Integrating Thinking Process Model Technique with Online Collaborative Learning to Promote Programming Logical Thinking

Kannika Daungcharone[1,*], Krittawaya Thongkoo[1,*], and Patcharin Panjaburee[2]

[1]College of Arts, Media and Technology, Chiang Mai University, Chiang Mai, Thailand
[2]Faculty of Education, Khon Kaen University, Khon Kaen, Thailand
Email: kannika.d@cmu.ac.th (K.D.); krittawaya.t@cmu.ac.th (K.T.); patchapan@kku.ac.th (P.P.)
*Corresponding author

Manuscript received September 19, 2023; revised November 14, 2023; accepted December 14, 2023; published March 22, 2024

***Abstract*—Logical thinking is essential for students learning computer programming, enabling them to analyze problems and design solutions effectively. Flowcharts, a thinking process model technique, are particularly useful for illustrating program processes and fostering logical reasoning. Additionally, transitioning to a collaborative learning environment enhances students' social skills and motivation through group interactions. This study explored the impact of a novel learning approach that merges the thinking process model with online collaborative learning among 410 university students. It examined how different learning approaches and individual preferences affected their learning achievement and motivation. The findings indicate that learning achievement varied significantly with different learning approaches and individual preferences but showed no significant variance when combining both factors. For learning motivation, significant differences were noted across all dimensions with varying learning approaches, and some dimensions were affected by individual preferences, with no significant change observed across all dimensions when combining learning approaches and individual preferences. The research concludes that incorporating thinking process models and collaborative learning can substantially improve students' learning outcomes and motivation.**

***Keywords*—online pedagogy, computer science education, higher education, individual differences**

I. Introduction

Recently, numerous professional fields require programming expertise. Computer programming is a skill required by various professionals, including programmers, engineers, data scientists, mathematicians, and education analysts [1]. Over the past decade, computer programming has gained popularity because of rapid technological advancements, and it is believed to foster other skills such as problem-solving, logical thinking, and creativity [2]. Computer programming is a course where students must practice extensive exercises to acquire good programming skills and debugging experience. However, today, students lose enthusiasm and interest in learning computer programming, particularly if they experience repeated failure while practicing independently. The need to enhance the teaching and learning of computer programming necessitates focusing on multiple factors influencing students' learning motivation. In addition to individual differences, learning approach, infrastructure, and classmate pressure can affect students' learning motivation and efficacy. Most students think learning computer programming is difficult because they need to challenge themselves with the huge number of programming language syntaxes, problem analysis, problem-solving design, and debugging errors between developing programs, which are the causes of students' high failure and dropout rates [3, 4]. Therefore, the teacher must discover creative approaches to motivating students to learn computer programming and encouraging them to achieve high levels of success. The thinking process model technique as a flowchart is widely used in computer programming courses because it facilitates programming logic comprehension, encourages algorithm development, and promotes logical thinking. Therefore, incorporating flowcharts into various teaching methodologies underscores their significance in equipping students for successful programming careers [5, 6]. In addition, online collaborative learning fosters effective collaboration skills among students by allowing them to participate actively in their learning process without limiting learning location, resulting in high student performance as they share their knowledge and learn from one another. This approach emphasizes group work, listening to peer feedback, and fostering student engagement, which promotes students' learning motivation and achievement [7–10].

This study aims to develop the Logical-Thinking Diagnosis System (LTDS) to transform the traditional programming classroom into a novel learning approach. The main feature of LTDS is the integration of the beneficial attributes of the thinking process model technique with online collaborative learning. It promotes the cultivation of logical thinking by assisting students in analyzing and designing problems into programming processes through flowchart diagrams. Additionally, this system enhances learning motivation and achievement by allowing students to engage in interactive knowledge sharing, knowledge construction, and developing higher-order thinking skills without being constrained by location or time limitations. In addition to this system being beneficial to students, it is also helpful to teachers by changing the role of the teacher to that of a mentor, who can observe students' learning behavior in the classroom and provide support by offering advice. Therefore, to ensure this approach can promote students' learning achievement and motivation, students who prefer to learn independently and in groups achieve achievement and motivation. Two questions have been formulated as follows:

RQ1: Are there any differences in learning achievement and motivation between students who learn in the

doi: 10.18178/ijiet.2024.14.3.2071

collaborative learning classroom and the traditional programming classroom?

RQ2: How do individual preferences affect students' learning achievement and motivation in the different learning approaches?

II. Literature Review

This study compiled data from various studies that support active learning, collaborative learning, and technology to enhance student's learning motivation and achievement in computer programming.

A. *Computer Programming and Logical Thinking*

To acquire computer programming, students must practice beyond theory and understand input, process, output, and process structures (sequence, selection, and repetition). Students must read and understand programming syntax and write programs to solve problems. This procedure presents challenges and frustrations that students must overcome with perseverance and commitment [11, 12]. In addition to the programming language syntax, computer programming requires computational and logical thinking to analyze, solve, and design program processes [13]. Computational thinking involves problem decomposition, pattern recognition, and algorithm development, while logical thinking encompasses systematic reasoning, deduction, and problem-solving through logical steps [14]. Computer programming courses are typically challenging and have a high dropout rate because most students lack a strategy for translating programming problems into practical plans and algorithms [4]. Both instructors and students in higher education consider mastering computer programming difficult. Traditional classroom learning through face-to-face interaction cannot support effective teaching or student achievement.

