Developing Computational Thinking of Freshmen Using Block-based Programming and Project-based Learning

Virawan Amnouychokanant
King Mongkut's University of Technology Thonburi, Thailand, *virawan.am@mail.kmutt.ac.th*

***Surapon Boonlue**
King Mongkut's University of Technology Thonburi, Thailand, *surapon.boo@kmutt.ac.th*

Saranya Chuathong
King Mongkut's University of Technology Thonburi, Thailand, *saranya.chu@kmutt.ac.th*

Kuntida Thamwipat
King Mongkut's University of Technology Thonburi, Thailand, *kuntida.tha@kmutt.ac.th*

*Corresponding author E-mail: surapon.boo@kmutt.ac.th

Received: 14 November 2021 Accepted: 21 December 2021

ABSTRACT: *In this study, we carried out an experiment with 90 participants (first-year students enrolled in a Bachelor of Education Program in educational technology and communications at King Mongkut's University of Technology Thonburi). At the beginning of the experiment, all participants were asked to take computational thinking (CT) test to measure their CT. During the sessions, all participants were taught by Microbit and needed to make groups to create Microbit artifacts. Although the severity of the COVID-19 pandemic lessened in Thailand, we provided a combination of synchronous learning (live streaming meetings via Zoom) and asynchronous learning (learning management system via Moodle) to prevent the gathering of students in the classroom. However, because project-based learning requires collaboration, members in the group were allowed to meet to complete the project. After the experiment, they took the CT test again. The results show that there was a great improvement in students' post-test results. An assessment based on the investigation of created artifacts demonstrates that the participants acquired CT skills related to programming at a proficient level.*

Keywords: block-based programming, computational thinking, COVID-19, online learning, project-based learning

1. Introduction

The COVID-19 pandemic has affected education at all levels around the world. To prevent the spread of COVID-19, most governments decided to temporarily close educational institutions (UNESCO, 2020). Educational institution closures not only impact students, instructors and families, but also have expansive social effects and economic problems (Bao, Qu, Zhang & Hogan, 2020). The effect was even more severe for underprivileged students and their families, causing disrupted learning, childcare problems, and economic problems for parents who lost income (Mustafa, 2020).

However, COVID-19 crisis has presented an opportunity to facilitate teaching and learning through new instructional methods. In Thailand, students learned from home during the COVID-19 pandemic. When the severity of the COVID-19 pandemic lessened, the students in the same class were divided into two groups: those who study online, and those who study face to face. In addition, it was found that the most appropriate educational innovation during the

COVID-19 epidemic was blended learning. One of the most challenging aspects of online learning for numerous students is the inability to concentrate on a screen for long periods of time. Students are more likely to be easily distracted by social media or other websites while they study online. Therefore, instructors should combine online teaching with different methods, such as the use of web-based technologies, face-to-face learning environments, and learning through practice. The instructors should also keep their online classes engaging and interactive to keep students focused on the lesson (Vanderlinden, (2014).

When considering computational thinking (CT), most people think of the complicated and messy letters of the alphabets on the computer screen. In fact, CT is not only an advanced skill for engineering and programming but also an essential skill in the 21st century (Wing, 2006). Many countries introduce CT courses into curriculums. For instance, the UK has provided a set of CT courses, including information technology, digital literacy, and computer science (Brown et al., 2014). In the same way, Australia has initiated CT courses with primary and secondary students, including the integration of digital technology (DT) courses with CT courses (Vivian, Falkner, & Falkner, 2014). DT courses are the interdisciplinary courses that include English, mathematics, science, and art. As students may already be familiar with the use of technology to solve complex problems, integrating CT courses with their existing strategies may be easier (Armoni, 2012). Furthermore, Poland classifies computer courses into three stages. The first stage of training is for primary school students to gain knowledge of necessary computer skills. The second stage emphasizes training secondary students to enable them to use CT to solve problems. In the third stage, Poland encourages the selection of a computer course as one of the subjects for the final examination at the secondary level. The main goal of the three stages is to assist the students to understand that they can solve problems by incorporating CT and applying CT in their daily lives (Sysło & Kwiatkowska, 2015). Moreover, Thailand is one of the countries in Asia that supports the development of the CT curriculum. Computing Science is a new subject in the Thai education curriculum that has changed from information and communication technology to appropriate with the environment of children. Computing Science consists of three branches of knowledge, including CT, digital technology, and media and information literacy. The main purpose of Computing Science in Thailand is to develop learners to have knowledge and understanding of CT skills, analytical thinking, systematic thinking, and problem-solving. Moreover, students are urged to apply knowledge of computer science and information technology to solve problems in real-life, effectively.

Nowadays, several countries highlight the importance of amalgamating CT courses with several subjects, including promoting it with student training to gain knowledge and understanding of computer and computer program usage. Therefore, when the students grow up, they will have the resources of CT to adapt to solve problems that they will face. Although CT can be integrated into many subjects in several countries, it is unable to imitate the instructional design or the CT development methods due to the differences in the education systems or learning cultures. The same teaching methods or tools may provide different results when used in different environments (Angeli & Giannakos, 2020).

