บทความเทคนิค (Technical Article)

Journal of Digital Education and Learning Engineering

ดำเนินการวารสารโดย สมาคมการศึกษาดิจิทัลและวิศวกรรมการเรียนรู้

Effects of Unplugged Coding Game on Promoting Computational Thinking Skills

Maliwan Janta[1], Soraya Thongtamma[2] & Patcharin Panjaburee[1,3*]

Faculty of Education, Khon Kaen University, Thailand[1], Suansanuk Municipal School, Thailand[2], Digital Education and Learning Engineering Association, Thailand[3*]

Received: May 29, 2025 Revised: June 16, 2025 Accepted: June 22, 2025

Abstract

In many primary schools with limited access to computers and stable internet connections, fostering computational thinking (CT) remains a challenge. Since CT is essential for problem-solving in the digital era, alternative approaches, such as unplugged coding games, offer hands-on, screen-free activities that engage students in decomposition, pattern recognition, abstraction, and algorithms without requiring digital devices. Therefore, this study examined the effectiveness of Unplugged Coding game sets compared to traditional lectures in developing CT skills and student motivation among second-grade students. A quasi-experimental research design was employed with 84 students divided into two groups: 41 in the experimental group using Unplugged Coding game sets and 43 in the control group receiving traditional lectures. ANCOVA results indicated that students using Unplugged Coding game sets achieved significantly higher computational thinking comprehension than those in the traditional lecture group. T-test analysis showed significant improvement in CT comprehension within the experimental group, whereas the control group's progress was not statistically significant. Moreover, students in the experimental group exhibited significantly higher achievement motivation than those in the control group. These findings suggest that unplugged coding game sets effectively enhance CT skills and student motivation in primary education settings, making them a viable alternative for schools with limited technological resources.

Keywords: Learning strategy, comprehension, achievement motivation, primary education

■ Introduction

In today's world, technology plays a significant role in various fields, including education, healthcare, and daily conveniences. As a result, education must adapt to the rapid advancements of the

*Corresponding author
Email address: patchapan@kku.ac.th

21st century, leading to curriculum adjustments that integrate Information and Communication Technology (ICT). It has resulted in the introduction of the Computational Science subject, which aims to develop students' computational thinking skills, enabling them to analyze and solve problems systematically and apply their knowledge effectively. However, despite the growing emphasis on computational thinking education, a significant challenge remains as the digital divide. Many schools face limitations in accessing computer devices and stable internet connectivity. These constraints hinder students from engaging with programming environments, coding exercises, and digital tools that are essential for developing computational thinking. The lack of adequate resources creates an inequitable learning experience, leaving students in resource-limited settings at a disadvantage compared to their peers in well-equipped schools.

Unplugged Coding Games provide a way to teach programming concepts without computers, using activities like card-based instructions and board games. According to Jirawarakiat (2021), Unplugged Coding helped students grasp computational thinking fundamentals through engaging activities such as puzzles, board games, and interactive exercises. This approach equips students with problem-solving and analytical skills, preparing them for increasingly complex digital challenges. Similarly, Yaiwong (2022) focused on developing Unplugged Coding games to enhance computational thinking in young students. The study introduced six games, including pattern-matching and adventure-based coding games, which were found to be highly effective learning strategies. The findings indicated that students who engaged with these games showed statistically significant improvements in computational thinking skills compared to their pre-test scores.

Computational thinking is crucial in daily problem-solving, as it involves structured and logical approaches. According to the Institute for the Promotion of Teaching Science and Technology (IPST, 2020), computational thinking consists of four key components, including Decomposition – Breaking down problems into smaller, manageable parts; Pattern Recognition – Identifying trends and commonalities in problems; Abstraction – Simplifying complex concepts into clear, understandable ideas; and Algorithm Design – Creating step-by-step problem-solving procedures. That is to say, challenges in implementing computational science education include insufficient computers in schools and the rotation of classrooms on alternate weeks, which disrupts consistent instruction. Additionally, students often lose interest in traditional lecture-based teaching but show engagement when learning through games and interactive activities.

Therefore, the current study aims to ensure that Unplugged Coding games effectively enhance students' computational thinking, making them particularly suitable for young learners. The following research questions have framed this study:

RQ1: Does learning through Unplugged Coding games and lecture-based instruction impact students' understanding of computational thinking?

RQ2: Does learning through Unplugged Coding games and lecture-based instruction affect students' motivation?

