Informatics in Education, 2022, Vol. 21, No. 2, 311–329

DOI: 10.15388/infedu.2022.09

Pre-Service Teacher Training Program of STEM-based Activities in Computing Science to Develop Computational Thinking

Wichaya PEWKAM, Suthida CHAMRAT
Faculty of Education, Chiang Mai University, Chiang Mai, Thailand
e-mail: wichaya.pe@cmu.ac.th, suthida.c@cmu.ac.th

Received: March 2021

Abstract. Computing science which focuses on computational thinking, has been a compulsory subject in the Thai science curriculum since 2018. This study is an initial program to explore how and to what extend computing science that focused on STEM education learning approach can develop pre-service teachers' computational thinking. The online STEM-based activity-Computing Science Teacher Training (CSTT) Program was developed into a two-day course. The computational thinking test (CTT) data indicated pre-service teachers' fundamental skills of computational thinking: decomposition, algorithms, pattern recognition, pattern generalization and abstractions. The post-test mean score was higher than the pre-test mean score from 9.27 to 10.9 or 13.58 percentage change. The content analysis indicated that there were five key characteristics founded in the online training program comprised: (1) technical support such as online meeting program, equipment, trainer ICT skills (2) learning management system such as Google Classroom, creating classroom section in code.org (3) the link among policy, curriculum and implementation (4) pre-service teachers' participation and (5) rigor and relevance of how to integrate the applications of computing science into the classroom.

Keywords: computational thinking, computing science STEM activity.

1. Introduction

Computing science became a compulsory subject in the Thai science curriculum in 2018 (Ministry of Education, 2017). Starting from Grade 1, 4, 7 and 9 in 2018, grade 2, 5, 8, 10 in 2019, cumulatively. By 2020, all computing science will be covered all grade from 1–12. This means that every student who attends a Thai school must learn computing science. There are three main strands for computing science: Computer Science (CS), Information and Communication Technology (ICT), and Digital Literacy (DL) (The Institute for the Promotion of Teaching Science and Technology, 2018). This educational movement is a big change in Thailand. Computing science has been considered as the factor for digital economy driven according to Thailand 4.0 policy (Measincee, 2016). Furthermore, the next generation who were born between 2010–2025, will be the first group who will accomplish this compulsory computing science curriculum. They will grow up and become the majority of human resources in the workforces for Thailand for the next 20 years. If there is a group that affected and to be affected by Thailand's 20-Year National Strategy (Royal Thai Government Gazette, 2017). This generation is the very important as they will be the game changers for Thai economic driven. The mechanism of education during their school time will be the game changer for them as well as Thailand.

Not only one factor but education providing to transformative learning that changes the ways of thinking and living their lives. Under the current condition of Gross Domestic Product (GDP) in Thailand that expanded around 3.5 %. If we can keep this current GDP growth rate, within 30 years, Thailand will become a developed country (Wongsithuwiset and Jaroonpipatkul, 2017). This paper also offers an interesting and desirable alternative pathway. If Thailand can increase GDP to 5%, the country will be a developed country within 20 years, which is the goal of Thailand's 20-Year National Strategy. To become a country as developed, the preparation of human resources is significant. The way we teach in schools, need to be changed, as the new curriculum has been introduced and implemented in the classrooms. Computing science is one of those changes because it delivered both basic knowledge and skillsets that enable persons to live in the 21st-century world (Yadav, Gretter, Good & Mclean, 2017). For example, coding, one of the components of CS, is becoming a required skill in an increasing number of national curricula (in countries including the United Kingdom, Israel, Estonia, Finland and Japan). Computing science also links to Computational thinking (CT), which refers to a collection of computational ideas and habits of mind that people in computing disciplines acquire through their work in designing programs, software, simulations, and computations performed by machinery (Tedre and Denning, 2016). According to Wing (2006), CT has the following characteristics: (1) conceptualizing, not programming, (2) fundamental, not rote skill (3) ways that humans, not computers, think (4) complements and combines mathematical and engineering thinking (5) ideas, not artifacts and (6) for everyone, everywhere. Both computing science and computational thinking are the needed skills for both economic driven and personal thinking development as a cognitive tool. From economic point of view, a better educated workforce is an essential element of an internationally competitive workforce, and computational thinking is an essential component of such an education. (National Research Council, 2010).

In summary, this study applied the framework of CS as an integrative subject with STEM as the context of activity to develop pre-service teachers' CT. In this study, we viewed CT as the analytical ability and a problem-solving process with a unique set of features in science, philosophy, and other areas of knowledge rather than programming skills. For example, the innovative and creative thinking or "thinking outside the box" (Wichaya *et al.*, 2019) using STEM concepts and practices to solve a complex problem are kinds of computational thinking. However, CT competencies are not explicitly included in the curricula of many higher education degrees those future teachers of primary and secondary education have to complete (Dodero Beardo *et al.*, 2017). Computational thinking will become commonplace for teacher education via blended-computing science courses or even specific computing science education programs. To do this, we need an initial program to explore how and to what extend computing science that focused on STEM education learning approach can develop pre-service teachers' computational thinking. The development of teacher training program of STEM-based activities in computing science will be the model for teacher professional development that can apply to the future training or teacher education program. The results from investigations of pre-service teachers' computational thinking will enable university instructors, teacher supervisors and stakeholders of teacher education to think and rethink pre-service teachers' competency to design, develop lessons, and teach computing science in the future computing science. The authors of this paper believe that teacher preparation processes, such as those planned in this research, are the one of the keys to the success of computing education in Thailand and meeting Thailand's 20-Year National Strategy.

2. Methodology

2.1. *Research Participants*

Undergraduate students (elementary pre-service teacher n = 20 and science pre-service teachers n = 10, 30 in total) from the faculty of education, Chiang Mai University, had participated in this study. They have all volunteered to join the project. They were informed that their online contributions would be recorded using video conferencing software. The interaction of research participants and interventions strictly followed Human Research Ethics undertaken with the approval of Chiang Mai University Ethics Committees, COA No.033/63, CMUREC No. 63/110.