A member of our research team is an instructor who teaches innovation in educational technology and mass communication. He had a desire to encourage their students' CT because he realized that CT is an essential skill in the 21st century and needed in the current and future job markets. From the survey, all students in this class had no prior experience with programming. Block-based programming is considered as an alternative to foster CT because text-based coding is not easy for be-ginners to start coding and the language syntax is a barrier for students to better understand CT concepts (Topalli & Cagiltay, 2018). Furthermore, block-based languages have a pallet of commands, making memorizing commands unneeded; therefore, it is easy for novices (Weintrop & Wilensky, 2019).

Project-based learning (PjBL) is a collaborative teaching and learning approach that is associated with the constructive learning theory (Kokotsaki, Menzies, & Wiggins, 2016). Many research studies show that project-based learning can help students become active learners and build their own experiences by working on their projects (Suryana et al., 2020; Bilgin, Karakuyu, & Ay, 2015; Wu & Wu, 2020). The students who learned with PjBL could perform better on their projects by using higher thinking skills under the facilitation of teachers (Shih & Tsai, 2017).

This research paper asks the following question: Can CT be improved by using block-based programming and PjBL? The purpose is to teach the basic concepts of programming and develop the CT of participants. The paper is organized as follows: Section 2 reviews background literature on CT and PjBL; Section 3 outlines the methodology, including the participants, procedure, and measuring tool; Section 4 presents the results of the experiment; and Section 5 summarizes the discussion and conclusion.

2. Literature Review

2.1 CT

CT is a set of problem-solving processes related to conveying problems and solutions in the same method that a computer operates (Wing, 2011). It involves the conceptual skills and practices to design computations that get computers to work for us and explain the world as a complicated information process (Denning, 2017). Wing (2008) clarified definitions of CT, as shown in Table 1.

Table 1. The definitions of CT

CT is …	CT is not …
thinking at multiple levels of abstraction	only the development process of programming language
a way of human thinking to solve problems	copying the computer's thinking mode
a combination of mathematics and engineering thinking that can apply to various subjects	a skill that is only applied in computer programming
a fundamental skill in daily life that everyone needs	a programming skill used only by programmers or computer scientists

Many scholars attempted to foster CT through different methods. For example, Pérez-Marín et al. (2020) conducted an experiment with 132 primary school students. At the beginning of the experiment, all students took three tests, including a validated test to measure CT, an ad-hoc programming and CT concepts test, and a new test to measure CT created for students aged 9 to 12 years. During the experiment, all students were taught CT and programming through Scratch (MECOPROG) and a drag-and-drop visual interface for Android tablets or smartphones named CompThink App. The app works with CT concepts, including loops, algorithms, patterns, conditionals, steps, instructions, and automats. After six weeks, the students took those same three tests again. The results showed that all levels of students (4th to 6th grades) had improved their knowledge of CT and programming. However, the results revealed that 5th grade students improved their performance in all tests while 4th grade students can understand programming and CT concepts better than 5th and 6th grade students can. Topalli and Cagiltay (2018) proposed that enhancing real-life problem-based game projects with Scratch in classical courses could improve students' CT, programming skills, and motivation. Marcelino et al. (2018) found that most teachers face difficulties when using Scratch in the classroom. For this reason, they developed a distance education course specially designed for primary school teachers to learn both CT concepts and Scratch via an e-learning course on Moodle. Results showed that teachers can im-prove their CT concepts and their use of Scratch after the online course and transfer the knowledge to develop useful teaching tools for their classroom. Curzon et al. (2014) explained that CT is a skill requiring an effective instructional design to achieve the learning goal. To develop CT, they provided the four-step framework: definition, concepts, classroom techniques, and assessment. They also focused on classroom techniques classified by CT components, including algorithm thinking, decomposition, abstraction, and generalization.

According to past studies, CT consists of various components. In this study, we focused on CT components that are often mentioned in previous studies, including decomposition, pattern recognition, abstraction, and algorithm design. In this course, the students need to create innovative artifacts to complete the course. We use Microbit as a teaching tool because it is a tangible device that helps students better understand CT concepts. When the students see the output of their work, it leads to a more concrete understanding of the concepts.

2.2 PjBL

PjBL is a teaching approach in which students learn by engaging in real-world problems and tangible projects. It is closely related to, and often used interchangeably with, problem-based learning, and is also utilized alongside other umbrella concepts such as the inquiry-based approach (Kokotsaki, Menzies, & Wiggins, 2016). However, the key features of PjBL are:

- Learning by doing: PjBL refers to the idea that learning is most powerful when students can apply what they have learned. In PjBL, the student's role shifts from learning by listening to learning by doing (Suryana et al., 2020).
- Real world problems: Across disciplines, the real-world challenge is fundamental to PjBL practice. PjBL establishes a connection between curriculum and external social circumstances through an assigned task

that is a complex and open-ended problem in order to allow for a variety of possible solutions (Kokotsaki, Menzies, & Wiggins, 2016).

- Transformation of the role of instructor: The instructor's role shifts from knowledge provider to facilitator, assisting students in their learning process by facilitating reflection processes and providing substantive support as required. Students are encouraged to take more responsibility for their own learning with less instructor control (Bell, 2010).
- Group work: It is important for students to work together while reflecting on the project because group work provides a great structure for students to receive individual feedback from their peers (Suryana et al., 2020).
- An end product: One of the key features of PjBL is the production of a quality product that drives project planning, production, and evaluation. The final product is a tangible artifact that reflects students' understanding, knowledge, and attitudes toward problems that arise during their answer-finding process (Wu & Wu, 2020).