Information technologies have been mandated to increase students' comprehension and motivation and overcome course learning difficulties. Gamification, online tools, and computer simulations help students learn and implement computer programming [15]. Several researchers investigated how to motivate students to improve their programming education. Most students could benefit from learning programming through a visual environment such as diagrams, animation, and drag-and-drop applications because they are joyful and at ease while learning, allowing them to develop higher-order reasoning skills instead of passively receiving ideas [3, 11, 16, 17]. In recent years, block-based programming languages like Scratch [18] and mBlock [19] have emerged to help students and beginners focus on programming logic and structure without syntax. This approach has been used to introduce students to programming due to its capacity to graphically depict complex concepts, its ease of use with drag-and-drop, and its descriptions of blocks in natural language.

B. *Thinking Process Model Technique in Computer Programming*

The flowchart is one technique of the thinking process model widely adopted in various fields such as business, engineering, science, and computer science. It serves as visual representations of algorithms, processes, and workflows, enabling students to grasp complex programming concepts through structured and organized visuals [19, 20]. Flowcharts offer numerous advantages. They foster and enhance understanding of programming logic by visualizing operations, sequences, selection, and repetition structures [21]. To create a program flowchart, students must comprehend the categories and meanings of various flowchart symbols. Then, they must design the program processes, arrange the symbols according to the process design, and define navigation by connecting them with arrows. It supports students in decomposing problems into manageable steps as a crucial pedagogical tool facilitating problem-solving and algorithmic thinking [5].

Flowcharts also contribute to developing logical thinking by illustrating the flow of processes and decision-making. It enables students to practice systematic reasoning, deduction, and problem-solving through logical steps. Consequently, flowcharts lead to a deeper understanding of the relationship between programming elements, resulting in improved code design and implementation [6, 20]. Educators have recognized the value of flowcharts in computer programming education and integrated this technique into various teaching methodologies, such as problem-based learning and collaborative learning. These approaches promote student collaboration in creating flowcharts and reinforcing programming concepts while fostering communication and teamwork skills [21].

C. *Online Collaborative Learning Promotes Learning Motivation and Achievement*

The European Higher Education Area proposed collaborative learning as a valuable instrument for enhancing university students' skills and abilities [22]. Due to the proliferation of new information technology, the use of this methodology within the framework of educational innovation has become pervasive in compliance with policy focusing on the education management of the Ministry of Education in Thailand since 2019 [22, 23]. Collaborative learning is an educational strategy that enables students to engage in learning activities through their collaborative efforts. It encourages students to reflect on their thoughts and develop social skills through group interaction. This learning activity impacts students' active engagement, leading to an enhanced and more profound learning experience [9, 10, 24]. In recent years, online collaborative learning has developed. It has become a popular method for motivating and achieving learning because students can interact, share, construct, and develop higher-order thinking skills without time or place constraints [10, 22]. These elements foster a sense of belonging and student engagement, which can be applied to onsite and online learning environments [8, 25]. Several studies have shown that behavioral engagement in online collaborative learning positively affects learning performance. Students with a high level of behavioral engagement in online collaborative learning tend to achieve superior academic results [9]. Besides, its characteristics, components, and structure contribute to its key advantages, resulting in effective outcomes through active, motivating, and collaborative learning experiences. In summary, online collaborative learning is a powerful approach that enhances learning motivation and achievement, which leads to effective learning outcomes such as improved academic performance and higher-order thinking skills [7].

III. Research Methodology

This study examines the effect of a novel learning environment incorporating the thinking process model technique with online collaborative learning on students' motivation and achievement. This study employed a quasi-experimental research design to collect data from two groups of students by considering the difference between students who learn in traditional classrooms and those who learn in collaborative learning classrooms with the LTDS. The description of the experiment is explained in the following section.

A. Participants

The participants in this study were 410 freshmen from five regions of Thailand who study information technology, computer science, or related fields, which have the general information list in Table 1. In terms of learning environment, according to the study's objective, 203 students learn in the Traditional Classroom (TDC), and 207 students learn in the Collaborative Learning Classroom (CBC). Regarding individual preferences, 175 students prefer to learn independently (SDL group), while 235 prefer to learn in a group (CBL group), as determined by a questionnaire about their favored learning behavior.

Table 1. Participants' sociodemographic characteristics

Name	Option	Frequency	Percentage (%)
Gender	Male	247	60.24
	Female	163	39.76
Age	18 years	112	27.32
	19 years	156	38.05
	20 years	72	17.56
	21 years	47	11.46
	22 years	23	5.61
Region	Northern	156	38.05
	Northeastern	60	14.63
	Central	61	14.88
	Eastern	78	19.02
	Southern	55	13.42

B. Development of a Logical Thinking Diagnosis System

Fig. 1 presents the system architecture of the Logical-Thinking Diagnosis System (LTDS), which includes front-end and back-end parts. The users are teachers and students, who can interact with the system via the front-end website to input data into the system and get information. Besides, the back end supports the system's operations by containing the file system for processing, connecting to the draw.io website to draw the flowchart, and managing the database. Teachers can manage the classroom by creating classrooms and defining the number of students and number of members per group in the classroom. Moreover, they can manage the classroom's courses by creating the courses, adding the course's contents and questions, checking the answers, chatting with students via chat box, and giving them feedback for improving their answers. On the other hand, students can learn via the LTDS by logging in to the classroom and choosing the course to learn. They can read and review the course's contents and questions, answer the questions by designing the flowchart via the draw.io website with their group members, and get feedback on their answers from the teacher.

