International Journal of Advanced Research in Education and Society
e-ISSN: 2682-8138 | Vol. 7, No. 2, 123-138, 2025
https://asianscholarsnetwork.com/asnet-journals

Integrating Computational Thinking and Multimedia Pre-Training in Mobile Learning Application: Exploring Effects on Programming Problem-Solving Skills Among Matriculation Students

Mimi Zairul Mohmad Fuzi[1*], Wan Ahmad Jaafar Wan Yahaya[1]

[1] Universiti Sains Malaysia, Penang, Malaysia

*Corresponding Author: mimizairul@gmail.com

Received: 4 January 2025 | Accepted: 25 March 2025 | Published: 1 April 2025

DOI: https://doi.org/10.55057/ijares.2025.7.2.12

Abstract: *This study examines the effect of the integration of computational thinking and multimedia pre-training principles on programming problem-solving skills in learning applications for matriculation students. Using a qualitative approach, interviews were conducted with five participants to identify the factors that influence low achievement in learning programming. Findings show that uninteresting learning materials and lack of interaction with lecturers are the main challenges. However, the use of computational thinking helps students in deciphering problems and organizing the solution process more systematically. These findings show the great potential of pre-training multimedia applications in increasing student engagement and achievement in learning programming.*

Keywords: computational thinking, multimedia pre-training principle, programming learning, matriculation students, qualitative interviews

1. Introduction

In the rapidly growing digital age, the need for programming skills and computational thinking is becoming increasingly important (George-Reyes et al., 2021). Programming is not only the basis for the development of modern technology, but also shapes how individuals solve problems in a more systematic and efficient way (Sambe et al., 2021). However, low achievement in programming among students remains a significant issue (Sobral & Oliveira, 2021). Many students have difficulty mastering these skills, which in turn negatively affects their overall performance in computer science courses.

In this context, CT and multimedia pre-training principles appear as potential approaches to overcome this challenge. CT refers to the thought process that involves formulating a problem and solving it in a way that can be implemented by a computer (Wing, 2017). It covers concepts such as abstraction, decomposition, pattern recognition, and algorithms. Meanwhile, the principle of multimedia pre-training involves the preparation of learning materials in the names and characteristics to introduce important concepts before students begin deeper learning in multimedia lesson (Mayer, 2009). According to Laura-Ochoa & Bedregal-Alpaca (2022) CT is a potential approaches to improves student performance and motivation in programming courses. While Mayer (2017) states that multimedia pre-training principles can help manage

essential processing in computer-based multimedia instructional materials for better academic learning.

This study aims to investigate the effect of the integration of CT and multimedia pre-training principles in the application of programming learning on matriculation students' problem-solving skills. There are several factors that have been identified as contributing to low achievement in programming problem solving skills among matriculated students. These factors include a lack of basic understanding of programming concepts, lack of exposure to effective problem solving techniques, and the absence of suitable and interesting learning materials (Cheah, 2020).

The first objective of this study is to identify the factors that contribute to low achievement in programming problem solving skills among matriculation students through the integration of CT and multimedia pre-training principles in learning applications. In an effort to achieve this objective, this study will examine the academic background of the students, the teaching methods used, as well as the quality and effectiveness of the available learning materials. The data collected will be analyzed to identify the main patterns and factors that affect student performance in programming.

The second objective of this study is to analyze the level of involvement of matriculation students in programming learning activities after the introduction of CT principles and multimedia pre-training in learning applications. Student engagement is an important aspect of the learning process, as it is closely related to motivation, willingness to learn, and the ability to retain learned information (Amin Husni et al., 2022). This study will use qualitative methods to assess the level of student engagement, including classroom observations, interviews with students and teachers, as well as analysis of student learning journals.

There are several reasons that motivate this study. First, low student engagement in programming learning activities is often associated with a lack of interest and motivation (Cao et al., 2022). Students may feel that programming is a difficult subject and not relevant to their daily life (Figueiredo & Garcia-Penalvo, 2021). By introducing CT and multimedia pre-training, it is hoped that students can see the connection between programming and real-world situations, further increasing their interest and motivation.

Second, low achievement in programming is often due to ineffective teaching approaches (Sobral & Oliveira, 2021). Traditional teacher-centered teaching methods may not be sufficient to meet the needs of students with diverse learning styles (Ghafar, 2023). The integration of CT and multimedia pre-training can provide a more student-centered approach, allowing them to learn at their own pace and in their own way. This can help improve their understanding of programming concepts and improve their problem solving skills.