It is critical to provide a clearly defined and specific goal for the students to achieve. When students were given a particular objective to achieve through PjBL, they produced greater competencies than when they were assigned to finish their work with no clear goal (Ngereja, Hussein, & Andersen, 2020). The students who were given a specific goal were able to point out their own and other students' mistakes, but this was not the case for students who were not given a specific goal (Boyle & Trevitt, 1997). Therefore, setting goals is an essential part of the PjBL process.

The key distinction between universities and other educational settings is universities strive to help students transition from a dependent to an autonomous mode of learning (Bedggood & Donovan, 2012). To fulfill this objective, the instructors should provide learning experiences that promote collaboration with others, effective hands-on experiences, the search for information from a variety of reliable sources, and autonomous working (Boyle & Trevitt, 1997).

3. Methodology

3.1 Participants

A total of 90 freshmen at King Mongkut University of Technology Thonburi in Thailand was a sample group. All of them enrolled in the Innovation in Educational Technology and Mass Communication course and had no prior experience with programming. This quasi-experimental research followed a one-group pre-test–post-test design because the human research ethics committees of KMUTT had concerns about students' equality; all students should receive the same treatments and assessments. Hence, we could not have a control group.

3.2 Procedure

The course for developing CT through block-based programming and project-based learning was conducted from September to October 2020. Although the severity of the COVID-19 pandemic tended to lessen in Thailand at that time, we provided online teaching that combined synchronous learning (video conferencing) and asynchronous learning (e-Learning) to prevent the large gathering of students in the traditional classroom. There were two instructors in this course. In this study, we followed Nizwardi's PjBL model (Jalinus, Nabawi, & Mardin, 2017). The roles of the instructors and the students at each step are shown in Table 2. At the beginning of the study, students were asked to take the pre-test in a Google Form to measure their CT. After taking the pre-test, the students were asked to make groups (5 students per group) for creating the projects. The topic of the project was artifacts related to COVID-19. The students needed to learn in the virtual classroom via Zoom three hours per week. The content of learning activities is presented in Table 3. Students had the opportunity to share their opinions about their projects and consult the instructors during class hours and outside the online classroom. They could also practice coding by watching video clips in Moodle provided by the instructors. The example of blocks in the videos is shown in Figure 1. The creation of the work required the collaboration of the members in the group. To not become a burden on someone in the group, students in each group were allowed to meet to continue the projects. The instructors sent a set of Microbit to representatives of each group. Every week, each group must hand in a report of their progress that identifies the duties of each member, which encourages everyone in the team to participate in the work. Each group was also required to create a video clip demonstrating how to use their invention. Each group created a variety of different Microbit artifacts, such as a counting machine for stores to control the number of people entering and exiting, a rhythm counter and timer for performing CPR when a patient stops breathing, an alarm that sounds when people are less than 2 meters apart, a temperature meter, and a multipurpose sterile storage box. Some examples of the Microbit artifacts are shown in Figure 2. At the end of the course, the students were asked to take the CT test again to determine if there was any difference in developed skills. The instructors then allowed the students to reflect on their learning experience via a post-it note on a wall in Padlet because some students may not be able to fully express their opinions. Writing anonymous comments may give the instructors a fuller picture of past learning activities.

Table 2. Roles of instructors and students

Stages	Steps of PjBL	Instructor's role	Student's role
Skill competency debriefing	Formulation of the expected learning outcome	Inform students of the learning outcomes. At this stage, the instructors must connect the content with real-world situations and encourage them to apply their knowledge in order to analyze and connect their problems to life or society.	Prepare a plan to access information from various sources in order to obtain factual information about the real-life problems that lead to the origin and importance of the project.
	Understanding the concepts of teaching materials	Employ teaching materials and combine them with interesting tasks.	Study the teaching materials provided by the instructors and seek further knowledge from reliable sources.
	Skills training	Train coding and CT skills through step-by-step demonstrations.	Practice coding via Microbit until proficient.
Project work	Designing the project theme	Allow the students to discuss real-world problems, leading to the project topic. Select the theme of the project that students offer. Let the students make the groups for creating artifacts.	Discuss real-world issues and present to the instructors.
	Making the project proposal	Provide suggestions, feedback, and approve projects when possible trends are seen.	Make a project proposal that identifies the problems, solutions, framework, and estimated production (materials, budget, production lead time, and production steps).
	Executing the tasks of the project	Facilitate and advise students when needed.	Create the project and consult with instructors when encountering problems that cannot be solved within the group.
Evaluation	Project presentation	Provide students with comments about their presentations and assess the artifacts according to the rubric score.	Present groups' work processes and working methods of the artifacts.