The LTDS includes features for assessing and giving feedback on students' logical thinking as it applies to programming, identifying areas where students may struggle or excel. The system uses flowcharts as a thinking process model technique. Flowcharts are a universally recognized method for diagramming the sequence of operations in a computer program or any process. Technically, the system has a user interface that allows for creating and manipulating flowchart elements like shapes and connectors. Flowcharts translate complex programming algorithms into visual diagrams, simplifying understanding of program structure and logic. This feature caters to visual learners and can help debug and plan before actual coding. With flowcharts, the system encourages algorithmic thinking, a step-by-step approach to solving problems, an essential skill in programming. The LTDS supports online collaborative learning, which technically would involve a networked environment where multiple users can interact with the same flowchart in real time. A chat box indicates that the LTDS has communication tools integrated into its interface, allowing for synchronous discussion and brainstorming, which is a part of collaborative learning. Technically, the system includes feedback mechanisms based on students' flowcharts, helping them iteratively improve their logical thinking skills.

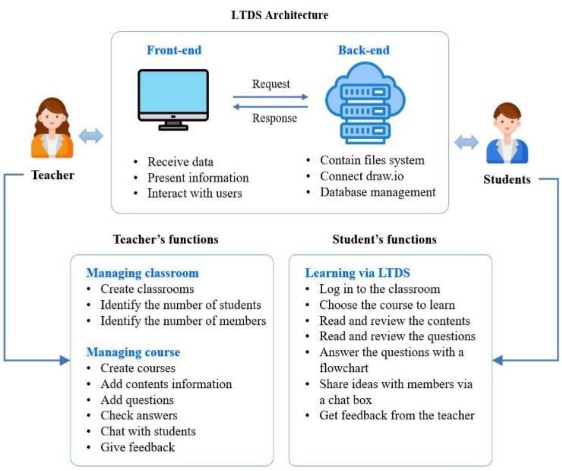

Fig. 1. The LTDS system architecture.

Fig. 2 represents the layout of LTDS, which allows students to design flowcharts by dragging and dropping symbols from symbol categories onto the collaborative area. Besides, while doing the activity, students can review the details of the content and its questions by clicking the menu on the right-hand side. All group members can design flowcharts and communicate with others via the chat box window, as shown in Fig. 3. After completing the group assignment or time-out, the teacher can evaluate their answers and provide feedback to help them comprehend their mistakes and how to rectify them.

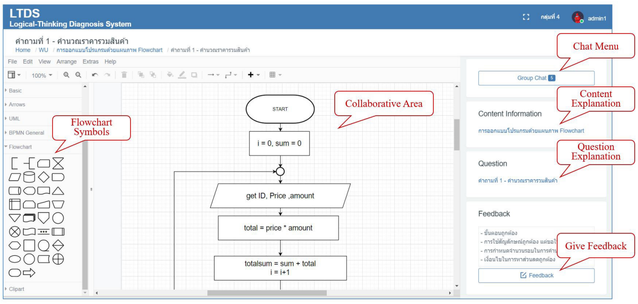

Fig. 2. The interface of LTDS system.

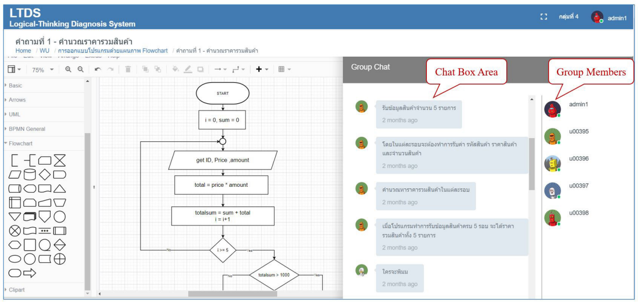

Fig. 3. The interface of collaborative learning via the designing area and chat box window.

C. Measurement Tools

In this study, three measurement instruments are devised, developed, and utilized for evaluating students' learning motivation and achievement. The pre-test is used to assess the student's performance before learning, while the post-test is used to evaluate performance after learning. They are created by lecturers who taught the Fundamentals of Computer Programming Course. The tests consist of twenty multiple-choice questions to assess the capability to analyze and design programming. The tests encompass using symbols, defining input and output values, and designing the program's processes, sequence, selection, or repetition structures. The KR-20 coefficient is used to assess the reliability of both the pre-test and post-test, with reliability values of 0.73 and 0.75, respectively, indicating reliable questions.

The motivation questionnaire is applied by Glynn *et al.* [26] and translated into Thai to evaluate students' intrinsic and extrinsic motivation when learning to analyze and design programming. The questionnaire consists of a five-point Linkert rating scale from 1 to 5, with responses including 1 meaning strongly disagree, 2 meaning disagree, 3 meaning neutral, 4 meaning agree, and 5 meaning strongly agree. It consists of 20 items distributed across four dimensions (five questions per dimension). Intrinsic Motivation (IM) refers to inherent enjoyment in learning content. Self-Determination Motivation (SDM) pertains to a student's confidence in their ability to study. Self-Efficacy Motivation (SEM) refers to a student's belief that they can learn effectively. Career Motivation (CM) is anticipating acquired knowledge in future careers. The questionnaire has Cronbach's alpha (α) value of 0.94, indicating its reliability.