Third, the absence of interesting and effective learning materials is also a factor that contributes to low achievement in programming (Da Silva Garcia et al., 2021). Learning materials that use various forms of media can help attract students' interest and increase their involvement in the learning process (Taqiyuddin et al., 2022). The principle of multimedia pre-training allows students to gain early exposure to important concepts before they engage in more complex learning activities. This can help them build a solid foundation and facilitate their understanding of more difficult concepts.

In this study, data will be collected through various qualitative methods, including interviews, observations, and document analysis. Interviews will be conducted with students and lecturers to obtain their views on the effectiveness of CT and multimedia pre-training in learning programming. Class observations will be conducted to assess the level of student involvement in learning activities. Document analysis will involve studying the learning materials used and student performance in tests and assignments.

In conclusion, this study is expected to provide a deeper understanding of the factors that contribute to low achievement in programming among matriculation students, as well as the effectiveness of CT and multimedia pre-training in improving their engagement and performance. By identifying and analyzing these factors, it is hoped that more effective teaching strategies can be developed to help matriculation students master programming skills and achieve their full potential in this field.

Significance and Contribution

This study has several significant and important contributions to various parties including the government, community, industry, as well as the addition of new knowledge in the field of programming education. This study can help the government in formulating and implementing more effective education policies, especially in the field of computer science and information technology. By identifying the factors that contribute to low achievement in programming, the government can take proactive steps to improve the quality of education at the matriculation level. This includes the introduction of a more appropriate curriculum, the provision of quality teaching materials, as well as training for teachers to integrate CT and multimedia pre-training principles in their teaching (Zha et al., 2020).

This study also benefits the education community, especially among students and teachers. Students can enjoy a more interesting and effective learning experience, which in turn can increase their interest and motivation in the field of programming (Efecan et al., 2021). Teachers can use the findings of this study to improve their teaching methods, further helping students achieve better performance. Indirectly, this will produce a generation that is more tech-savvy and ready to face the challenges of the digital age.

The information and communication technology industry will benefit from this study through the production of more quality graduates ready to work in this growing field. Graduates who have good problem solving skills and are able to apply CT will be valuable assets to technology companies (Rosita et al., 2020). This will also increase the competitiveness of the country's industry at the global level.

This study contributes to the addition of new knowledge in the field of programming education by providing empirical data on the effectiveness of CT and multimedia pre-training. The findings from this study can be used as a reference for future studies as well as help academics and researchers in developing more innovative and effective teaching theories and models. Overall, this study not only benefits students and teachers, but also contributes to improving the quality of education and the development of the country's information technology industry.

Scope of the Research

The scope of this study includes 18-year-old matriculated students who follow the Four Semester System majoring in Science and taking Computer Science subjects. The justification for the selection of this scope is because students in this category are currently in a critical

phase in their education, where they are preparing to continue their studies at the university level in fields related to science and technology.

The selection of matriculation students is appropriate because they are going through a period of intensive and in-depth learning, which requires strong problem-solving skills as well as mastery of important programming concepts. The Four Semester System provides sufficient time for students to master CT and the principles of multimedia pre-training, and enables a more comprehensive assessment of the impact of these two approaches on their learning.

By focusing on students majoring in Science who take Computer Science subjects, this study can provide a more specific and relevant view on how the proposed learning approach can help improve student achievement and involvement in the field of programming.

Article Structuring

This article will be organized according to an orderly structure, starting with an introduction that introduces the title of the study, followed by a methodological section that explains the research methods used. Next, the results of the study will be presented together with data analysis. This article will end with discussion, conclusions, and recommendations for future research.

2. Literature Review

Programming education plays an important role in providing students with the problem-solving and creative thinking skills needed in this digital age. In this context, the integration of CT and multimedia pre-training principles has emerged as a promising approach to improve programming learning, especially among matriculation students (Mohmad Fuzi & Wan Yahaya, 2024). This study aims to examine the effect of this integration on programming problem solving skills among matriculation students.

Basic concepts in programming and the importance of problem-solving skills become focal points in the programming education literature. Programming is not just about writing code, but also involves the skills to identify problems, formulate appropriate algorithms, and solve problems systematically (Sambe et al., 2021). These skills are important in preparing students to face increasingly complex technological challenges.