Table 3. Learning activities in online class

No.	Activities	Blocks
1	• Introduction to Microbit • Coding on LEDs Microbit • Run code on board	• Basic: on start, forever, show string, show leds, show icon, and clear screen • Input: on button A, B, and A+B pressed
2	Measure the brightness of light	Tell the brightness value as a number from 0-255, where 0 is the darkest and 255 is the brightest. • Basic: forever and show string • Advanced: text and join • Input: light level Present the brightness value as a bar graph. • Basic: forever • Led: plot bar graph • Input: light level
3	Measure the temperature	• Basic: forever • Led: plot bar graph • Input: temperature
4	Enjoy with music	• Basic: on start and forever • Music: play tone, ring tone, rest (ms), start melody, set tempo to (bpm), and change tempo by (bpm) • Input: on button pressed, tilt left, and tilt right • Arrays: create array with
5	Create a compass	Measure the value of angle degree as numbers. • Basic: show number, forever, show string, and clear screen • Input: compass heading • Advanced: text and join Tell the direction in letters such as north, represented by the letter N. • Basic: forever and show string • Variables: make a variable, angle, and set item to • Input: compass heading • Logic: if then else
6	Create a dice game	• Input: on shake • Variables: make a variable and set item to • Math: pick random 0 to • Logic: if then else • Basic: show leds and pause (ms) • Input: on button pressed • Function: make a function • Loops: repeat 6 times