D. Experimental Procedures

This study's experimental procedures are intended to evaluate the LTDS's capacity to promote student learning motivation and achievement. Fig. 4 shows the experimental procedures for which participants were required to complete a pre-test for 20 minutes. Then, the participants learned to analyze and design programming processes and participated in classroom activities for 120 minutes. The participants were divided into two groups by the systematic sampling technique: 203 students were control group participants learning in the Traditional Classroom (TDC) environment, and 207 students were experimental group participants learning in the Collaborative Learning Classroom (CBC) environment. For the TDC activities based on synchronous learning, a teacher acts like a knowledge conveyer to teach the topic contents, explain the examples, give the assignments, discuss the results, and summarize the lesson. Meanwhile, the students were required to listen, take notes, ask questions, do assignments, and discuss the results. On the other hand, in the CBC activities, a teacher acts like a mentor to brief the topic contents, give assignments, observe students' behavior, give suggestions, discuss the results, and summarize the lesson. The CBC students were required to listen to the overview of topic contents and be allowed to learn and practice in groups via the LTDS to collaborate for brainstorming, sharing ideas, summarizing answers, helping each other, doing assignments, and discussing the results. After finishing the class activities, the participants were required to complete a post-test for 20 minutes and a learning motivation questionnaire for 20 minutes.

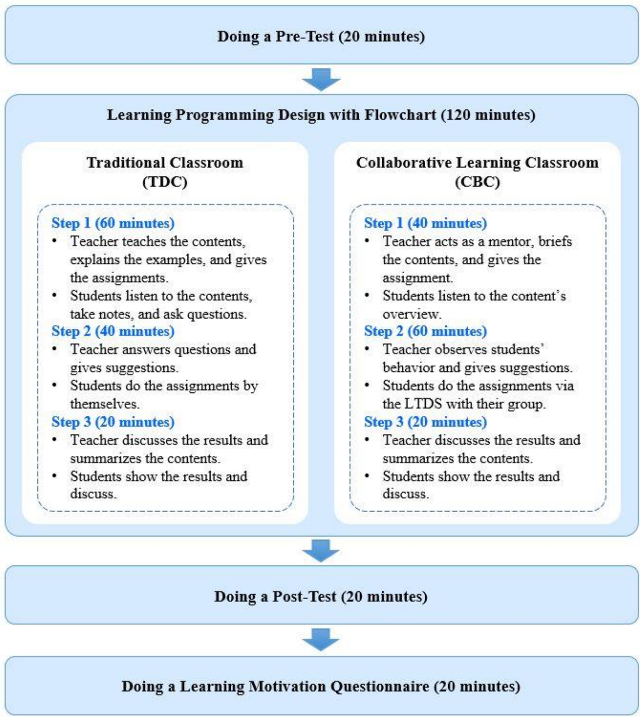

Fig. 4. The experimental procedures.

IV. Experimental Results

This study evaluates students' motivation and achievement in different learning environments and the relationship between the learning environment and individual preferences. The experimental results are explained below.

A. Results of Learning Achievement

The first experimental result evaluates students' learning achievement in a traditional classroom and a classroom with collaborative learning in a new environment. The post-test score is the dependent variable, while the learning environment group is the independent variable. Besides, the pre-test score is the covariance variable regarding the influence of prior knowledge with statistical significance of $t = 3.033$ and $p = 0.003$. The post-test scores of both groups are analyzed with one-way ANCOVA. Table 2 shows that the post-test scores of the students learning in the traditional and collaborative learning classrooms are statistically significant ($F = 21.242$, $p = 0.001$, $\eta^2 = 0.049$).

Table 2. Results of one-way ANCOVA on students' learning achievement in different learning approaches

Learning approaches	*n*	*M*	*SD*	*MS*	$F_{(1,408)}$	η^2
TDC	203	10.87	2.981	204.659	21.242*	0.049
CBC	207	12.29	3.220			

*$p < 0.05$

The results confirm that the learning achievement of the collaborative learning classroom ($M = 12.29$, $SD = 3.220$) is higher than that of the traditional classroom ($M = 10.87$, $SD = 2.981$). It suggests that students learning in the collaborative classroom accomplish a learning level higher than those learning in the traditional classroom.

The dimension of individual preferences is considered by defining the post-test score as the dependent variable, the student's individual preferences as the independent variable, and the pre-test score as the covariance variable. Table 3 shows the two-way ANCOVA results for the different individual preferences in each learning environment. No significant impact is found on the interaction between learning approaches and individual preferences ($F = 1.588$, $p = 0.208$, $\eta^2 = 0.004$). The significant effects are confirmed with learning approaches ($F = 14.141$, $p = 0.001$, $\eta^2 = 0.034$) and individual preferences ($F = 7.157$, $p = 0.008$, $\eta^2 = 0.017$) on the student's learning achievement as shown in Table 4.

Table 3. Results of two-way ANCOVA on students' learning achievement in different individual preferences

Sources	*SS*	*df*	*MS*	*F*	η^2
Learning approaches	83.776	1	83.776	14.141*	0.034
Individual preferences	42.402	1	42.402	7.157*	0.017
Learning approaches→ Individual preferences	9.406	1	9.406	1.588	0.004

*$p < 0.05$

Table 4. Descriptive data of students' learning achievement in the different individual preferences

Learning approaches	Individual preferences	*N*	*M*	*SD*
TDC	SDL	90	11.39	2.963
	CBL	113	10.46	2.963
CBC	SDL	85	13.25	3.047
	CBL	122	11.61	3.179

The learning achievement of the self-directed learning style ($M = 11.39$, $SD = 2.963$) is higher than the collaborative learning style ($M = 10.46$, $SD = 2.963$) in the traditional classroom. Similarly, in the collaborative learning classroom, the learning achievement of the self-directed learning style ($M = 13.25$, $SD = 3.047$) is higher than the collaborative learning style ($M = 11.61$, $SD = 3.179$). The result implies that the new learning strategy, with LTDS support, could enhance student achievement in self-directed and collaborative learning. Besides, it indicates that students who prefer self-directed learning could learn independently in any environment. Nevertheless, with a connotation, LTDS can encourage students who prefer self-directed learning to have higher achievement due to its features that encourage students to learn together, share opinions, solve problems together, and receive guidance from the teacher throughout their learning.