2.2. *STEM-based Activity-Computing Science Teacher Training (CSTT) Program*

The two-day program involved the design and development of STEM-based activity, integrated with concepts and practices of computing science. Since pre-service teachers who enrolled in this CSTT are not future computer scientists or computer science teachers, the activity was designed to reflect the ideas and frameworks of STEM education. Context relevance and real-world applications were the main focus of the activities. The design followed Chamrat (2017) key characteristics of STEM Education learning activity consisting of six features:

(1) The integration of science, technology, engineering and mathematics, in order to address STEM literacy.
(2) Designing learning activities based on learning progression framework.
(3) Providing context-based learning.
(4) Emphasizing 21st-century learning.

(5) Organizing designing and problem-solving activities.
(6) Employing authentic and formative assessment.

The definition and practices of STEM education, including its key features of the learning activities, will blend into the development of five units of CSTT. The activity of the programs adapted from the Project of Promoting Coding Skills toward Digital Era in the Future (Digital Economy Promotion Agency, 2019a, 2019b). The researchers designed and edited the activity from a face-to-face format into an online format via Zoom meeting (as shown in Fig. 1).

The teacher training program consisted of five units started from a whole-view perspective regarding national policies then shifted to national education policy. After that, the program narrowed down to the classroom levels and focused on applying computing science with real-life projects. The details of each unit are as follows:

Unit 1 Computing Science: Rationales and significances (Thailand 4.0, Thailand's 20-Year National Strategy, digital disruptions).

Unit 2 Introduction to the new science curriculum (Indicators and concepts in a science subject (revision B.E.2560) according to basic education core curriculum B.E.2551.

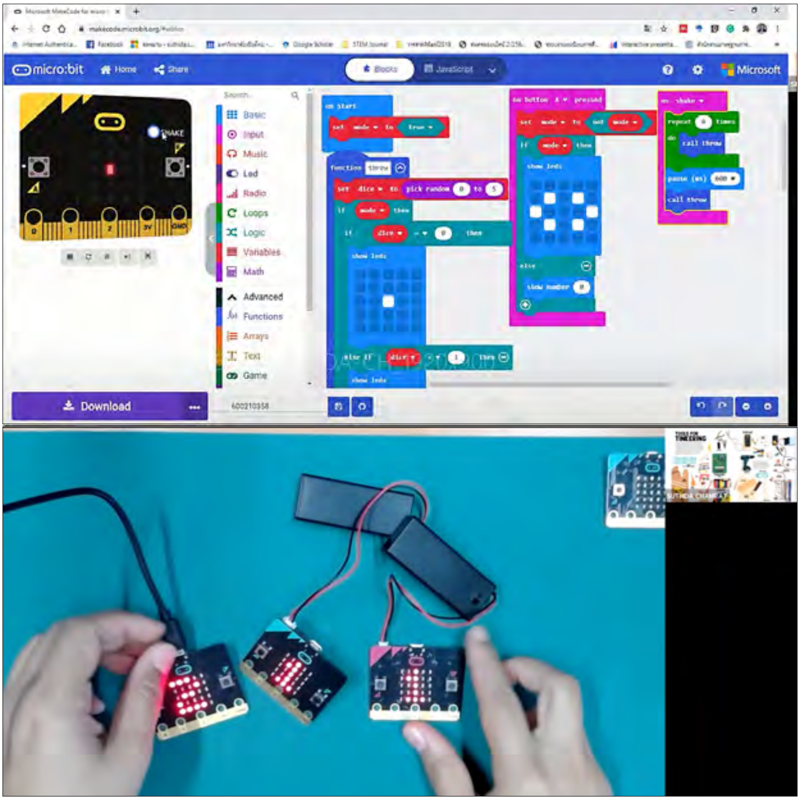


Fig. 1. Screen capture of the online teacher training program in this study via Zoom meeting.

Fig 2. The example project of coding microcontroller integration into chemistry for secondary school.

Unit 3 Computing science for elementary schools (Coding.org, codingthailand.org, unplugged activity, plugged-in activity) as shown in Fig. 1.

Unit 4 Computing science for secondary schools (Plugged-in activity, coding micro-controller) as shown in Fig. 2.

Unit 5 The application of computing science in a learning activity (Think outside the box, microcontroller project).

2.3. *Data Collection*

2.3.1. *Qualitative Data Collection*

Pre-services teachers (n = 30) who participated in this program took the computation thinking test before and after the two-day online teacher training program. The interview was used to collect supplementary data after participating in the STEM-based activity-Computing Science Teacher Training (CSTT) program. Video recordings throughout the CSTT program were used as supplementary research data. There were total 12 hours. Pre-service teachers' output from the training, such as graph paper programming, *.hex file from makecode.org, activity recording from code.org section was also considered as supplementary data.

2.3.2. *Quantitative Data Collection*

Pre-service teachers' computational thinking was measured by the computational thinking test or CTT, which was modified from Daungjun (2018). The test comprised four types of question assessing the pillars of computational thinking: decomposition, pattern recognition, abstraction and algorithm. Each test had four levels of scoring 0–3. The lowest point is 0, mean explanation shows very limited understanding of the under-

