., Pearson, J. N., Tapia, T., Wherfel, Q. M., & Reese, G. (2015). Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. *Computers & Education, 82,* 263–279.

Ke, F. (2014). An implementation of design-based learning through creating educational computer games: A case study on mathematics learning during design and computing. *Computers & Education, 73,* 26–39.

Koehler, M. J., Mishra, P., & Yahya, K. (2007). Tracing the development of teacher knowledge in a design seminar: Integrating content, pedagogy and technology. *Computers & Education, 49*(3), 740–762.

Liu, M., Hu, W., Adey, P., Cheng, L., & Zhang, X. (2013). The impact of creative tendency, academic performance, and self-concept on creative science problem-finding. *PsyCh Journal, 2*(1), 39–47.

Maloney, J. H., Peppler, K., Kafai, Y., Resnick, M., & Rusk, N. (2008). Programming by choice: urban youth learning programming with scratch. *ACM, 40*(1), 367–371.

Maloney, J., Resnick, M., Rusk, N., Silverman, B., & Eastmond, E. (2010). The scratch programming language and environment. *ACM Transactions on Computing Education (TOCE), 10*(4), 16.

Mishra, P., & Koehler, M. J. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. *Teachers College Record, 108*(6), 1017–1054.

Mouza, C., Yang, H., Pan, Y.-C., Ozden, S. Y., & Pollock, L. (2017). Resetting educational technology coursework for pre-service teachers: A computational thinking approach to the development of technological pedagogical content knowledge (TPACK). *Australasian Journal of Educational Technology, 33*(3).

Orvalho, J. (2017). *Computational Thinking for Teacher Education.* Paper presented at the Scratch2017BDX: Opening, Inspiring, Connecting.

Pintrich, P. R., Smith, D. A. F., Garcia, T., & McKeachie, W. J. (1991). *A manual for the use of the motivated strategies for learning questionnaire (MSLQ).* MI: National Center for Research to Improve Postsecondary Teaching and Learning. (ERIC Document Reproduction Service No. ED 338122).

Prabhu, V., Sutton, C., & Sauser, W. (2008). Creativity and certain personality traits: Understanding the mediating effect of intrinsic motivation. *Creativity Research Journal, 20*(1), 53–66.

Ross, M., Perkins, H., & Bodey, K. (2016). Academic motivation and information literacy self-efficacy: The importance of a simple desire to know. *Library & Information Science Research, 38*(1), 2–9.

Sáez-López, J. M., Román-González, M., & Vázquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using "Scratch" in five schools. *Computers & Education, 97,* 129–141.

Schmidt, D. A., Baran, E., Thompson, A. D., Mishra, P., Koehler, M. J., & Shin, T. S. (2009). Technological pedagogical content knowledge (TPCK): The development and validation of an assessment instrument for preservice teachers. *Journal of Research on Technology in Education, 42*(2), 27.

Williams, F. E. (1991). *Creativity assessment packet: Test manual*. Austin, TX: Pro-Ed.

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

Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J. T. (2014). Computational thinking in elementary and secondary teacher education. *ACM Transactions on Computing Education (TOCE), 14*(1), 5.

Yu, Y.-T., Liu, Y.-C., & Huang, T.-H. (2016). Support-object-personnel mobile-learning readiness model for primary and secondary schools. *Journal of Research in Education Sciences, 61*(4), 89–120.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.