B. Results of Learning Motivation

The second experimental result involves evaluating the student's learning motivation in terms of intrinsic motivation (Intrinsic (IM), Self-Determination (SDM), and Self-Efficacy (SEM)) and extrinsic motivation (Career (CM)) by considering the learning environment and individual preferences. Firstly, the student's learning motivation is examined in different learning environments. Table 5 shows the result of a one-way MANOVA of learning motivation between traditional and collaborative learning classrooms. The motivation in the IM, SDM, SEM, and CM dimensions of traditional and collaborative learning classrooms are found with statistical significance ($F = 8.891$, $p = 0.003$, $\eta^2 = 0.021$; $F = 6.068$, $p = 0.014$, $\eta^2 = 0.015$; $F = 6.513$, $p = 0.011$, $\eta^2 = 0.016$; $F = 5.710$, $p = 0.017$, $\eta^2 = 0.014$, respectively).

Table 5. Results of one-way MANOVA on students' learning motivation

Motivation	Dimension	*MS*	$F_{(1,408)}$	η^2
Intrinsic	IM	6.559	8.891*	0.021
	SDM	4.068	6.068*	0.015
	SEM	4.606	6.513*	0.016
Extrinsic	CM	4.042	5.710*	0.014

*$p < 0.05$

Consequently, the mean values in Table 6 show the descriptive data on learning motivation in different learning environments; they indicate that students who learned in the collaborative learning classroom are more motivated to learn ($M = 4.111$, $SD = 0.845$; $M = 4.139$, $SD = 0.815$; $M = 4.141$, $SD = 0.840$; $M = 4.172$, $SD = 0.837$, respectively) than students who learn in the traditional classroom ($M = 3.858$, $SD = 0.873$; $M = 3.940$, $SD = 0.823$; $M = 3.930$, $SD = 0.842$; $M = 3.973$, $SD = 0.845$ respectively) in both intrinsic and extrinsic motivation in four dimensions. Therefore, using the LTDS to support the new learning approach could motivate students to learn how to analyze and design programming processes with intrinsic and extrinsic motivation. With the system's support, the students learn in groups, engage in group activities via the collaborative area, share their ideas via the chat box, and get feedback from the teacher during learning activities.

In terms of the relationship between learning environments and individual preferences, Table 7 shows the two-way MANOVA results of learning motivation in different learning environments and individual preferences with Levene's test $F_{(3,406)} = 0.167, 0.365, 0.357, 0.463$, and $p = 0.919, 0.779, 0.784, 0.708$ of the IM, SDM, SEM, and CM values, respectively. All motivation dimensions show no statistically significant interaction between learning approaches and individual preferences (F = 0.197, p = 0.658, η^2 = 0.001; F = 1.123, p = 0.290, η^2 = 0.003; F = 0.554, p = 0.457, η^2 = 0.001; F = 0.324, p = 0.569, η^2 = 0.001, respectively).

Table 6. Descriptive data on learning motivation in different learning approaches

Motivation	Dimension	Learning approaches	*n*	*M*	*SD*
Intrinsic	IM	TDC	203	3.858	0.873
		CBC	207	4.111	0.845
	SDM	TDC	203	3.940	0.823
		CBC	207	4.139	0.815
	SEM	TDC	203	3.930	0.842
		CBC	207	4.141	0.840
Extrinsic	CM	TDC	203	3.973	0.845
		CBC	207	4.172	0.837

Table 7. Two-way MANOVA results of learning motivation in different learning approaches and individual preferences

Motivation	Dimension	Source	*MS*	$F_{(1,406)}$	η^2
Intrinsic	IM	Learning approaches	6.311	8.669*	0.021
		Individual preferences	5.315	7.301*	0.018
		Learning approaches →Individual preferences	0.143	0.197	0.001
	SDM	Learning approaches	4.103	6.295*	0.015
		Individual preferences	8.137	12.540*	0.030
		Learning approaches →Individual preferences	0.732	1.123	0.003
	SEM	Learning approaches	4.581	6.566*	0.016
		Individual preferences	4.921	7.055*	0.017
		Learning approaches →Individual preferences	0.387	0.554	0.001
Extrinsic	CM	Learning approaches	4.042	5.728*	0.014
		Individual preferences	2.077	2.943	0.007
		Learning approaches →Individual preferences	0.229	0.324	0.001

*$p < 0.05$

Table 8 shows the motivation scores of students with a preference to learn in a group in the traditional classroom (M = 3.977, SD = 0.866; M = 4.104, SD = 0.766; M = 4.055, SD = 0.816; M = 4.058, SD = 0.843, respectively) and in the collaborative learning classroom (M = 4.190, SD = 0.840; M = 4.221, SD = 0.811; M = 4.207, SD = 0.852; M = 4.211, SD = 0.877, respectively) are more motivated than students who prefer self-directed learning in the traditional classroom (M = 3.709, SD = 0.864; M = 3.733, SD = 0.848; M = 3.771, SD = 0.851; M = 3.867, SD = 0.840, respectively) and in the collaborative learning classroom (M = 3.998, SD = 0.843; M = 4.021, SD = 0.812; M = 4.047, SD = 0.820; M = 4.115, SD = 0.780, respectively) in both learning environments in all dimensions of intrinsic and extrinsic motivation. The results suggest that learning via groups could promote both students with the preference for self-directed learning and group learning due to the characteristics of collaborative learning.