Table 1
Rubric for the assessment and scoring of computational thinking

Item	Scoring		
	3	2	1
1	Consider the route into subsections for each pair to calculate the correct time for all pairs. And indicate the time spent doing activities in each pair of routes	Consider the route into subsections for each pair to calculate the correct time for all pairs, without specifying the time spent on activities in each pair of routes	Consider the path into subsections for each pair to correctly calculate the time spent, less than or equal to 3 pairs
2	Specify three correct routes for doing activities according to the conditions of problems	Specify two correct routes for doing activities according to the conditions of problems	Specify only one correct route for doing activities according to the conditions of problems
3	The map can be drawn by identifying locations consistent with reality, identifying distances between two places and showing the conditional route sequence.	Can draft a map by identifying the corresponding location. The distance between the two locations was specified, but the order of the route did not meet the conditions.	Can outline the map but identifying the location is inconsistent with reality. There is no specified distance between the two locations. The path order given does not meet the conditions
4	Write down steps for solving problems in a precise sequence. And complete procedures leading to correct problem-solving	Write down steps for solving problems in a precise sequence. But incomplete steps may result in incorrect problem-solving.	Write down some troubleshooting steps and incomplete procedures, resulting in inaccurate solutions

lying concepts needed to solve the problem or was not written (excluded from Table 1). The 1 point indicated incomplete answer or the inability to solve the problem correctly. A score of 2 points indicated the right answer but with some points missing that needed to be used in problem solving. The full score of 3 indicated that the test taker could complete the test with comprehensive problem-solving. Details of rubric scoring from point 1–3 showed in Table 1.

2.4. *Data Analysis*

2.4.1. *Qualitative Data Analysis*

Qualitative data from video recording was uploaded into Atlas.ti for content analysis. The theme emerged from code and family code grounded into the key characteristics from the STEM-based activity-Computing Science Teacher Training (CSTT) online training program. The results from the qualitative data were used as supplementary data in the result and discussion session.

2.4.2. *Quantitative Data Analysis*

The answers from 30 pre-service teachers were checked and scored using three scales rubric. The response for each item was scored 1–3 according to the level of performance in relation to the descriptions of decomposition (Item 1), pattern recognition

(Item 2), abstraction (Item 3) and algorithm (Item 4), presented in Table 1. The data were analyzed for their descriptive statistics. The results presented in Table 2 are for average score, minimum score, maximum score and standard deviation. The pre-post comparison in terms of percentages was used to identify the progression of pre-service teachers' computational thinking skills before and after participating in this online teacher training program.

The overview procedure of research can be summarized in Fig. 3.

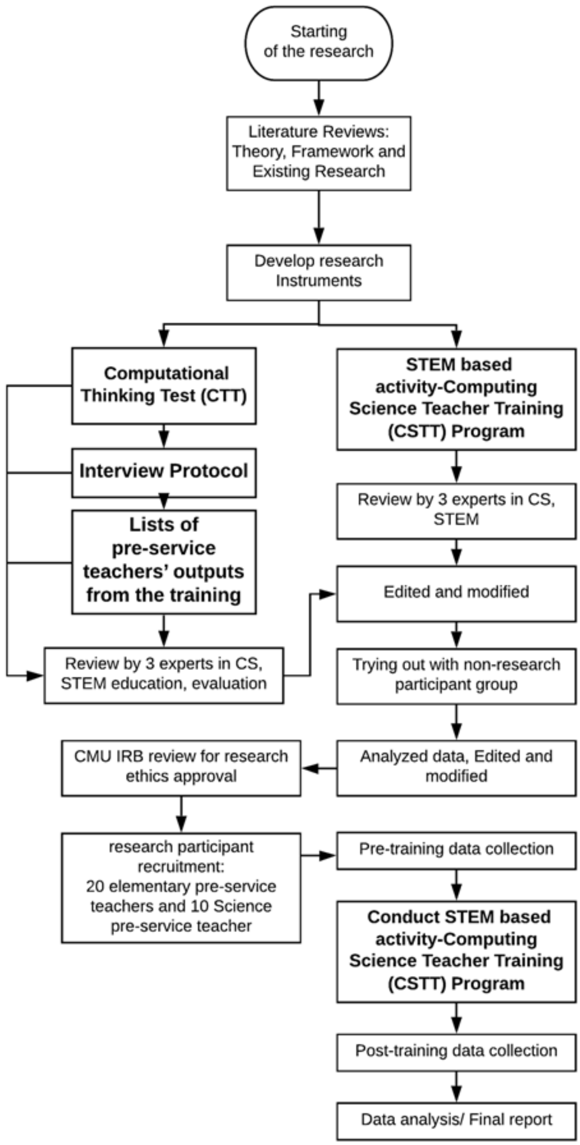


Fig. 3. The research plan presented in flowchart.

3. Results

The results from the computational thinking test are shown in Table 2.

The overall research results from the computational thinking test showed that pre-service teachers could develop the score prior to the CSTT (Table 1). According to the overall scores, the pre-test was 9.27 from 12 and the post-test was 10.90. When considering each pillar of computational thinking, the data showed pre-service teachers could develop their thinking skills for all aspects. The algorithm pillar has the highest change in term of development from 1.93 to 2.53 (total score is 3 for each item).

Pattern recognition is the computational aspect that pre-service teachers could get the highest scores in the first place in the comparison of other pillars of computational thinking. However, the data showed the progression in pattern recognition, even if only slight. From Fig. 4, the data from each pre-teacher (n = 30) showed trends of develop-

Table 2

Computational thinking score of pre-service teachers before and after participation in the teacher training program of STEM-based activities in computing science

Computational thinking	Decomposition item 1 (3 point)		Pattern recognition item 2 (3 point)		Abstraction item 3 (3 point)		Algorithm item 4 (3 point)		Total point (12 point)	
	Pre	Post	Pre	Post	Pre	Post	Pre	Pre	Pre	Post
Min	0.00	2.00	0.00	2.00	0.00	2.00	0.00	1.00	0.00	10.00
Max	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	11.00	12.00
Average	2.17	2.53	2.70	2.97	2.47	2.87	1.93	2.53	9.27	10.90
Standard Deviation	0.83	0.51	0.92	0.18	0.90	0.35	1.05	0.63	2.38	0.80