CTrefers to a structured and systematic thinking process in solving problems using computer concepts (Bilbao et al., 2021). This thinking includes the use of abstraction, problem decomposition, pattern identification, and the use of algorithms, all of which provide a solid foundation for learning to program. In an educational context, this approach helps change the way students understand and solve problems with a more organized approach. The principles of multimedia pre-training, on the other hand, bring elements of early exposure to the concepts to be learned (Mayer, 2021). Media such as video, animation, and interactive simulations can reinforce understanding and prepare students before they engage in deeper learning activities (Laksmi et al., 2021). The use of this media not only increases understanding, but can also increase student motivation (Katona et al., 2023) in learning programming.

Cognitive Theory of Multimedia Learning (CTML) introduced by Richard Mayer is a relevant theory in this context. This theory emphasizes the importance of delivering information through media that combines visual and auditory aspects, by associating new information with existing information in short-term memory (Keshavarz et al., 2022). By using CTML principles in

learning programming, the teaching process can be more effective by increasing the depth of understanding and retention of student information.

Previous studies have shown that CT can have a positive impact in improving students' programming problem-solving skills (Ma et al., 2021). The multimedia pre-training principles used, based on CTML, also have a positive impact on students' understanding of multimedia learning materials (Kaye, 2019). However, the need for further studies remains to investigate the actual impact of the integration of CTand multimedia pre-training principles, based on CTML on the overall academic performance of students, especially at the matriculation level.

Factors such as cognitive abilities (Shi & Qu, 2021), motivation (Howard et al., 2021), and learning experience (Araya et al., 2022) also play an important role in determining student achievement in programming. This study focuses on how these factors can affect the effectiveness of the teaching approach used at the matriculation level. Teaching models that are effective in improving programming skills also need attention. These models not only involve effective delivery techniques, but also appropriate evaluation strategies to measure student achievement in learning programming.

In the context of teaching programming at the matriculation level, there is an urgent need to identify specific challenges and needs in ensuring effective learning. The integration of CTand multimedia pre-training principles, underpinned by CTML, is expected to overcome these challenges by providing a more dynamic and relevant approach to learning programming. Related previous studies are important to provide a solid and relevant theoretical foundation for the research approach used in this study. By including an appropriate theoretical framework, including the CTML theory, and linking previous studies with the research objectives, this study is expected to provide a more robust and consistent guide in understanding the issues involved in learning programming at the matriculation level.

3. Methodology

This study uses qualitative research methods and involves interviews with five participants. The research design used is 'Basic Qualitative Inquiry'. This design was chosen because it allows the researcher to explore a complex and deep phenomenon, as well as understand the subjective experiences and views of the participants in a detailed context (Cresswell, 2014). In accordance with the objective of the study to identify the factors that influence low performance in programming problem solving skills among matriculation students, the qualitative approach allows for a deeper exposure to these aspects.

This study uses interviews as the main method to collect data. Structured interviews were used to obtain participants' views on their experiences in learning programming after the introduction of CT and multimedia pre-training. Informal observation is also done to strengthen the understanding of the learning context that takes place.

The rationale for choosing this method is because interviews allow the researcher to gain an in-depth view of the participants' experiences and perceptions of the integration of CT and multimedia pre-training in learning programming. This approach is in line with the objective of the study to analyze the level of student engagement and the factors that affect their achievement.

Sampling in this study uses the purposive sampling method, where participants are selected based on characteristics relevant to the study (Chua, 2011), such as experience in programming and involvement in learning. The target population is 18-year-old matriculation students taking computer science subjects. A sample size of five people was chosen to ensure diversity in their perspectives and experiences in the context under study.

Data analysis techniques involve manual coding and thematic analysis using Microsoft Word. This approach was chosen because it allowed the researcher to detect emerging interview patterns and identify key themes related to this study. Coding helps in categorizing the data according to themes that have been set before, while thematic analysis helps in developing a deep understanding of the issues arising from the data.

This study has considered research ethics by ensuring permission and consent from participants before conducting interviews. Safety, confidentiality, and the interests of participants are also prioritized in every research process. The certainty of the data is guaranteed through the process of data triangulation, where the use of various data sources such as interviews and observations helps to strengthen the reliability of the data. The use of manual methods in coding and thematic analysis also ensures the validity of the data analysis process.