Literature Reviews

Unplugged Coding

Unplugged Coding is a teaching approach that introduces fundamental programming and computational thinking concepts without the use of computers. Instead, it utilizes hands-on activities, card-based exercises, board games, and interactive puzzles to help learners develop logical thinking and problem-solving skills engagingly and tangibly. According to Bell, Witten, and Fellows (2009), Unplugged Coding provided an alternative way to teach computing concepts through physical activities, allowing students to grasp key ideas in computer science without needing digital devices. This method has been particularly effective for young learners, as it enhances their ability to decompose problems, recognize patterns, apply abstraction, and design algorithms—which are core components of computational thinking (Wing, 2006). Moreover, Resnick (2017) emphasized that learning through interactive and playful experiences fosters creativity and deeper engagement, making Unplugged Coding a valuable tool in early education.

The implementation of Unplugged Coding in the learning process followed the structured steps to ensure the effective development of computational thinking skills. Scholars have suggested Unplugged Coding activities in a classroom setting. That is, in the introduction to the computational concept step, the teacher introduces fundamental computational thinking concepts such as problem decomposition, pattern recognition, abstraction, and algorithm design (Wing, 2006). Students were engaged to discuss real-world problems that relate to computational thinking principles. In the engagement through interactive activities, the teacher presents an Unplugged Coding activity using tangible materials such as flashcards, board games, or puzzles (Bell et al., 2009). Thus, students were encouraged to participate in hands-on learning experiences that illustrate the logic behind Coding without using computers (Nokkeaw, 2020). In guided problem-solving and exploration, the teacher facilitates student engagement by encouraging collaborative problem-solving and experimentation with different strategies. Such that, the students could work in groups to design algorithms, recognize patterns, and simplify complex tasks through the given activity (Resnick, 2017). In the discussion and reflection, after completing the activity, students were engaged to discuss their findings and reflect on their problem-solving approaches. The teacher could help them connect their learning to real-world computational applications (Soponpanich, 2019). Furthermore, in the application to real-world scenarios, the students apply the computational thinking skills learned from Unplugged Coding to everyday tasks, such as sequencing steps, breaking down problems, and designing solutions (Wing, 2006). While, the teacher provides follow-up activities to reinforce key concepts and encourage critical thinking (Nokkeaw, 2020).

Computational Thinking

Computational Thinking (CT) is a problem-solving process that involves logical reasoning, pattern recognition, and algorithmic thinking to develop efficient solutions. Wing (2006) defined computational thinking as a fundamental skill for problem-solving, emphasizing the ability to decompose problems, recognize patterns and abstract concepts, and design algorithms to reach solutions effectively. According to The Institute for the Promotion of Teaching Science and Technology (IPST, 2020), computational thinking is not limited to computer science but can be applied across various disciplines to improve analytical and structured thinking. According to IPST (2020) and Wing (2006), computational thinking consists of four essential components: (1) Decomposition – Breaking down complex problems into smaller, more manageable parts. It helps simplify the problem-solving process and allows for a structured approach (Wing, 2006); (2) Pattern Recognition – Identifying similarities, trends, and repeated elements in a problem to find common solutions (Bell, Witten, & Fellows, 2009); (3) Abstraction – Focusing on the essential details of a problem while filtering out unnecessary information, allowing for a clearer understanding and generalization of the solution (Resnick, 2017); and (4) Algorithm Design – Developing a sequence of well-defined steps or rules to solve a problem efficiently (IPST, 2020).

Motivation

The ARCS Model of Motivation, developed by John M. Keller in 1983, is an instructional design framework that aims to enhance student motivation through four key components: Attention, Relevance, Confidence, and Satisfaction. This model emphasizes that motivation is not a fixed trait but a dynamic state that can be fostered through deliberate instructional strategies. The Attention component focuses on capturing and sustaining learners' interest by using techniques such as thought-provoking questions, multimedia, or varied teaching methods. Relevance involves connecting the content to learners' needs, goals, or prior experiences to make the material more meaningful. Confidence addresses learners’ beliefs in their ability to succeed by setting clear expectations, offering support, and promoting incremental success. Finally, Satisfaction ensures learners feel a sense of accomplishment and value in their learning, which can be achieved through recognition, opportunities to apply knowledge, and appropriate reinforcement. Together, these elements create a structured approach for designing motivational learning experiences (Keller, 1983).

■ Unplugged Coding Game Sets

In this study, the Unplugged Coding game set consists of five games designed by the researcher based on the four key components of computational thinking: Decomposition, Pattern Recognition, Abstraction, and Algorithm Design. Four of the games were specifically designed by the researchers to focus on each computational thinking component, while the final game integrates all four components into a single comprehensive activity as follows (Figure 1):

- Game 1: Chick Drawing – The Chick Drawing Game corresponds to decomposition. Students learn to break down the components of a chick image provided by the teacher. They then reconstruct the image by drawing each part separately and assembling them to match the original picture.