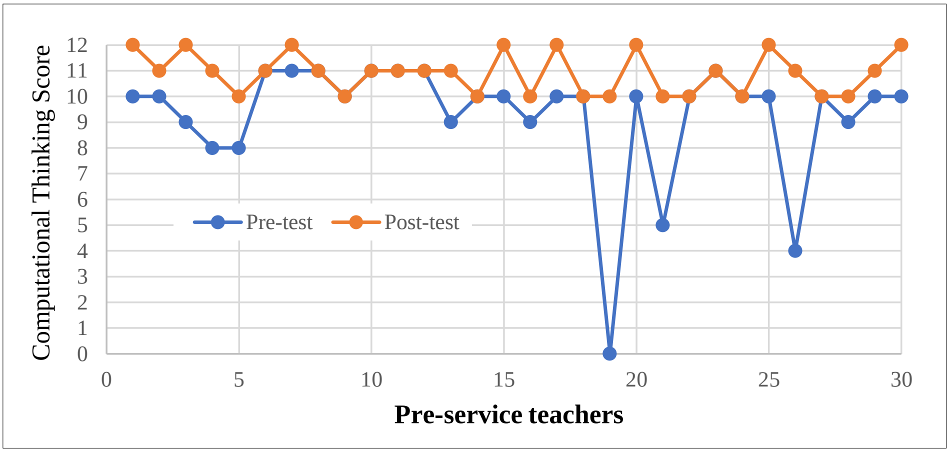


Fig. 4. Comparison of pre-service teachers' computational thinking score before and after participation in the teacher training program of STEM-based activities in computing science.

ment in computational thinking as the post-test was higher that the pre-test scores in a whole picture. Since computational thinking consists of 4 pillars that are needed to solve each item's problem, the different changes in terms of percentage were used to compare the development for all pillars of computational thinking, as the result of pre-service teachers' participation in CSTT program, as shown in Table 3.

The results from Table 3 indicate that the computational thinking pillar for which pre-service teachers scored highest score was pattern recognition, while algorithm scored lowest in the CTT pre-test. After participation in the CSTT program Pre-service teachers improved their scores for all computational thinking skills after taking the CSTT program. Pattern recognition was still the highest score that pre-service teachers could perform the best. The second and third highest scores were abstraction, decomposition, respectively. In terms of the development after participating in CSTT, algorithm was the aspect that had the highest percentage change. It started with the lowest score from pre-test then pre-service teachers could elevate their score after the training program. Algorithm is needed for complex problem solving, especially in higher education (Sondakh, 2018), and was the cognitive skill with the highest percentage change (of 20). The results from the study revealed that before the CSTT program, algorithm was the skill in which pre-service teachers had the lowest score. Considering the measurement of the amount of variation or dispersion of a set of scores by using standard deviation (SD), the CSTT program could narrow the differences of computational thinking level among pre-service teachers.

The CSTT can develop computational thinking based on the assessment of CTT by 1.63 scores of 12 or 13.58 percentage. Since the CTT was developed based on the senior high school level, the pre-test was relatively high. The mean score of the CTT pre-test was 9.27 from 12 teachers, regardless of their background. This research participant mainly from two different programs, elementary program (n = 20) and chemistry program (n = 10). The pre-test score showed slightly different in computational thinking by SD as 2.38, while after the CSTT program, the standard deviation was decreased to 0.8. It indicated that the 12-hrs training program in the format of online training could develop pre-service teachers 'computational thinking and could be used in pre-service teachers from different academic backgrounds (indicated by the high SD

Table 3

Comparison of computational thinking score before and after participating in the teacher training program of STEM-based activities in computing science by the percentage change of mean

Computational thinking	Mean ($\bar{x}$)		Standard Deviation (SD)		Mean Difference	Percentage Change
	Pre	Post	Pre	Post		
Overall	9.27	10.90	2.38	0.80	1.63	13.58
Decomposition	2.17	2.53	0.83	0.51	0.37	12.22
Pattern Recognition	2.7	2.97	0.92	0.18	0.27	8.89
Abstraction	2.47	2.87	0.9	0.35	0.40	13.33
Algorithm	1.93	2.53	1.05	0.63	0.60	20.00

in the pre-test). From the supplementary data, the results grounded into the key characteristics that the two different teacher programs (elementary and chemistry program) could share their understanding and made a learning progression together. The five key characteristics found in the online training program were:

- *Technical support such as online meeting programs, equipment, trainer ICT skills of both trainers and trainees*

  Since 2020, the pre-Service Teacher Training Program has had to be provided online, due to the Covid-19 pandemic. This could not be done without the support of long-duration meeting online programs such as Zoom meeting Pro. The training station used three webcams to incorporate online training activities. In Fig. 5, the trainer flashed codes from pre-service teachers and showed the results to all trainees. The trainer switched between webcam 1 (in the right corner of the picture) and webcam 2 (top-view camera) to show how the .hex files sending from the pre-service teachers worked.

  This activity was found to be very engaging. Pre-service teachers were enthusiastic to see if their codes in .hex file worked properly. A mobile phone could be used as a third camera, to show the broader angle of a moving robot. The online meeting screen in Fig.6 shows both block-based code tray and a moving robot that also involves pre-service teachers during the online program. This kind of activity was called "the activity-on-demands" strategy as the trainer used synchronous online training to benefit from getting pre-service teachers involved in the activity. The trainer used the block-based programming to move according to pre-service teachers' demands, and the camera showed how it worked to complete the mission in real time. This is because the equipment such as robots, microcontrollers, were only available in trainers' studio. Via conference software like Zoom meeting, live coding on demands was the example of the computing science online training during the pandemic that need social distancing approach.