Overall, the qualitative research method used in this study allows researchers to deepen the experiences and subjective views of matriculation students in the context of programming learning. By using an appropriate approach for data collection and analysis, this study is expected to provide a deeper and more relevant insight into the issues being studied.

4. Findings

In this study, the main findings related to the use of CT and multimedia pre-training principles in improving programming problem solving skills among matriculation students were identified through interviews conducted with five participants. This study focuses on how these two elements can help increase student engagement and performance in learning programming.

1) Lack of Interactive and Interesting Learning Materials

The first participant emphasized that the learning materials used in class were less interactive and only in the form of notes. This makes it difficult for students to feel motivated and fully engaged in learning programming. This situation reflects the need for more interesting and interactive materials that can stimulate students' interest.

2) Inadequate Learning Materials

The second participant stated that the learning materials provided were not sufficient to help them understand the concepts in solving programming problems. The lack of comprehensive material makes it difficult for students to achieve a deep understanding of the subject.

3) Uninteresting Learning Materials

The third participant repeated a similar sentiment by emphasizing that the digital learning materials provided were still in the form of notes and were not interesting. Although digital, the material lacks interactive elements that can improve their learning experience.

4) Lack of Guidance Outside the Classroom

The fourth participant complained about the lack of time allocated by the lecturer for out-of-class discussion sessions. Students feel that they need more guidance and opportunities

to ask questions outside of formal class time to truly understand and master programming concepts.

5) No Specific Techniques

The fifth participant felt that there were no specific techniques taught to master solving programming problems. They just follow the sample questions and answers given by the lecturer without understanding the techniques underlying the problem solving.

6) Use of Computational Thinking

When asked about how CT helps in learning programming, all participants admitted that it does help. CT allows them to more easily identify inputs, processes, and outputs in programming problems. They are also able to decompose problems into smaller components and solve them separately. This is very helpful in understanding and solving more complex problems.

7) Engagement in Learning Activities

The first participants reported active involvement in the programming challenges and exercises provided by the learning application. The app offers a variety of interesting and challenging challenges, keeping students interested and engaged. The second participant stated that the app's friendly user interface made it easy for them to access and navigate various programming exercises, allowing for a more flexible learning schedule.

8) Learning App Features

A third participant appreciated the customizable learning path features and topic marking in the app, which allowed them to tailor the learning experience to their interests and goals. The fourth participant liked the interactive tutorials and hands-on features in the application, which made learning programming concepts more interesting and practical. The fifth participant highlighted the benefits of the offline mode in the application, allowing them to download challenges and learning materials ahead of time and continue learning even without Internet access.

9) Challenges in Using CT and Multimedia Pre-training

There are several challenges identified by the participants in using CT and multimedia pre-training in learning programming. The first participant stated that too much information in learning applications can be confusing, and that they need to understand CT before they can apply it in problem solving. The second participant faced challenges in understanding how to develop CT concepts for students with different skill levels. A third participant felt that balancing the theoretical aspects of CT with the practical skills of programming can be challenging.

The fourth participant stated that they need to use the application repeatedly to master programming concepts, which can be time-consuming and require a high level of commitment. The fifth participant suggested that the application should provide a mark or score to track their progress, which could help in measuring development and self-motivation.

10) Findings and Implications

Findings from this study show that the integration of CT and multimedia pre-training in programming learning applications has great potential to improve programming problem-solving skills among matriculation students. However, there are several challenges that need

to be overcome to ensure the full effectiveness of this approach. Learning materials need to be more interactive and interesting to increase student engagement. In addition, lecturers need to provide more out-of-class guidance to help students understand difficult concepts. Learning applications also need to provide features that can track student progress to improve their motivation and understanding.

This study emphasizes the importance of using CT and multimedia pre-training in learning programming and identifies factors that need to be improved to achieve better results among matriculation students. This finding is expected to provide guidance to educators and learning application developers in an effort to improve the effectiveness of learning programming.

5. Main Results and Observations

Based on the findings in this study, there are several main results and observations that can be formulated. Qualitative analysis through interviews with five participants shows various aspects that affect the performance and involvement of students in learning programming using applications that integrate CT and multimedia pre-training.