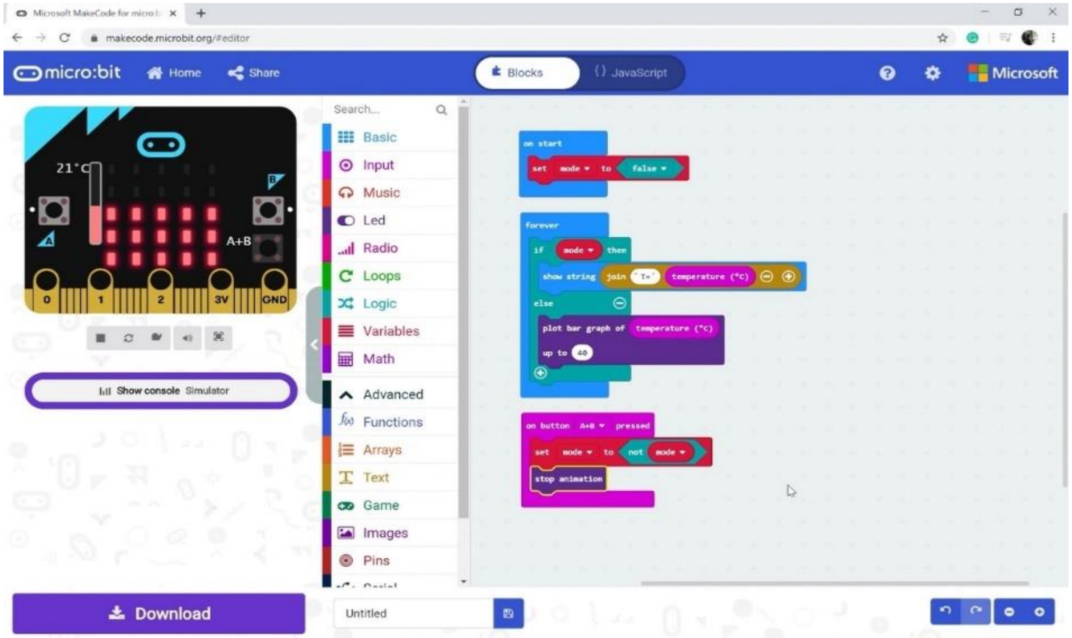

Figure 1. Example of blocks

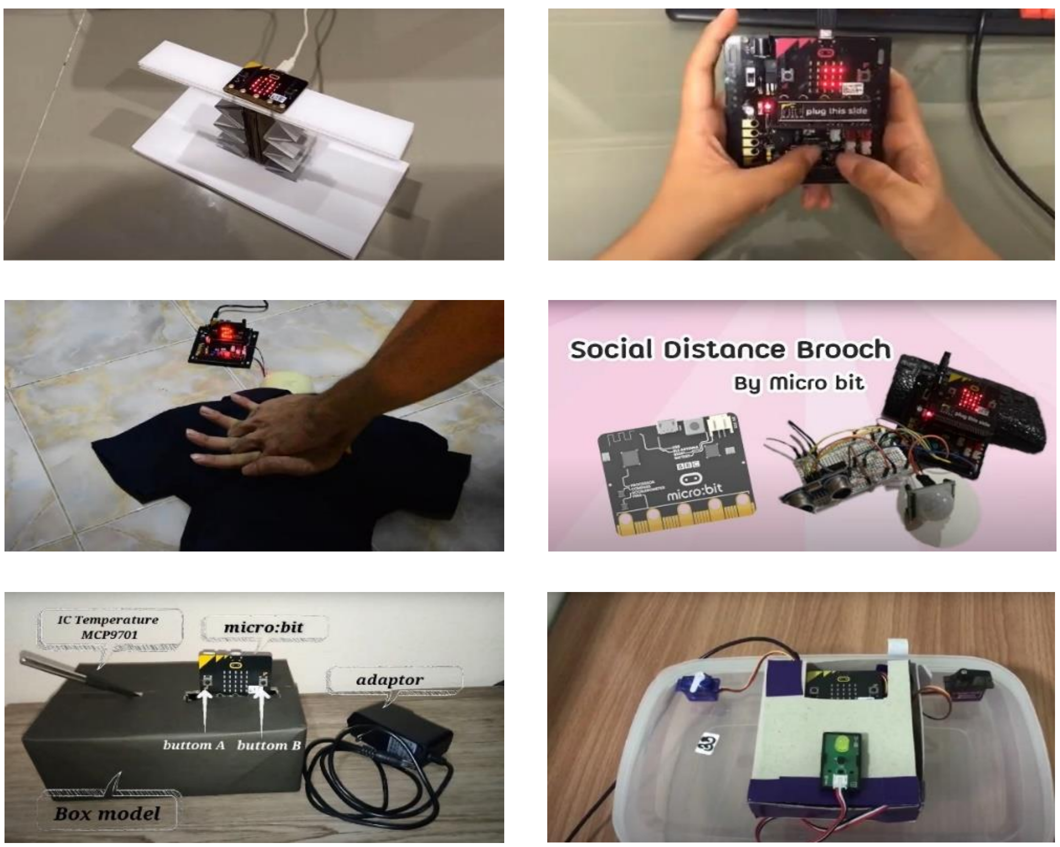

Figure 2. Example of the Microbit artifacts

3.3 Measuring tool

In this study, the CT Test was designed to assess the students' CT abilities. A CT test is specifically designed for students in higher education. The instrument type of the test is a multiple-choice test with 4 answer options. The test is composed of 20 items. The estimated completion time is 30 minutes. Each item addresses one or more of the following four CT components: decomposition, pattern recognition, abstraction, and algorithm design. These concepts are aligned with the Talent Search Computational Challenge of Bebras Organization (Bebras Organization, 2017). The reliability as internal consistency of the CT test measured by Cronbach's Alpha is 0.79, which that can be considered as high reliability. The average of the 20 items is p = 0.59 (medium difficulty), ranging from p = 0.26 (quite difficult) to p = 0.76 (quite easy). Examples of CT test items translated into English are shown in Figure 3 and Figure 4.

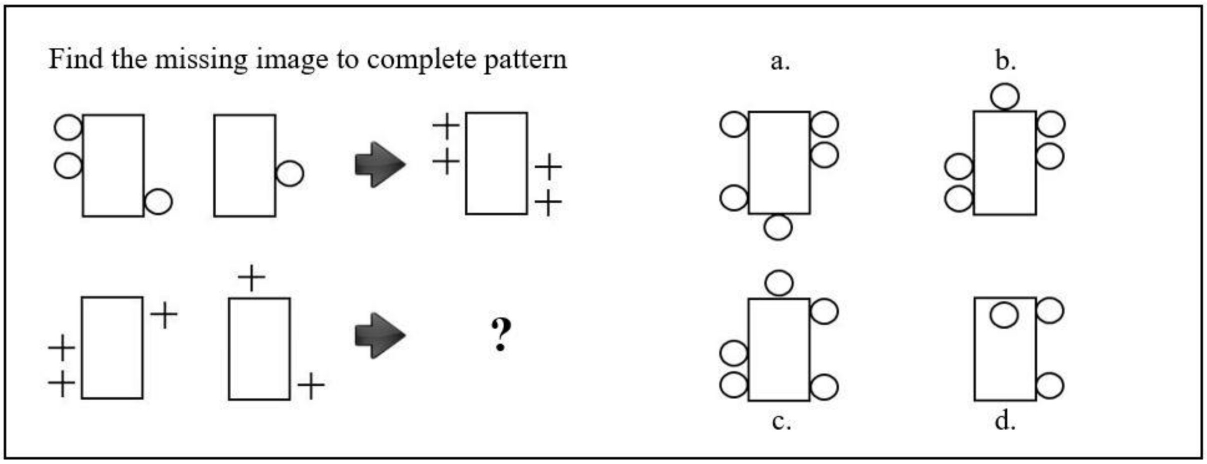

Figure 3. item 2; pattern recognition

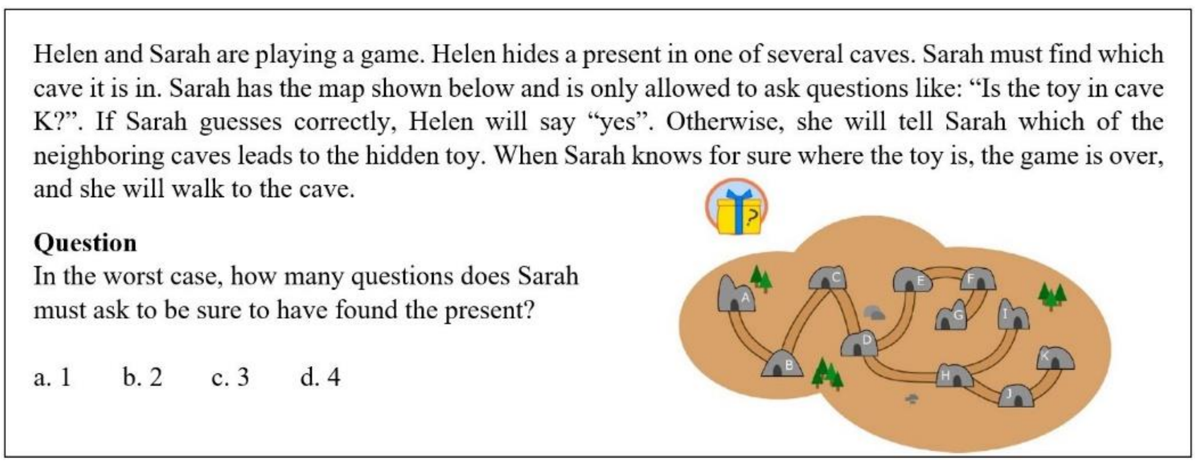

Figure 4. item 8; algorithm design

4. Results

The course for promoting CT through block-based programming and project-based learning was conducted for nine weeks. 90 freshmen attending the Department of Educational Communications and Technology of KMUTT participated in this course. All participants had no prior experience with programming. During the sessions, the instructor taught coding step-by-step via Zoom. Outside of class, the students could practice their coding skills on

Microbit via clip videos in Moodle. They were asked to divide into groups and create Microbit artifacts related to the COVID-19 pandemic to complete the course. The assessment was divided into two parts: group assessment from creating Microbit artifacts and individual assessment from pre- and post-test.