- Game 2: Jigsaw Puzzle – Jigsaw Puzzle Game corresponds to Pattern Recognition. Students develop pattern recognition skills by observing the original image and experimenting with trial and error to correctly assemble the jigsaw puzzle pieces to restore the original picture.
- Game 3: The Little Bee Collecting Nectar – The Little Bee Collecting Nectar Game corresponds to abstraction. Students learn to plan the bee's journey to collect as many flowers as possible before returning to the hive. They analyze different paths and eliminate less efficient routes with fewer flowers, demonstrating the process of filtering out unnecessary details.
- Game 4: FIFA World Cup – The FIFA World Cup Game corresponds to Algorithm Design. Students practice choosing the optimal path to guide the ball to the goal, focusing on the sequence of actions rather than the number of command cards used.
- Game 5: Shopping List – Shopping List Game integrates all four Computational Thinking Components. This game combines Decomposition, Pattern Recognition, Abstraction, and Algorithm Design. Students must plan their shopping routes based on a specific list while avoiding obstacles to reach their destination efficiently.

Figure *1.* Unplugged Coding Game.

■ Research Methodology

Participant

The participants of this study consisted of 81 2nd-grade students enrolled in the first semester of the 2024 academic year at a single school in the Northeastern part of Thailand. The study included two classrooms, which were designated as the experimental group and the control group through nonprobability sampling using the purposive sampling method. The classrooms were assigned as follows:

(1) Experimental Group – This group consists of 41 students from a class, who will learn using Unplugged Coding games.

(2) Control Group – This group consists of 43 students from another class who will receive traditional lecture-based instruction.

Measuring Tools

Pre and Post Tests

This test measures students' understanding before and after learning through Unplugged Coding games and traditional lecture-based instruction to assess their computational thinking skills. It consists of 18 multiple-choice questions with four answer choices. The test's reliability was determined using Cronbach's Alpha, yielding a value of 0.737. The discrimination index (r) ranged from 0.11 to 0.49, indicating that the test effectively distinguishes between high-achieving and low-achieving students. Therefore, this test is considered reliable and valid for data collection.

Motivation Questionnaire

This questionnaire was modified from Hao and Lee's (2019) study and used to evaluate students' achievement motivation after learning through Unplugged Coding games and traditional lecture-based instruction. It comprises 18 items using a 5-point rating scale, with responses as 1 = Strongly Disagree, 2 = Disagree, 3 = Neutral, 4 = Agree, and 5 = Strongly Agree. The questionnaire's reliability was assessed using Cronbach's Alpha, which resulted in a value of 0.938, indicating that it is a highly reliable instrument suitable for data collection.

Data Analysis

This study utilized Statistical Package for the Social Sciences (SPSS) version 29 to analyze the data regarding the research questions.

Analysis of Learning Achievement

The pre-test and post-test scores of computational thinking skills between the experimental group and control group were analyzed by conducting an Analysis of Covariance (ANCOVA) to determine the effectiveness of Unplugged Coding games compared to traditional lecture-based instruction.

Analysis of Learning Motivation

The achievement motivation levels between the experimental group and control group were analyzed by performing the Independent Samples T-test to assess differences in motivation between the two instructional methods.

■ Results and Conclusion

Learning Achievement Results

Table 1 shows that students in both groups had statistically significant differences in their understanding of computational thinking prior to participating in the learning activities. It indicates that their pre-existing knowledge differed before the intervention in this study. Therefore, in the post-test analysis, it was necessary to use pre-test scores as a covariate to accurately compare the differences in adjusted post-test mean scores between the two groups.

Table 1

T-test Analysis of Pre-Test Scores between the Experimental and Control Groups

Group	N	Mean Score ± SD	*T*
Experimental	41	8.59 ± 3.755	2.228*
Control	42	10.28 ± 3.202	

* *p < .05*

According to Table 2, in the data analysis, pre-test scores were used as a covariate, while post-test scores were treated as the dependent variable. The adjusted mean and standard error (SE) for the experimental group were 13.687 and 0.420, respectively, whereas for the control group, they were 10.135 and 0.409, respectively.

The results clearly showed that the experimental group achieved significantly higher post-test scores than the control group, with $F_{(1, 84)} = 35.656$ ($p < .05$). Additionally, the effect size ($\eta^2 = 0.306$), based on Cohen (1988), indicates a moderate-to-large effect. Thus, it can be concluded with 95% confidence concluded that students learning through Unplugged Coding games had a better understanding of problem-solving and programming than those who received traditional instruction. The effectiveness of this game-based learning approach is attributed to its design based on computational thinking principles and the collaborative nature of the activities, which promote deeper learning. These findings support previous research by Grover and Pea (2013), highlighting the importance of non-digital activities in developing computational thinking skills typically absent in traditional lectures.