1) Lack of Interactive and Interesting Learning Materials

Theme: Non-Interactive and Uninteresting Learning Materials

Coding dan Theme:

Coding	Theme
Learning materials in the form of notes	Non-Interactive Learning Materials
Uninteresting	Uninteresting Learning Materials

Description:

The first and third participants complained that the learning material they used was only in the form of notes and not interesting, even though it was in digital form. This reflects the need for more interactive and engaging materials to stimulate students' interest and increase their engagement in learning programming.

2) Inadequate Learning Materials

Theme: Lack of Comprehensive Learning Materials

Coding dan Theme:

Coding	Theme
Inadequate learning materials	Lack of Comprehensive Learning Materials

Description:

The second participant stated that the learning materials provided were not sufficient to help them understand the concepts in solving programming problems. This shows the need for more comprehensive learning materials to help students achieve a deep level of understanding.

3) Lack of Out-of-Class Guidance

Theme: Lack of Out-of-Class Guidance

Coding dan Theme:

Coding	Theme
Lack of Out-of-Class Guidance	Lack of Out-of-Class Guidance

Description:
The fourth participant complained about the lack of time allotted by the lecturer for out-of-class discussion sessions. Students need more guidance and opportunities to ask questions outside of formal class time to truly understand and master programming concepts.

The fourth participant complained about the lack of time allotted by the lecturer for out-of-class discussion sessions. Students need more guidance and opportunities to ask questions outside of formal class time to truly understand and master programming concepts.

4) No Specific Techniques

Theme: No specific techniques

Coding dan Theme:

Coding	Theme
No specific techniques	Lack of Specific Techniques in Problem Solving

Description:
The fifth participant stated that there are no specific techniques taught to master solving programming problems. They just follow the sample questions and answers given by the lecturer without understanding the techniques underlying the problem solving.

5) Use of Computational Thinking

Theme: Computational Thinking Helps Learning

Coding dan Theme:

Coding	Theme
Identifying the inputs, processes, and outputs	Computational Thinking Aids Learning
Breaking problems down into smaller components	Computational Thinking Aids Learning
Organizing programming problems systematically	Computational Thinking Helps Learning
Need to understand before being able to apply	Computational Thinking Helps Learning

Description:
All participants acknowledged that computational thinking helps in learning programming. Computational thinking allows them to more easily identify inputs, processes, and outputs in programming problems, as well as break problems into smaller components and solve them separately. However, the fifth participant stated that they need to understand computational thinking before being able to apply it in problem solving.

6) Involvement in Learning Activities

Theme: Active Engagement in Learning Applications

Coding and Theme:

Coding	Theme
Active in programming challenges	Active Engagement in Learning Applications
User-friendly interface	Active Engagement in Learning Applications

Customizable learning path	Active Engagement in Learning Applications
Interactive tutorials and hands-on	Active Engagement in Learning Applications
Offline Mode Active	Active Engagement in Learning Applications

Description:
The first, second, third, fourth, and fifth participants reported active engagement in the programming challenges and exercises provided by the learning application. A friendly user interface and flexibility in learning schedules help them stay engaged. Customizable learning paths and interactive tutorials and hands-on features improve their learning experience. Offline mode allows students to continue learning even without internet access.

7) Challenges in Using Computational Thinking and Multimedia Pre-Training
Theme: Challenges in Computational Thinking and Multimedia Pre-Training

Coding dan Theme:

Coding	Theme
Too much information	Challenges in Computational Thinking and Multimedia Pre-training
Concept scaffolding for various skill levels	Challenges in Computational Thinking and Multimedia Pre-training
Balancing the theoretical and practical aspects	Challenges in Computational Thinking and Multimedia Pre-training
Repeated application use	Challenges in Computational Thinking and Multimedia Pre-training
Pre-Exercise score system requirements	Challenges in Computational Thinking and Multimedia Pre-training

Description:
Participants identified several challenges in using computational thinking and multimedia pre-training in learning programming. There is too much information that can be confusing, and participants need to understand computational thinking before they can apply it. Scaffolding concepts for students with varying skill levels is also a challenge. Balancing the theoretical and practical aspects and the need to use the application repeatedly also add to the difficulty. A fifth participant suggested the need for a scoring system to track their progress.