The artifacts of each group were assessed according to the rubric score in Table 4. The criteria for the assessment project are divided into five categories: CT concepts, creativity, usefulness, cost, and possibility. A group that obtains scores in each category of more than 2.5 points is considered at the proficient level, from 2.01 to 2.5 points the developing level, and less than 2 points the basic level. A group in which the obtained total score is lower than 9 points is considered basic level. A group that obtains from 9 to 12 points is at the developing level, and those that obtain more than 12 points are at the proficient level. The analysis of the projects produced by the students reveals that most groups are proficient with CT concepts and creativity but are still at the developing level in terms of other categories. The average of total scores is at the developing level. The mean and the standard deviation of all Microbit projects are presented in Table 5.

The obtained results demonstrate that the students progressively acquired more creativity and CT skills. One possible explanation of this is the nature of the assignment, as the group projects demanded that the students recall the three dimensions of CT (concepts, practices, and perspectives). When students code via Microbit, they develop CT concepts such as sequences, loops, events, parallelism, conditionals, operators, and data. To complete the project, the members in each group must test, debug, and develop iteratively (CT practices). Working in a group can facilitate students' CT perspectives (expressing, connecting, and questioning) because they need to be able to interact with each other.

Table 4. Rubric for Microbit project assessment

	Basic (1 point)	Developing (2 points)	Proficient (3 points)
CT Concept	Consists of one or two of the following: sequences, loops, parallelism, events, conditionals, and operators.	Consists of three or four of the following: sequences, loops, parallelism, events, conditionals, and operators.	Consists of five or six of the following: sequences, loops, parallelism, events, conditionals, and operators.
Creativity	The artifact comes from an existing idea that many others have built.	The artifact is interesting, but it is not new.	The artifact is new and interesting.
Usefulness	The artifact is useless and does not align with the objective.	The artifact is helpful but does not align with the objective.	The artifact is helpful and aligns with the objective.
Cost	The budget is too expensive despite the possibility of using other materials at a lower cost.	The budget is quite expensive, but it is necessary because other materials cannot be substituted.	The used budget is reasonable.
Possibility	The possibility of it being produced is very low because production costs are high and not worth producing.	There is a possibility that it will be produced. However, it may be necessary to modify some functions or codes to fulfill production.	It is highly probable that it will be produced.

Table 5. Mean and standard deviation of all Microbit projects

	M	SD	Interpretation
CT Concept	2.61	0.50	Proficient
Creativity	2.56	0.51	Proficient
Usefulness	2.22	0.65	Developing
Cost	2.33	0.49	Developing
Possibility	2.28	0.89	Developing
Total	12.00	1.94	Developing

To assess students' CT, a pre- and post-test of 20 questions was conducted. A boxplot of the results is presented in Figure 5. Fifty percent of the central data are represented in the box. The interquartile range (Q3–Q1) in the post-test is higher than Q3–Q1 in the pre-test, which means that the CT scores of most students in this course greatly improved. We also found that there is a positive difference in pre- and post-median of 5 points, with 10 points on pre-test and 15 points on post-test. The average in pre-test is 10.4 and in post-test 14.4. However, Q3–Q1 in the post-test is wider than in the pre-test. It can be assumed that the post-test score has more variability than the pre-test score. There are some outliers in the pre-test that show abnormally low and high scores.