Table 2

ANCOVA Analysis of Post-Test Scores between the Experimental and Control Groups

Group	N	Mean	SD	Adjusted mean	SE	F	η^2
Experimental	41	13.37	1.854	13.687	.420	35.656*	.306
Control	42	10.44	3.673	10.135	.409		

** p < .05*

Learning Motivation Results

Table 3 presents the results of the Independent Samples *T*-test comparing the achievement motivation levels between the experimental and control groups. It was found that the experimental group had a significantly higher mean score of attention, relevance, confidence, and satisfaction than the control group.

Table 3

T-test Analysis of Students' Achievement Motivation Questionnaire between the Experimental and Control Groups

Category	Student Groups	Number of Students (persons)	Mean Score ± SD	t-value
Attention	Experimental group	41	4.34 ± 0.27	11.926*
	Control group	43	3.13 ± 0.45	
Relevance	Experimental group	41	4.43 ± 0.35	10.924*
	Control group	43	3.08 ± 0.45	
Confidence	Experimental group	41	4.44 ± 0.33	11.343*
	Control group	43	3.09 ± 0.46	
Satisfaction	Experimental group	41	4.58 ± 0.28	12.970*
	Control group	43	3.22 ± 0.34	

** p < .05*

The *T*-test analysis confirms that there is a statistically significant difference in achievement motivation between the experimental and control groups across all dimensions. These results indicate that students who learned through Unplugged Coding games demonstrated significantly higher achievement motivation compared to those who received traditional lecture-based instruction, with a 95% confidence level. The findings of this study are consistent with the research of Dickey (2005), which reported that learning activities emphasizing participation and hands-on engagement enhance students' motivation and satisfaction in learning.

■ Conclusion

The comparison of computational thinking understanding between students who learned through Unplugged Coding games and those who received lecture-based instruction revealed that students in the Unplugged Coding group demonstrated significantly higher computational thinking understanding than those in the traditional lecture group. Similarly, the comparison of achievement motivation between the two groups showed that students who learned through Unplugged Coding games had higher mean scores across all dimensions of achievement motivation, particularly in the satisfaction dimension, which was significantly higher than the control group. These findings suggest that Unplugged Coding games effectively enhance students' achievement motivation. Additionally, group collaboration and social interaction during gameplay may be contributing factors in further increasing students' engagement and motivation.

■ Acknowledgement

The researchers thank the staff, students in the school and parents of students for their consent to participate in the study and cooperation during the data collection process. Their assistance has been instrumental in making this research possible.

■ References

Bell, T., Witten, I. H., & Fellows, M. (2009). Computer science unplugged: An enrichment and extension programme for primary-aged children. University of Canterbury.

Cohen, J. (1988). Statistical Power Analysis for the Behavioral Sciences (2nd ed.). Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers. https://doi.org/10.4324/9780203771587

Dickey, M. D. (2005). Engaging by design: How engagement strategies in popular computer and video games can inform instructional design. Educational Technology Research and Development, 53(2), 67-83. https://doi.org/10.1007/BF02504866

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. Educational Researcher, 42(1), 38–43. https://doi.org/10.3102/0013189X12463051

Institute for the Promotion of Teaching Science and Technology (IPST). (2020). Guidelines for developing computational thinking skills in the computational science curriculum for elementary education. Bangkok: IPST.

Hao, K. C., & Lee, L. C. (2019). The development and evaluation of an educational game integrating augmented reality, ARCS model, and types of games for English experiment learning: an analysis of learning. Interactive Learning Environments, 29(7), 1101–1114.

Jirawarakiat, S. (2021). Unplugged Coding: Learning coding without computers. Bangkok: Thai Education Publishing.

Keller, J. M. (1983). Motivational design of instruction. In C. M. Reigeluth (Ed.), Instructional design theories and models: An overview of their current status (pp. 386-433). Lawrence Erlbaum Associates.

McClelland, D. C. (1961). The achieving society. Princeton, NJ: Van Nostrand.

Nokkeaw, A. (2020). Unplugged coding activities: A strategy for developing computational thinking without computers. Journal of Educational Technology, 12(1), 45-58.

Resnick, M. (2017). Lifelong kindergarten: Cultivating creativity through projects, passion, peers, and play. MIT Press.

Soponpanich, K. (2019). The role of unplugged Coding in early childhood and primary education. Bangkok: Institute of Learning Development.

Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33-35. https://doi.org/10.1145/1118178.1118215

Yaiwong, P. (2022). Development of Unplugged Coding games to enhance computational thinking skills for early elementary students. Journal of Computer Education, 10(2), 45-60.