Findings from this study show that the integration of computational thinking and multimedia pre-training in programming learning applications has great potential to improve programming problem-solving skills among matriculation students. However, there are several challenges that need to be overcome, including the lack of interactive and engaging learning materials, the lack of out-of-class guidance, and the need for specific techniques in problem solving. Additionally, it is important to ensure that computational thinking and multimedia pre-training are delivered in a way that is accessible to students of various skill levels and abilities. This finding is expected to provide guidance to educators and learning application developers in an effort to improve the effectiveness of learning programming.

Model Name: Model of Computational Thinking and Multimedia Pre-training Integration in Programming Problem Solving Learning for Matriculation Students
Based on the objectives of this study, a model was developed to describe the relationship between the identified factors, student engagement, and learning effectiveness.

Model Components:
This model has three main interrelated components, which are the factors contributing to low achievement, the level of student engagement, and learning outcomes in the application of programming problem solving.

1) Factors Contributing to Low Achievement:

a) Uninteresting Learning Materials:
Less interactive (Coding: Uninteresting Learning Materials)
Notes only (Participant 1, 3)

b) Inadequate Learning Materials:
Not enough to understand the concept (Coding: Inadequate Learning Materials)
There is no specific technique (Participant 2, 5)

c) Limited Interaction with Lecturers:
No time for discussion outside of class (Coding: Limited Interaction with Lecturer)
Participant 4

2) Level of Student Involvement:

a) Active Involvement:
Interesting programming challenges and exercises (Coding: Active Engagement)
Offline mode that facilitates learning (Participant 1, 5)

b) Learning Path Suitability:
Adaptable learning path (Coding: Learning Path Adaptability)
Marking topics according to interests and learning goals (Participant 3)

c) Practical Learning Experience:
An interactive tutorial that provides a practical learning experience (Coding: Practical Learning Experience) Participant 4

3) Learning Outcomes:

a) Understanding and Application of Computational Thinking:
Facilitating identification of inputs, processes, and outputs of programming problems (Coding: Computational Thinking)
Decomposing a problem into smaller problems (Participant 1, 2)
Help organize problems systematically (Participant 4)
Help solve programming problems (Participant 5)

b) Challenges in Computational Thinking and Multimedia Applications:
Too much information in an application (Coding: Application Challenge)
Understand the concept of computational thinking first (Participant 1)
Balancing theoretical and practical aspects (Participant 3)
Repeated application use (Participant 4)
Requirements for progress scores (Participant 5)

Model Description:

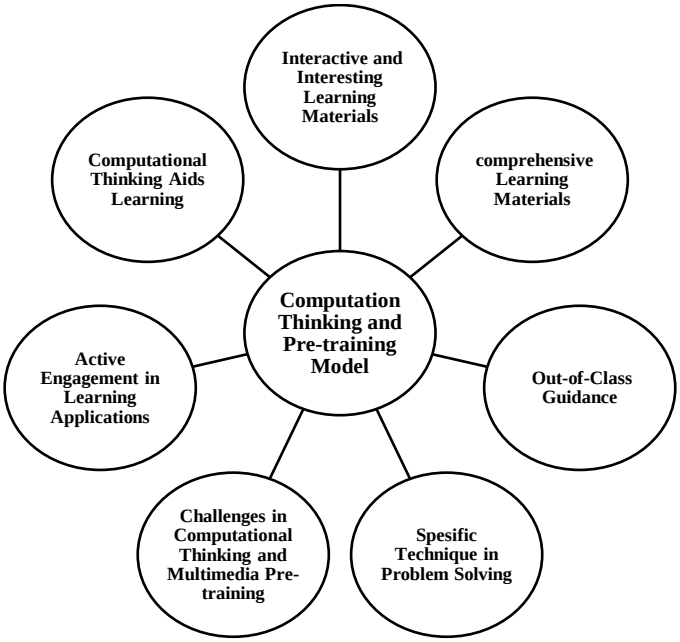

Figure 1: Model of Computational Thinking and Multimedia Pre-training Integration in Programming Problem Solving Learning for Matriculation Students

This model shows how factors contributing to low achievement affect the level of student involvement in learning programming and subsequently learning outcomes. The integration of computational thinking and multimedia pre-training in programming learning applications can improve matriculation students' problem-solving skills, but there are challenges that need to be overcome to achieve maximum effectiveness. This model can be a guide for educators and application developers in an effort to improve programming learning approaches.

6. Conclusion

This study aims to examine the effects of the integration of computational thinking (CT) and multimedia pre-training principles on programming problem-solving skills in learning applications for matriculation students. The following is a summary of the main findings, a discussion of the findings, as well as the contribution of this study to the field of study.