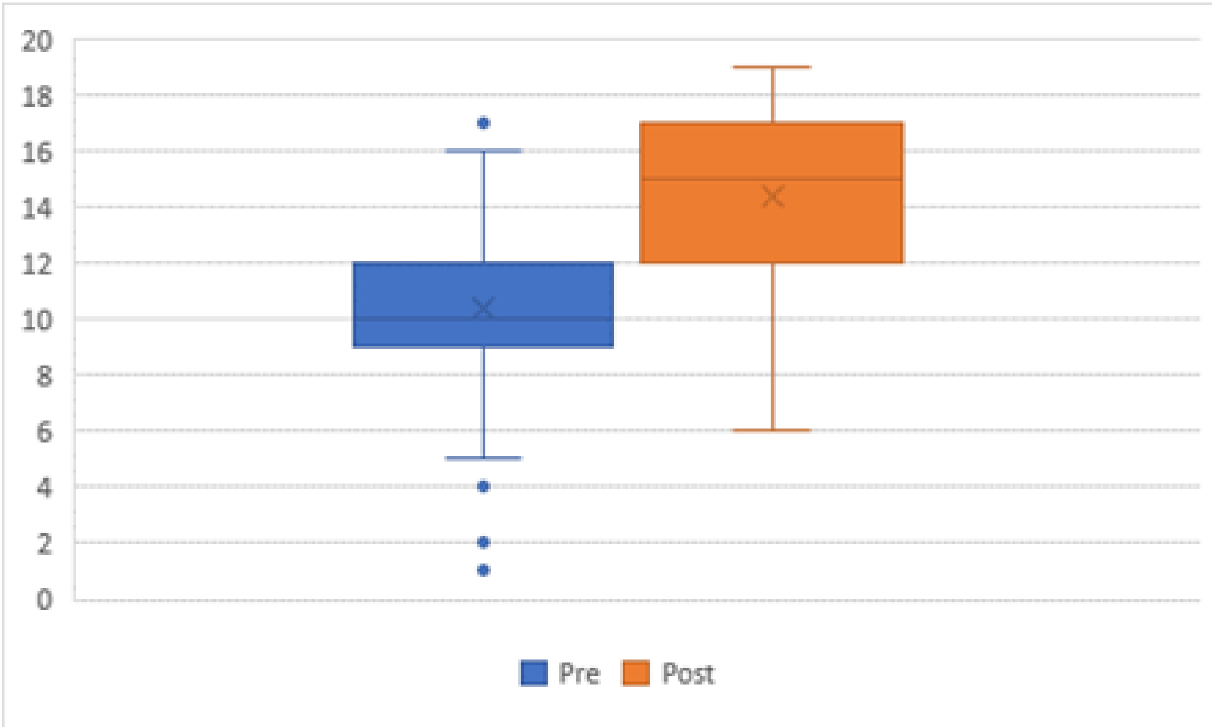

Figure 5. Box plots for CT test score in pre- and post-test

5. Discussion

Online learning was conducted in the Innovation in Educational Technology and Mass Communication course. We taught the students via Zoom and provided the teaching materials for students in Moodle. At that time, the severity of the COVID-19 pandemic lessened. Therefore, we required the students to divide into groups and create Microbit artifacts.

In this article, we presented the learning activities and the assessment of the project involving artifacts creation using Microbit. The purpose of this course is to promote CT skills. We designed the activities through a combination of block-based programming and project-based learning. One of the results shows that there was a great improvement in students' post-test results. This suggests that it is possible to develop the CT of students who have no prior experience with programming even in a short period of time. The selection of Microbit as a teaching tool was beneficial. It corresponds to the research of Voštinár and Knežník (2020) that shows how Microbit is one of the most intriguing low-cost hardware devices whose main advantages include teaching basic programming and algorithmic thinking at all educational levels. This partly supports Weintrop and Wilensky's study on how block-based programming makes programming more accessible and intuitive for beginners. Several features in block-based programming environments are designed to make programming easier. For instance, a block's visual depiction provides cues for where a command should be used. These visual cues prevent incompatible or incorrect statements from combining to form invalid statements (Weintrop & Wilensky, 2019). It can be said that the block-based approach eliminates syntax errors during program development while retaining the tradition of authoring programs instruction by instruction. Block-based programming environments also assist programmers by displaying available commands in logically ordered drawers that can be browsed quickly, a function that learners have found to lower the barriers of programming.

Additionally, one of the results found that most groups have average CT concepts and creativity scores at the proficient level. This is consistent with previous studies (Bell, 2010; Boyle & Trevitt, 1997; Chang, Kuo, & Chang, 2018; Boaler, 1999) that have shown that using PjBL as an instructional approach will raise CT and creative thinking and help students within higher education construct real-life experiences. Because of the rapid digital transformation of today's society, graduates are expected to demonstrate their skills in ways that move far beyond their ability to pass exams with excellent grades (Perrone, 1991). Employers prefer to recruit graduates that have not only hard skills but also soft skills, such as communication, collaboration, teamwork, problem-solving, decision-making, leadership, and emotional intelligence (Ngereja, Hussein, & Andersen, 2020; Touloumakos, 2020). Therefore, the instructors should focus on encouraging both hard and soft skills. In this study, the instructors asked students to create artifacts related to COVID-19, which facilitated learning in order to solve real-world problems. Furthermore, when the students tried to create their artifacts, they discussed and criticized their ideas with one another, which led to improvements in skills and new knowledge. It should be noted, however, that using only one teaching approach in the class is not an efficient way to promote student learning; different methods stimulate various perspectives of learning, so incorporating several methods into a course will help students learn more effectively.

6. Conclusion

This research provided a combination of synchronous learning (live streaming meetings via Zoom) and asynchronous learning (learning management system via Moodle). Block-based programming and PjBL were used to develop the CT abilities of freshmen enrolled in the Innovation in Educational Technology and Mass Communication course. At the beginning of the experiment, all participants were asked to take a CT test in a Google Form to measure their CT. During the sessions, all participants were taught via Microbit. They needed to create artifacts related to COVID-19 to complete the course. The artifacts of each group were assessed according to the rubric. At the end of the course, they took the CT test again. The results show that there was a great improvement in students' post-test results. The students' scores also showed a level of proficiency in CT concepts and creativity. The results of this study may indicate that block-based programming and PjBL can be combined to potentially help students to perform better, in turn affecting their performance in projects.

7. Acknowledgement

The authors would like to express sincere thanks to Petchra Pra Jom Klao Ph.D. Research Scholarship from King Mongkut's University of Technology Thonburi for supporting the expense of research.

References

Angeli, C., & Giannakos, M. 2020. Computational thinking education: Issues and challenges. *Computers in Human Behavior, 105*, 106185.

Armoni, M. (2012), Teaching CS in kindergarten: How early can the pipeline begin? *Acm Inroads, 3*(4), 18–19.

Bao, X., Qu, H., Zhang, R., & Hogan, T. P. (2020). Modeling reading ability gain in kindergarten children during COVID-19 school closures. *International Journal of Environmental Research and Public Health, 17*(17), 6371.