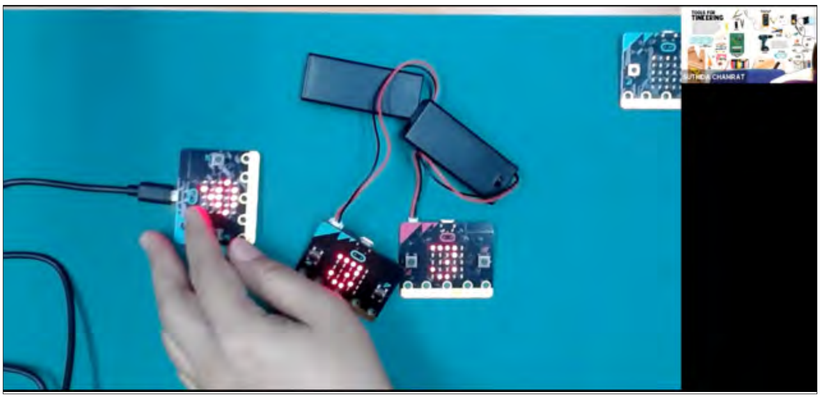


Fig. 5. The use of first and second camera to assist online pre-service teacher training.

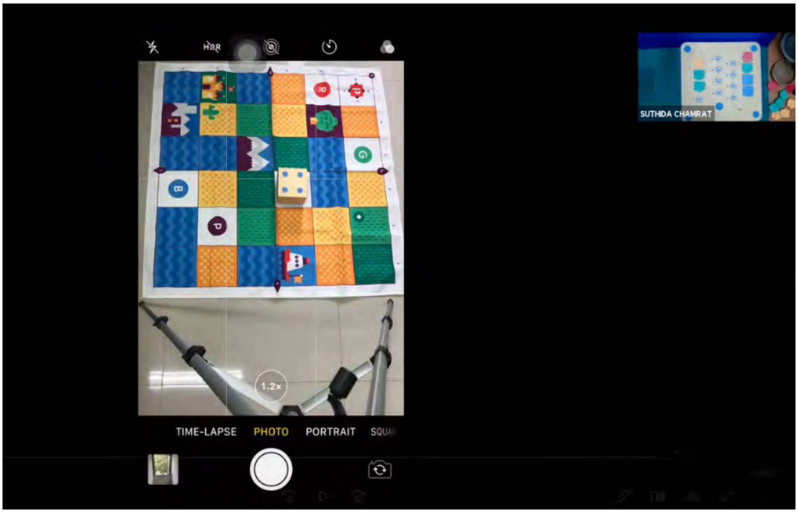


Fig. 6. Live coding of Robot using Activity-on-Demands strategy via Zoom Meeting.

- *Learning management system such as Google Classroom, creating classroom section in* `code.org`

  Learning management systems (LMS) played an important role in online training programs. When they learned to code, preservice teachers could send their files via Google Classroom. The trainer was able to download their files to programming flash to the microcontroller. Furthermore, online platforms like Code.org (Fig. 7) and Codingthailand.org can be useful as they provide primary computing science learning platforms for both students and teachers. Teachers can create

Fig. 7. The trainer monitored pre-service teachers' progress in Code.org classroom section Dance Party 2019.

a classroom section that links to Google Classroom. In classroom section, trainers can monitor and give feedback to the pre-service teacher as they needed. It was useful to the Online CSTT as trainers can assess block-based programming as results of individual pre-service teachers to give feedbacks and point similarities and differences of pre-service teachers' ways of thinking.

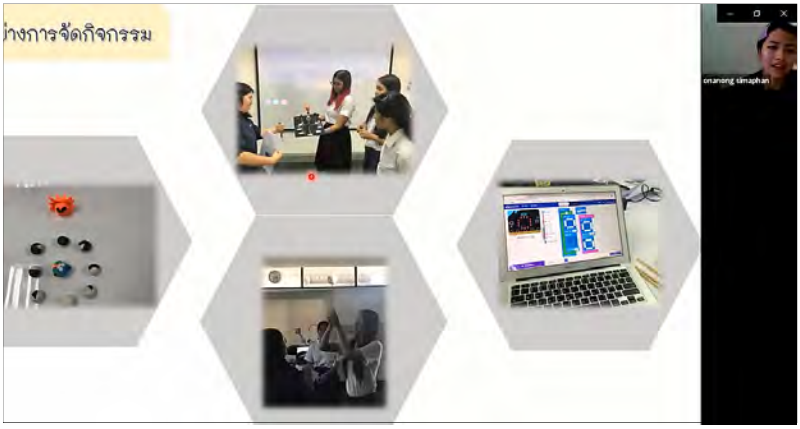
- *The link among policy, curriculum, and implementation*

  Codingthailand.org platform (Fig.8) both meets and aligns with core curriculum and standards. Pre-service teachers saw the opportunity to use this platform in their present and future classroom as teacher interns in the next semester. This activity linked Unit 1 (policy) to unit 2 (Curriculum) with unit 3–4 (implementation of computing science in school).

- *Pre-service teachers' participation*

  The CSTT in this study planned to get pre-service teachers involved in programming activity. Three pre-service teachers (Onanong, Nattarika and Suthikan) who had experience of teaching and learning with the integration of computing science shared and conducted parts of the training program. In Fig. 9, Onanong shared her experience of using Micro:bit in the Moon phase lesson. In method science class, she asked her peer (her students in teaching practice) to simulate moon phase, modeling moon phase cycles and coding Micro:bit to show the moon's eight phases (New moon, Waxing Crescent, First Quarter, Waxing Gibbous, Full moon, Waning Gibbous, Third Quarter and Waning Crescent) using makecode.microbit.org. Onanong also presented her project using Arduino to detect carbon monoxide from car exhaust both inside and outside the car and from motorcycle exhaust comparing between 3-year-old and a new motorcycle (less than a year) in the same brand and specifications.