Analysis of findings from interviews with five matriculation participants has identified several important factors that influence low achievement in programming problem-solving skills. These factors include uninteresting learning materials, lack of adequate learning materials, and limited interaction with lecturers outside of class time. However, through learning applications that provide interesting and interactive challenges and customizable learning paths, students can be actively involved in learning programming.

The main findings show that less interactive and uninteresting learning materials are one of the main factors contributing to low achievement among students. This is in line with the literature that suggests that interesting learning materials such as videos, animations, and interactive simulations can increase student motivation and understanding (Mayer, 2021; Laksmi et al., 2021). For example, one of the participants stated that the current learning materials are less interesting and only consist of notes.

The use of computational thinking in learning programming is also proven to help students break down problems into smaller parts and organize the solution process systematically. However, there are challenges in understanding the concept of computational thinking and balancing between theory and practical application, which can affect the impact of its use. This study found that the use of computational thinking helps students in solving programming problems more systematically. This supports previous studies that show that CT can improve students' problem-solving skills (Bilbao et al., 2021; Ma et al., 2021). The use of CT allows students to break down problems into smaller problems and solve them one by one, which is a more structured and effective approach.

This finding is directly related to the objective of the study to identify factors that contribute to low achievement in programming problem-solving skills among matriculation students, as well as analyze their level of involvement in programming learning activities after the introduction of computational thinking and multimedia pre-training in learning applications. In particular, the integration of computational thinking and multimedia pre-training in learning applications is able to improve the student's learning experience by providing interesting and interactive learning materials and enabling the adaptation of learning paths according to individual interests and goals. However, the need for more interesting learning materials and more careful planning in conveying the concept of computational thinking is an important aspect that needs to be paid attention to.

This study has outlined some key findings that highlight the importance of the integration of computational thinking (CT) and multimedia pre-training principles in improving programming problem-solving skills among matriculation students. Through the interviews conducted, several key themes were identified, including the need for more interesting learning materials, the effectiveness of using CT, and challenges in balancing theory and practical application. However, this study also found that there are challenges in balancing between theory and practical application of computational thinking. Study participants faced difficulties in balancing theoretical aspects with practical programming skills, indicating the need for a more dynamic and relevant teaching approach. This reflects the relevance of the Cognitive Theory of Multimedia Learning (CTML) which emphasizes the importance of delivering information through media that combines visual and auditory aspects (Keshavarz et al., 2022).

This study contributes to the field of study by providing empirical evidence about the positive effects of the integration of computational thinking and multimedia pre-training principles in learning programming. By supporting CTML theory and relating previous studies to the findings of this study, it provides stronger guidance for educators and researchers in improving teaching approaches and programming learning materials. This study also underlines the need for a more interactive and relevant teaching approach, which can increase student engagement and performance in learning programming.

In conclusion, this study has achieved the objective to identify factors that contribute to low achievement and to analyze the level of student involvement in programming learning activities after the introduction of CT and multimedia pre-training principles. The themes identified show that the integration of this approach has the potential to have a positive impact on the learning process of matriculation students, thus contributing to their increased performance and engagement in programming learning, especially for matriculation students who face challenges in learning.

References

Amin Husni, N. H., Jumaat, N. F., & Tasir, Z. (2022). Investigating Student's Cognitive Engagement, Motivation and Cognitive Retention in Learning Management System. *International Journal of Emerging Technologies in Learning (IJET)*, *17*(09), 184–200. https://doi.org/10.3991/ijet.v17i09.29727

Araya, I., Beas, V., Stamulis, K., & Allende-Cid, H. (2022). Predicting student performance in computing courses based on programming behavior. *Computer Applications in Engineering Education*, *30*(4), 1264–1276. https://doi.org/10.1002/cae.22519

Bilbao, J., García, O., Rebollar, C., Bravo, E., & Varela, C. (2021). Skills, Attitudes and Concepts of the Computational Thinking. *EARTH SCIENCES AND HUMAN CONSTRUCTIONS*, *1*, 71–76. https://doi.org/10.37394/232024.2021.1.12

Cao, C., Zhao, X., & Hopfgartner, F. (2022). Enhancing Chinese International Students' Motivation and Engagement in a STEM Subject in UK Higher Education: A Case of Gamification in Programming. *Proceedings of the 14th International Conference on Education Technology and Computers*, 6–11. https://doi.org/10.1145/3572549.3572551

Cheah, C. S. (2020). Factors contributing to the difficulties in teaching and learning of computer programming: A literature review. *Contemporary Educational Technology*, *12*(2), 272. https://doi.org/10.30935/cedtech/8247

Chua, Y. P. (2011). *Kaedah dan Statistik Penyelidikan: Kaedah Penyelidikan* (2nd Editio). McGraw-Hill Education.