Table 8. Descriptive data on learning motivation in different individual preferences

Motivation Dimension	Learning Environment	Individual preferences	*n*	*M*	*SD*
IM	TDC	SDL	90	3.709	0.864
		CBL	113	3.977	0.866
	CBC	SDL	85	3.998	0.843
		CBL	122	4.190	0.840
SDM	TDC	SDL	90	3.733	0.848
		CBL	113	4.104	0.766
	CBC	SDL	85	4.021	0.812
		CBL	122	4.221	0.811
SEM	TDC	SDL	90	3.771	0.851
		CBL	113	4.055	0.816
	CBC	SDL	85	4.047	0.820
		CBL	122	4.207	0.852
CM	TDC	SDL	90	3.867	0.840
		CBL	113	4.058	0.843
	CBC	SDL	85	4.115	0.780
		CBL	122	4.211	0.877

V. Discussion

This study examines the effectiveness of a new learning approach that transforms the traditional classroom into an active learning environment by combining the usefulness of the thinking process model technique as a flowchart with online collaborative learning to promote the logical programming thinking of first-year students in information technology, computer science, or related fields from six universities in five regions of Thailand. This study's primary objective is to evaluate learning motivation and achievement on programming analysis and design among students learning in different classroom environments between the traditional and collaborative learning classrooms and those with other individual preferences between self-learning and group learning.

To answer RQ1, according to the experimental results, the one-way ANCOVA result indicates that students who learn in the new learning approach supported by LTDS demonstrate higher achievement in programming analysis and design than those who learn in the traditional classroom environment in terms of different learning environments (Table 2). It indicates that the use of collaborative learning and the thinking process model technique as a flowchart could support students' abilities to comprehend the problem, analyze the problem, decompose the program processes, design the program processes, use the appropriate flowchart symbols, define the correct navigation, and place the correct labels. During learning activities, the students could collaborate within their group and were encouraged to share ideas, help solve problems, suggest mistakes, and support each other. In contrast, the two-way ANCOVA result indicated that the collaborative learning classroom is ineffective for individual preferences. Students who prefer self-directed learning have higher mean scores than those who learn in groups (Table 3 and Table 4). Students who preferred self-directed learning could learn independently in any environment. With a connotation, LTDS can encourage students who prefer self-directed learning to have higher achievement due to its features that encourage students to learn together, share opinions, solve problems together, and receive guidance from the teacher throughout their learning.

Furthermore, the results coincide with previous studies [8–10, 25]. It indicates that collaborative learning can support students' high achievement and enjoyment because it promotes effective collaboration skills by permitting them to actively participate in the learning process, resulting in increased student performance as they share their knowledge and learn from one another. It influences student participation in collaborative learning, resulting in a superior and more comprehensive learning experience. Through student interactions, shared knowledge construction, and the development of higher-order thinking skills, online collaborative learning provides unique opportunities to increase learning motivation and achievement. This strategy emphasizes collaborative work, peer feedback, and participation. These elements foster students' sense of behavioral engagement in online collaborative learning and influence learning performance positively by encouraging active, motivating, and collaborative learning experiences, leading to effective outcomes such as improved academic performance and higher-order thinking skills.

Accordingly, adjusting the learning model to a new learning approach has contributed to the teacher's role, transforming them into mentors. The mentor can monitor the students' conduct, monitor their thinking in real time, and provide prompt feedback to the students. This outcome is advantageous for the children and teachers. During the learning activities, teachers can learn about the thinking skills of individual students and the entire classroom. It serves as a framework for modifying the material and instructional methods to enhance its appropriateness for student needs in the future.

In the meantime, to answer the RQ2, the one-way MANOVA test reveals that the students who learn in the collaborative learning classrooms supported by LTDS are more motivated in terms of intrinsic motivation, self-determination motivation, self-efficacy motivation, and career motivation (Table 5 and Table 6). The results indicated that a collaborative learning environment could promote intrinsic and extrinsic motivation by stimulating students' curiosity. In addition, the two-way MANOVA test also reveals that students who prefer to learn in groups have higher intrinsic and extrinsic motivation than those who prefer to learn by themselves (Table 8). There are no statistically significant differences in all dimensions of motivation when considering the interaction between learning approaches and individual preferences (Table 7). This result indicates that learning in a collaborative group can promote students' learning motivation in dimensions of intrinsic, self-determination, self-efficacy, and career because they can consult with their group members, brainstorm ideas, and discuss problems. It makes them feel less lonely to learn, dare to ask questions, and enjoy learning. The findings of this study align with the outcomes of other research investigations [6, 19, 20]. It suggests that the use of visual tools for the analysis and design of program processes can enhance students' motivation to study and foster the development of logical thinking abilities. Visuals facilitate students' comprehension of intricate programming ideas by offering structured and organized pictures that depict algorithms, processes, and procedures. Furthermore, it allows students to apply systematic thinking and problem-solving techniques via logical processes. As a result, flowcharts enhance understanding of the correlation between programming parts and facilitate improving code design and implementation.