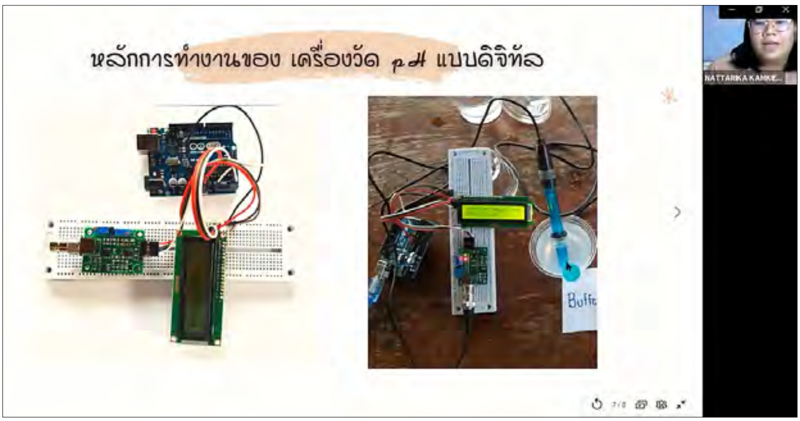
Fig. 8. Codingthailand.org – the online learning platform for computing science that meet and align with Thai core curriculum and standards.



*Translation: Examples of learning activity

Fig. 9. pre-service teacher presented the integration of using Micro:bit in Moon phase class.

In Fig. 10, Nattarika talked about her project that applies the microcontroller and sensor to invent a DIY digital pH meter. She also highlighted the potential of computing science to integrate into a science classroom and STEM project. The pre-service teachers' opportunity to participate in the program to share their experiences and ideas could narrow the gap between trainers and pre-service teachers. Concepts and practices of computing science integrating with STEM based activity could be more accessible via peer instruction. Pre-services teachers were urged to ask and talk more. With their peers, the online meeting became a safe learning environment to communicate effectively. This section helped to build and foster comfortable atmosphere to open up and express their thoughts and ideas where pre-service teachers could thrive, prosper and learn together.



*Translation: The principle of digital pH meter

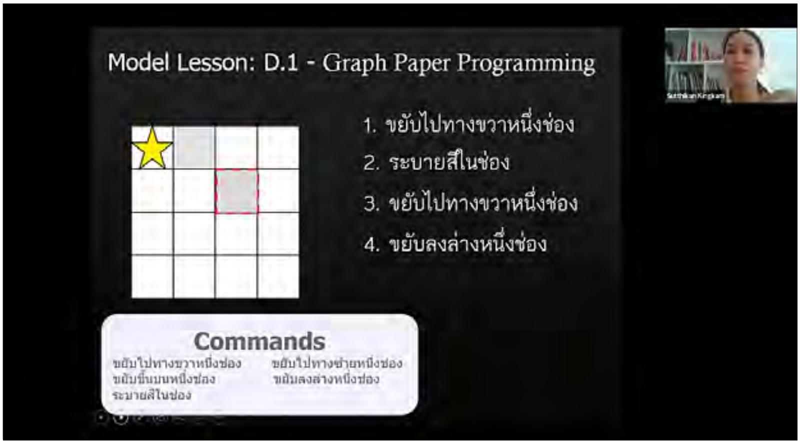
Fig. 10. Pre-service teacher presenting the project "The Applications of Microcontroller for the invention of digital pH meter."

- *Rigor and relevance application of computing science in classroom.*

  The concrete example of computing science teaching and learning in both stand-alone subject and integrative subject into science classroom presented by Suthikan, the pre-service teachers in elementary education major. She had experiences in computing science and used to teach young students in her teaching practicum preparation during her fourth year of becoming student teacher. Teacher education in Thailand before 2019 was a five-year program (Faikamta *et al.*, 2019) so she was in school as an intern teacher. Among three pre-service teachers, Suthikan was the only one who had experiences in school contexts. During the CSTT program, Suthikan was still in elementary school for her teacher internship. Her presentation captured other pre-service teachers' interests because they were going to be teacher intern in the next year. Suthikan pointed that computing science is very important for elementary classroom because it is a basic skill for learning. She mentioned benefits of teaching and integrating computer science into elementary classroom as

> *"Computer science fosters creativity and teaches students critical thinking skills to become proactive learners, so elementary school is the ideal time for students to be introduced to CS"*
> (Code.org, 2020)

Her activity use graph paper programming integrating into art class (Fig. 11). From the elementary pre-service teachers' point of view, transdisciplinary and integration are significantly addressed as the foundation learning philosophy for young learners. The rigor and relevance of how to integrate computing science



*Translation: 1. move right one block
2. paint one block
3. move right one block
4. move down one block

Fig. 11. Pre-service teacher presented her activity – graph paper programing related to art.

into classroom meet pre-service teachers' needs and wants. Some examples of relevant information pre-service teachers could deliver to students through example activity include graph paper programing, unplugged coding via Yoga programming and Move it, Move it – the coding exercises delivered by Code.org (2019). Suthikan also pointed that computing science at the elementary level significantly focuses on computational thinking. Students are not necessary to use a computer to program; activity like unplugged are easily applied to other subjects such as Art, mathematics or language.

Furthermore, the activity during the CSTT allows pre-service teachers to solve problems together by the Breakout Rooms function in Zoom. They learned how to use decomposition, pattern recognition, abstraction, and algorithm to solve problems such as "how can we use computational thinking to find the sum of 1 to 200". This is a typical elementary/middle school level math problem. During this time, the activity sheds light on computational thinking into the way pre-service teachers solve this problem. Working in groups, pre-service teachers were able to provide the solution by using computational thinking, as described in Fig. 12.

1. **Decomposition**
   Decompose the problem to the concrete and understandable representations

   1 + 2 + 3 + 4 + 5 + 6 + 7…………… ….197 + 198 + 199 + 200 = ?

2. **Pattern recognition**
   Try to seek the pattern of the number representations

   1+2+3+4+5+6+7……………………….197+198+199+200

   From the outsides inwards adding:

   Starting with 1 + 200 = 201
   Then adding 2 + 199 = 201
   Next adding 3 + 198 = 201
   Next adding 4 + 197 = 201
   .
   .
   Last adding 100+101 = 201

3. **Abstraction**
   From adding 1 to 200, 2 to 199... 100 to 101 (from pattern recognition), there are always have 201. From 100 pairs, there was only 201 with 100 times.
   The abstraction would be 100 × 201 = 20100

4. **Algorithm**
   Design and draw steps to resolve the problem
   4.1 Represent the numbers from 1 to 200
   4.2 Pattern seeking of paring summation 1+ 200 = 201, 2 + 199 = 201 …. 100 + 101 = 201
   4.3 Draw the representation of all pair to 100 x 201
   4.4 Calculate 100 pairs by multiply with 201
   100 × 201 = 20100

Fig. 12. A rigorous example of computational thinking to solve an upper elementary school/middle school math problem.