Bebras Organization, 2017, Talent Search 2017 Online Solutions [Online], Available: https://olympiad.org.za/talent-search/wp-content/uploads/sites/2/2017/08/TS-2017-Solutions-Guide.pdf [2019, January 1].

Bedggood, R. E., & Donovan, J. D. (2012). University performance evaluations: what are we really measuring? *Studies in Higher Education, 37*(7), 825–842.

Bell, S. (2010), Project-based learning for the 21st century: Skills for the future. *The clearing house, 83*(2), 39–43.

Bilgin, I., Karakuyu, Y., & Ay, Y. (2015). The effects of project-based learning on undergraduate students' achievement and self-efficacy beliefs towards science teaching. *Eurasia Journal of Mathematics, Science and Technology Education, 11*(3), 469–477.

Boaler, J. (1999). Mathematics for the moment, or the millennium. *Education Week, 17*(29), 30–34.

Boyle, P., & Trevitt, C. (1997). Enhancing the quality of student learning through the use of subject learning plans. *Higher education research & development, 16*(3), 293–308.

Brown, N. C., Sentance, S., Crick, T., & Humphreys, S. (2014). Restart: The resurgence of computer science in UK schools. *ACM Transactions on Computing Education (TOCE), 14*(2), 1–22.

Chang, C. C., Kuo, C. G., & Chang, Y. H. (2018). An assessment tool predicts learning effectiveness for project-based learning in enhancing education of sustainability. *Sustainability, 10*(10), 3595.

Curzon, P., Dorling, M., Ng, T., Selby, C. and Woollard, J. (2014), Developing computational thinking in the classroom: a framework [Online], Available: http://www.chi-med.ac.uk/publicdocs/WP259.pdf [2019, July 16].

Denning, P. J. (2017). Computational Thinking in Science. *American Scientist, 105*(1), 13.

Jalinus, N., Nabawi, R. A., & Mardin, A. (2017, January). The seven steps of project-based learning model to enhance productive competences of vocational students. In *International Conference on Technology and Vocational Teachers (ICTVT 2017). Atlantis Press*. 251–256.

Kokotsaki, D., Menzies, V., & Wiggins, A. (2016). Project-based learning: A review of the literature. *Improving schools, 19*(3), 267–277.

Marcelino, M. J., Pessoa, T., Vieira, C., Salvador, T., & Mendes, A. J. (2018). Learning computational thinking and scratch at distance. *Computers in Human Behavior, 80*, 470–477.

Mustafa, N. (2020). Impact of the 2019–20 coronavirus pandemic on education. *International Journal of Health Preferences Research*, 1–12.

Ngereja, B., Hussein, B., & Andersen, B. (2020). Does Project-Based Learning (PBL) Promote Student Learning? A Performance Evaluation. *Education Sciences, 10*(11), 330.

Pérez-Marín, D., Hijón-Neira, R., Bacelo, A., & Pizarro, C. (2020). Can computational thinking be improved by using a methodology based on metaphors and scratch to teach computer programming to children? *Computers in Human Behavior, 105*, 105849.

Perrone, V. (1991). *Expanding Student Assessment*. Association for Supervision and Curriculum Development, 1250 N. Pitt Street, Alexandria, VA 22314.

Shih, W. L., & Tsai, C. Y. (2017). Students' perception of a flipped classroom approach to facilitating online project-based learning in marketing research courses. *Australasian Journal of Educational Technology, 33*(5), 32–49.

Suryana, F., Jalinus, N., Rahmad, R., & Efendi, R. (2020). Cooperative project-based learning models in programming languages: A proposed. *International Journal of Advanced Science and Technology, 29*(6), 1876–1886.

Sysło, M. M., & Kwiatkowska, A. B. (2015). Introducing a new computer science curriculum for all school levels in Poland. In *International conference on informatics in Schools: Situation, evolution, and perspectives*, 141–154.

Topalli, D., & Cagiltay, N.E. (2018), Improving programming skills in engineering education through problem-based game projects with Scratch. *Computers & Education, 120*, 64–74.

Touloumakos, A. K. (2020). Expanded Yet Restricted: A Mini Review of the Soft Skills Literature. *Frontiers in Psychology, 11*, 2207.

UNESCO, U. (2020). COVID-19 educational disruption and response. *UNESCO*.

VanDerLinden, K. (2014). Blended learning as transformational institutional learning. *New directions for higher education, 2014*(165), 75–85.

Vivian, R., Falkner, K., & Falkner, N. (2014). Addressing the challenges of a new digital technologies curriculum: MOOCs as a scalable solution for teacher professional development, *Research in Learning Technology, 22*, 1 – 19.

Voštinár, P., & Knežník, J. (2020, April). Experience with teaching with BBC micro: bit. In *2020 IEEE Global Engineering Education Conference (EDUCON)*. IEEE. 1306–1310.

Weintrop, D., & Wilensky, U. (2019). Transitioning from introductory block-based and text-based environments to professional programming languages in high school computer science classrooms. *Computers & Education, 142*, 103646.

Wing, J. (2011). Research notebook: Computational thinking—What and why. *The link magazine, 6*, 20–23.

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

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

Wu, T. T., & Wu, Y. T. (2020). Applying project-based learning and SCAMPER teaching strategies in engineering education to explore the influence of creativity on cognition, personal motivation, and personality traits. *Thinking Skills and Creativity, 35*, 100631.