The experimental results demonstrate that the new learning approach transforms the traditional classroom into an active learning environment by combining the usefulness of the thinking process model technique as a flowchart with online collaborative learning, thereby enhancing students' learning achievement and intrinsic and extrinsic motivation. There are several key messages of this study. Firstly, the thinking process model technique can enhance students' logical thinking abilities. It is attributed to the properties of flow charts, which function as visual depictions of algorithms. Presenting information in structured flow charts enables students to comprehend intricate programming concepts and augment comprehension of programming logic [19–21]. Secondly, an online collaborative learning environment facilitates student engagement in learning activities via collaborative endeavors without the limitation of place and time. It encourages students to reflect on their thoughts and develop social skills through group interaction. Such affects students' active involvement, resulting in a better and deeper learning experience [9, 10, 24]. Lastly, the study's results show that students who prefer to learn in groups are more motivated than those who prefer to learn independently. Students who prefer to learn independently have higher learning achievement than those who prefer to learn in groups. It shows that the online collaborative learning environment can support students learning independently even though they do not enjoy learning with others. Therefore, from the benefits of the new learning approach, it is postulated that it may facilitate students' enjoyment of learning the programming subject, alleviate any associated anxiety about the programming difficulty, and ultimately enhance their academic performance.

VI. Conclusion

According to the research results, this study highlights the synergistic benefits of combining online collaborative learning with the thinking process model. Online collaborative learning platforms could provide opportunities for students to engage in peer-to-peer interactions, which can foster social learning and promote the exchange of ideas. Besides, the thinking process model provides a structured approach to problem-solving that can guide collaborative activities and ensure that students are actively engaged in meaningful learning experiences. Consequently, it can be confirmed that integrating thinking process models into online collaborative learning environments can enhance students' learning achievements and motivation. Because it supports students in developing logical thinking by practicing decomposing problems into program steps and designing programs in the form of visuals through collaborative learning with their classmates. This function encouraged students to understand the sequence of problem-solving and programming steps, dare to share their opinions, and learn to solve problems from different perspectives.

The experiment results provide theoretical implications for researchers to indicate a new research direction. Integrating the thinking process model technique for constructivist learning environments could help create rich environments where students can engage in critical thinking and problem-solving, which is essential for programming. With the importance of using evidence-based instructional strategies in programming education, curriculum designers can incorporate thinking process model-based activities into programming courses, and instructors can use the thinking process model to guide collaborative learning sessions.

Although the experiment results show the success of creating a new learning approach for the computer programming course, there are several concerns to be addressed within further study. First, insert hints when students drag and drop the flowchart symbols in the collaborative designing area to suggest what to do next or correct the mistakes in real time. Second, it provides opportunities for students to practice alone and in groups. Third, study the system's characteristics to determine which features affect students' learning achievement and motivation to develop that feature into a strength that will support learning better. Lastly, applying an AI-based programming learning platform to the instructional methodology will augment the new classroom experience and promote the development of students' logical thinking abilities. That is to say, further work should ensure the effectiveness of the developed system by comparing learning performance between students receiving the LTDS and those receiving the existing programming learning systems.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Kannika Daungcharone designed the research, collected the data, analyzed the data, and wrote the manuscript. Associate Professor Dr. Patcharin Panjaburee assisted with the review of analytics tools and their reliability. Krittawaya Thongkoo collected data. All authors had approved the final version.

FUNDING

This research was funded by a grant from the National Research Council of Thailand (NRCT), RGNS 63-083.

REFERENCES

[1] D. Song, H. Hong, and E. Y. Oh, "Applying computational analysis of novice learners' computer programming patterns to reveal self-regulated learning, computational thinking, and learning performance," *Computers in Human Behavior,* vol. 120, pp. 1–9, February 2021. https://doi.org/10.1016/j.chb.2021.106746

[2] R. Scherer, F. Siddiq, and B. S. Viveros, "A meta-analysis of teaching and learning computer programming: Effective instructional approaches and conditions," *Computers in Human Behavior*, vol. 109, pp. 1–18, March 2020. https://doi.org/10.1016/j.chb.2020.106349

[3] K. Daungcharone, "Effects of blending digital games into traditional lecture-based learning on university students' programming learning achievement," in *Proc. the 18th European Conference on e-Learning,* pp. 151–159, November 2019, doi: 10.34190/EEL.19.070.

[4] F. C. O. Yang, H. M. Lai, and Y. W. Wang, "Effect of augmented reality-based virtual educational robotics on programming students' enjoyment of learning, computational thinking skills, and academic achievement," *Computers & Education*, vol. 195, pp. 1–22, December 2022. https://doi.org/10.1016/j.compedu.2022.104721

[5] T. Doleck, P. Bazelais, D. J. Lemay, A. Saxena, and R. B. Basnet, "Algorithmic thinking, cooperativity, creativity, critical thinking, and problem solving: exploring the relationship between computational thinking skills and academic performance," *Journal of Computers in Education,* August 2017, doi 10.1007/s40692-017-0090-9.