4. Discussion

Of the five key characteristics on which this study is based, the following section discusses the connection of CSTT and pre-service teachers' computational thinking. The function of each characteristic and the combination of them that affected on four pillars of computational thinking will be considered.

At first, pre-service teachers' average score in computational thinking was considered as high, at 9.27 out of 12. This suggested they were already good at problem-solving by computing. However, some pre-service teachers hold the difficulties of solving the problems using computational thinking as they had zero points in the pre-test. Then they could develop their scores in the post-test. The overall and separate pillars of computational thinking were increased, as presented in Table 3. The significant discussion towards key findings presented as follows:

- It is possible to provide an intensive program for elementary and science major pre-service teachers to experience the integration of computing science into their classrooms. The concrete examples of using computational thinking started from non-complex math problems then shifted to pH values calculation, including the DIY digital pH meter. STEM projects provided in the final unit of the CSTT preceded the development of pre-service teachers' computational thinking. The research found that the training program in the session of problem solving, cloud started with straightforward math problems that can be solved using basic arithmetic skills, then used computational thinking (as presented in Fig. 12). Later, when the activity progressed, the development of their understanding of computational thinking must be link and relevant the context of the subject matter they teach (Yadav *et al.*, 2014). Since real-life context is also important when teaching computing science, the CSTT module and CTT used real-life situations related to local problems (daily routes in Chiang Mai, a familiar context).
- The positive results of this study are consistent with existing studies. This is because of the proper duration of the training that not too long for the pre-service teachers who have a good basic computational thinking. The training program must meet pre-service teachers' interest and the quality of the course. Even though Veen *et al.* (2012) argued that duration and sustainability are essential features of an effective course, this study suggested that other features such as innovative design, content focus, quality of the content provided, active and inquiry-based learning, collective participation and coherence were equally important in improving the computational skills of the pre-service teachers in this study. It is also important for the content of the program to be situated in real life, as Dolgopolovas and Dagienė (2021) indicated. Contextual environment and related social communication and team-work skills are crucial for computing science, STEM and science education.

- The engagement and participation of pre-service teachers in the CSTT were the essential factors of the effective training program. Three pre-service teachers shared their lead role in CSTT activities instead of the trainer. This exampled the feasibility of implementing computing science in pre-service teacher professional development using peer tutoring that beneficial effect on students at the higher education level as they have chances to construct their own mental model rather than passively listening to class-wide discussion (Nicol & Boyle, 2003). Peer instructions in teaching programming reported by Brown and Wilson (2018) shared similar findings in this study. They suggested ten good practices to lay the foundation of any successful teaching of programming. Those tips also found to be overall effective in CSTT such as peer instruction, live coding, pair programming and use authentic tasks.
- Use of an online platform produced positive results, as shown by multiple studies. The perceptions of lacking teaching resources in coding education hinder teachers' confidence to teach computing science (Kong & Wong, 2017). On the other hand, providing them with multiple sources of learning platforms like Scratch, code.org, makecode.microbit.org, etc., could develop teachers' computational thinking (Adler and Kim, 2018). This benefits pre-service teachers' competency because computational thinking was reported related and supported technological pedagogical content knowledge (TPACK). (Mouza *et al.*, 2017). In Thailand, the organization Digital Economy Promotion Agency (DEPA) had launched the "Coding Thailand" project in collaboration with the private sector to encourage Thai youths to develop computational thinking skills. Codingthailand.org received positive feedback from pre-service teachers because the platform aligned each lesson with the Thai national computing science core curriculum. This paper argues that those online coding platforms are beneficial to both pre-service and in-service teachers, as well as students. The achievements of the core curriculum, standards, and indicators are an important consideration for professional development. The program provider, such as a university, educational institute or teacher education agency needs to carefully align the activity, online platform with the core curriculum.

However, although the results in this study were positive that post-test score was higher than pre-test score and the percentage change rank from 8–20. The discussion of Lee *et al.* (2011) suggests, implementation of computation thinking in the school faces many obstacles. The lack of opportunities for teachers to learn CT is also a significant one. The use of a short but intense CSTT program will be the strategy to promote CT (Bean *et al.*, 2015). Others reported gathering workshops or training for teachers, students and adult learners (Swaid, 2015). The CSTT shows that pre-service teachers can engage in key aspects of STEM activity and computing science within programs focusing on decomposition, pattern recognition, abstraction, and algorithm, as well as recognize computing science and computational thinking with the existing science curriculum. Pre-service teachers from a range of backgrounds were able to develop their computational thinking, regardless of their program.

Acknowledgments

This research was supported by Murata Science Foundation, Japan, in the fiscal year 2019.

The authors are grateful to Mr. John Tucker, MA in Language Testing, University of Lancaster, for kind help in the English correction.

Ethics Statement

All research activity by the author included in the review have been undertaken with approval of Chiang Mai University Ethics Committees, COA No.033/63, CMUREC No. 63/110

All photos of pre-service teachers in this research, used with permission.

References

Adler, R.F., Kim, H. (2018). Enhancing future K-8 teachers' computational thinking skills through modeling and simulations. *Education and Information Technologies*, 23(4), 1501–1514.

Bean, N., Weese, J., Feldhausen, R., Bell, R.S. (2015, October). Starting from scratch: Developing a pre-service teacher training program in computational thinking. In *2015 IEEE Frontiers in Education Conference (FIE)* (pp. 1–8). IEEE.