[6] Z. Hu, C. Fan, Q. Zheng, W. Wu, and B. Liu, "Asyncflow: A visual programming tool for game artificial intelligence," *Visual Informatics,* vol. 5, pp. 20–25, 2021. https://doi.org/10.1016/j.visinf.2021.11.001

[7] C. W. Liao, C.H. Chen, and S. J. Shih, "The interactivity of video and collaboration for learning achievement, intrinsic motivation, cognitive load, and behavior patterns in a digital game-based learning environment," *Computers & Education,* vol. 133, pp. 43–55, May 2019. https://doi.org/10.1016/j.compedu.2019.01.013

[8] M. Courtney, J. Costley, M. Baldwin, K. Lee, and M. Fanguy, "Individual versus collaborative note-taking: Results of quasi-experimental study on student note completeness, test performance, and academic writing," *The Internet and Higher Education,* vol. 55, pp. 1–14, October 2022. https://doi.org/10.1016/j.iheduc.2022.100873

[9] J. M. Ye, and J. Zhou, "Exploring the relationship between learning sentiments and cognitive processing in online collaborative learning: A network analytic approach," *The Internet and Higher Education,* vol. 55, pp. 1–11, 2022. https://doi.org/10.1016/j.iheduc.2022.100875

[10] N. S. Sabarillo, D. A. G. Sumalinog, J. M. S. Diaz *et al*., "Design, implementation, and evaluation of an online flipped classroom with collaborative learning model in an undergraduate chemical engineering course," *Education for Chemical Engineers*, vol. 43, pp. 58–72, April 2023. https://doi.org/10.1016/j.ece.2023.01.007

[11] A. Vahldick, P. R. Farah, M. J. Marcelino, and A. J. Mendes, "A block-based serious game to support introductory computer programming in undergraduate education," *Computers in Human Behavior Report,* vol. 2, pp. 1–12, 2020. https://doi.org/10.1016/j.chbr.2020.100037

[12] G. Polito and M. Temperini, "A gamified web based system for computer programming learning," *Computers and Education; Artificial Intelligence*, vol. 2, pp. 1–13, July 2021. https://doi.org/10.1016/j.caeai.2021.100029

[13] M. Roman-Gonzalez, J. C. Perez-Gonzalez, and C. Jimenez-Fernandez, "Which cognitive abilities underlie computational thinking? Criterion validity of the computational thinking test," *Computers in Human Behavior,* vol. 72, pp. 678–691, July 2017. http://dx.doi.org/10.1016/j.chb.2016.08.047

[14] B. A. Soufan, B. O. Bairkdar, E. A Soufan, and M. Samaan, "How do college course and materials affect students' logical thinking of the medical college at AI Baath University in Syria," *Eudacion Medica,* vol. 24, pp. 1–6, 2023. https://doi.org/10.1016/j.edumed.2023.100797

[15] K. Daungcharon, P. Panjaburee, and K. Thongkoo, "Implementation of mobile game-transformed lecture-based approach to promoting C programming language learning," *International Journal of Mobile Learning and Organisation*, vol. 14, pp. 236–254, January 2020, doi: 10.1504/IJMLO.2020.10026332.

[16] N. Bak, B. M. Chang, and K. Choi, "Smart Block: A visual block language and its programming environment for IoT," *Journal of Computer Languages,* vol. 60, pp. 1–19, October 2020. https://doi.org/10.1016/j.cola.2020.100999

[17] Y. Lin and D. Weintrop, "The landscape of block-based programming: characteristics of block-based environments and how they support the transition to text-based programming," *Journal of Computer Languages,* vol. 67, pp. 1–18, December 2021. https://doi.org/10.1016/j.cola.2021.101075

[18] K. Sigayret, A. Tricot, and N. Blanc, "Unplugged or plugged-in programming learning: A comparative experimental study," *Computers & Education,* vol. 184, pp. 1–14, July 2022. https://doi.org/10.1016/j.compedu.2022.104505

[19] K. Daungcharone and K. Thongkoo, "The effects of thinking process model technique on logical thinking skills influencing programming achievement," in *Proc. the 7th International Conference on Digital Arts, Media and Technology and 5th ECTI Northern Section Conference on Electrical, Electronic Computer and Telecommunications Engineering (NCON),* pp. 137–141, January 2022.

[20] T. Yokoyama, H. B. Axelsen, and R. Gluck, "Fundamentals of reversible flowchart languages," *Theoretical Computer Science,* vol. 611, pp. 87–115, 2016, http://dx.doi.org/10.1016/j.tcs.2015.07.046

[21] V. Dagiene and G. Stupuriene, "Bebras–a sustainable community building model for the concept based learning of informatics and computational thinking," *Informatics in Education,* vol. 15, pp. 25–44, May 2016, doi: 10.15388/infedu.2016.02.

[22] J. P. M. Miguel, C. S. D. Blas, F. A. Rodriguez, and A. E. G. Sipols, "Collaborative learning in management subjects to university students: A multi-level research to identify group profile, engagement and academic performance," *The International Journal of Management Education,* vol. 21, pp. 1–15, March 2023. https://doi.org/10.1016/j.ijme.2022.100762

[23] L. M. Serrano-Camara, M. Paredes-Velasco, C. M. Alcover, and J. A. Velazquez-Iturbide, "An evaluation of students' motivation in computer-supported collaborative learning of programming concepts," *Computers in Human Behavior,* vol. 31, pp. 499–508, February 2014. http://dx.doi.org/10.1016/j.chb.2013.04.030

[24] Z. Zhang, T. Liu, and C. B. Lee, "Language learners' enjoyment and emotion regulation in online collaborative learning," *System,* vol. 98, pp. 1–15, June 2021. https://doi.org/10.1016/j.system.2021.102478

[25] M. Pilotti, S. Anderson, P. Hardy, P. Murphy, and P. Vincent, "Factors related to cognitive, emotional, and behavioral engagement in the online asynchronous classroom," *International Journal of Teaching and Learning in Higher Education,* vol. 29, pp. 145–453, 2017.

[26] S. M. Glynn, P. Brickman, N. Armstrong, and G. Taasoobsirazi, "Science motivation questionnaire II :Validation with science majors and nonscience majors," *Journal of Research in Science Teaching,* vol. 48, pp. 1159–1176, December 2011, doi:10.1002/tea.20442.