Brown, N.C.C., Wilson, G. (2018). Ten quick tips for teaching programming. *PLOS Computational Biology*, 14(4). `https://doi.org/10.1371/journal.pcbi.1006023`

Code.org. (2019). *Unplugged: Move It, Move It*. `https://code.org/curriculum/course1/2/Teacher`

Code.org. (2020). *CS Fundamentals for Elementary Schools*. `https://code.org/educate/curriculum/elementary-school`

Daungjun, S. (2018). *Effects of Using Stem Education in Physics on Computational Thinking Ability of Upper Secondary School Students*. Master of Education in Science Education, Department of Curriculum and Instruction, Faculty of Education, Chulalongkorn University.

Digital Economy Promotion Agency. (2019a). *The Program for Promoting Coding Skills for the Future Digital Society: Elementary School Level*. `https://www.depa.or.th/storage/app/media/file/coding1.pdf`

Digital Economy Promotion Agency. (2019b). *The Program for Promoting Coding Skills for the Future Digital Society: Secondary School Level*. `https://www.depa.or.th/storage/app/media/file/coding2.pdf`

Dodero Beardo, J.M., Mota, J.M., Ruiz-Rube, I. (2017). Bringing computational thinking to teachers' training: A workshop review. In J. M. Dodero, M. S. Ibarra Sáiz, I. Ruiz Rube (Eds.), *Fifth International Conference on Technological Ecosystems for Enhancing Multiculturality* (TEEM'17) (Cádiz, Spain, October 18–20, 2017) (pp. Article 4). New York, NY, USA: ACM. doi:10.1145/3144826.3145352

Dolgopolovas, V., Dagienė, V. (2021) Computational thinking: Enhancing STEAM and engineering education, from theory to practice. *Computer Applications in Engineering Education*. 29, 5–11. `https://doi.org/10.1002/cae.22382DOLGOPOLOVAS AND DAGIEN□|11`

Faikhamta, C., Ketsing, J., Tanak, A., Chamrat, S. (2018). Science teacher education in Thailand: a challenging journey. *Asia-Pacific Science Education*, 4(1), 1–18.

Kong, R., Wong, G.K. (2017). Teachers' perception of professional development in coding education. In *2017 IEEE 6th International Conference on Teaching, Assessment, and Learning for Engineering (TALE)* (pp. 377–380). IEEE.

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

Maesincee, S. (2016). *Thailand 4.0 Thriving in the 21st Century through Security, Prosperity, Sustainability*. `http://www.ait.ac.th/news-and-events/2016/news/1thailand-4.0-english-dr.-suvit.pdf`

Ministry of Education. (2017). *Indicators and Concepts in Science Subject (Revision B.E.2560) According to Basic Education Core Curriculum B.E.2551*. Bangkok: Chumnumsahakorn Karnkaset.

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).

National Research Council. (2010). *Report of a Workshop on the Scope and Nature of Computational Thinking*. National Academies Press.

Nicol, D.J., Boyle, J.T. (2003). Peer instruction versus class-wide discussion in large classes: A comparison of two interaction methods in the wired classroom. *Studies in Higher Education*, 28(4), 457–473.

Royal Thai Government Gazette. (2018). *Thailand's 20-Year National Strategy (B.E.2561–2580)*. `http://www.ratchakitcha.soc.go.th/DATA/PDF/2561/A/082/T_0001.PDF`

Sondakh, D.E. (2018). Review of Computational Thinking Assessment in Higher Education. *Retrieved May*, 29, 2019.

Swaid, S I. (2015). Bringing computational thinking to STEM education. *Procedia Manufacturing*, 3, 3657–3662.

Tedre, M., Denning, P.J. (2016). The long quest for computational thinking. In: *Proc. of the 16th Koli Calling International Conference on Computing Education Research*, Koli, Finland, pp. 120–129.

The Institute for the Promotion of Teaching Science and Technology. (2018). *Course Descriptions of Computing Science in k-12 Level*. `http://oho.ipst.ac.th/download/mediaBook/ IPSTprimarysci4year.pdf`

Van Veen, K., Zwart, R., Meirink, J. (2012). What makes teacher professional development effective. *Teacher Learning that Matters: International Perspectives*, 3–21.

Wichaya Pewkam *et al.* (2019). *Teaching Outside the Box: Elementary Outside the Box*. Chiang Mai: Jarusturakijkarnpim.

Wing, J.M. (2006). Computational Thinking. *Communications of the ACM*, 24(3), 33. `https://doi.org/10.1145/1118178.1118215`

Wongsithuwiset, P., Jaroonpipatkul, N. (2017). kap dak sētthakit thī rọ̄kā rọ̄kāo khām [Middle Income Trap: Trap that waits for to be overcome]. *Economy Analysis*. `https://www.bot.or.th/Thai/ResearchAndPublications/DocLib_/Article_7Nov2017.pdf`

Yadav, A., Gretter, S., Good, J., Mclean, T. (2017). Emerging Research, Practice, and Policy on Computational Thinking. Emerging Research, *Practice, and Policy on Computational Thinking*, 205–220. `https://doi.org/10.1007/978-3-319-52691-1`

W. Pewkam was graduated from Chulalongkorn University. Her doctoral degree is curriculum and instruction. Her research interest lies in the creative teaching and learning, area-based teaching and learning as well as computing science for elementary students. She was also the head project of the teachers' preparation for local community which and grant the scholars and incubate future teachers for schools in remote area communities.

S. Chamrat is an instructor at the Faculty of Education, Chiang Mai University, teaching science education at undergraduate and graduate levels. Prior to that, she was with the Ministry of Education of Thailand for 13 years, starting as a chemistry teacher in secondary schools and then a science educator at the Bureau of Academic Affairs and Education Standards. Suthida 's expertise lies in instructional design that reflects the Nature of Science and Technology (NOST). Her recent interest is in using making and tinkering to develop computational thinking skills. Suthida's research focus on transformative learning by innovative pedagogy, such as STEM-based approaches, ICT integrated learning, computational thinking, related to science teacher professional development.