Cresswell, J. W. (2014). *Research Design : Qualitative, Quantitative and Mix Methods Approaches* (4th Editio). SAGE Publications, Inc.

Da Silva Garcia, F. W., Da Costa Carvalho, E., & Bezerra Oliveira, S. R. (2021). Use of active methodologies for the development of a teaching plan for the algorithms subject. *2021 IEEE Frontiers in Education Conference (FIE)*, 1–9. https://doi.org/10.1109/FIE49875.2021.9637482

Efecan, C. F., Sendag, S., & Gedik, N. (2021). Pioneers on the Case for Promoting Motivation to Teach Text-Based Programming. *Journal of Educational Computing Research*, *59*(3), 453–469. https://doi.org/10.1177/0735633120966048

Figueiredo, J., & Garcia-Penalvo, F. (2021). Teaching and Learning Tools for Introductory Programming in University Courses. *2021 International Symposium on Computers in Education (SIIE)*, 1–6. https://doi.org/10.1109/SIIE53363.2021.9583623

George-Reyes, C. E., Rocha Estrada, F. J., & Glasserman-Morales, L. D. (2021). Interweaving Digital Literacy with Computational Thinking. *Ninth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'21)*, 13–17. https://doi.org/10.1145/3486011.3486412

Ghafar, Z. N. (2023). The Teacher-Centered and the Student-Centered: A Comparison of Two Approaches. *International Journal of Arts and Humanities*, *1*(1), 18–23. https://doi.org/10.61424/ijah.v1i1.7

Howard, J. L., Bureau, J. S., Guay, F., Chong, J. X. Y., & Ryan, R. M. (2021). Student Motivation and Associated Outcomes: A Meta-Analysis from Self-Determination Theory. *Perspectives on Psychological Science*, *16*(6), 1300–1323. https://doi.org/10.1177/1745691620966789

Katona, B., Venkataragavan, J., Nina, E., Ulrika, B., & Björn, O. (2023). Use of Visual Learning Media to Increase Student Learning Motivation. *World Psychology*, *1*(3), 89–105. https://doi.org/10.55849/wp.v1i3.381

Kaye, M. (2019). Teaching Complex Content in OT Education: Using the Multimedia Principle of Pretraining to Foster Meaningful Learning. *American Journal of Occupational Therapy*, *73*(4_Supplement_1). https://doi.org/10.5014/ajot.2019.73s1-po3028

Keshavarz, H., Fallahnia, S., & Hamdi, F. (2022). How university instructors apply the design principles for electronic courses: a comparative study based on Richard Mayer's model on multimedia learning. *The International Journal of Information and Learning Technology*, *39*(4), 319–339. https://doi.org/10.1108/IJILT-11-2021-0173

Laksmi, N. K. P., Yasa, I. K. A., & Mirayani, K. A. M. (2021). The use of animation video as learning media for young learners to improve EFL students' motivation in learning English. *Lingua*, *17*(1), 42–52. https://doi.org/10.34005/lingua.v17i1.1378

Laura-Ochoa, L., & Bedregal-Alpaca, N. (2022). Incorporation of Computational Thinking Practices to Enhance Learning in a Programming Course. *International Journal of Advanced Computer Science and Applications*, *13*(2). https://doi.org/10.14569/IJACSA.2022.0130224

Ma, H., Zhao, M., Wang, H., Wan, X., Cavanaugh, T. W., & Liu, J. (2021). Promoting pupils' computational thinking skills and self-efficacy: a problem-solving instructional approach. *Educational Technology Research and Development*, *69*(3), 1599–1616. https://doi.org/10.1007/s11423-021-10016-5

Mayer, R.E. (2017). Using multimedia for e-learning. *Journal of Computer Assisted Learning*, *33*(5), 403–423. https://doi.org/10.1111/jcal.12197

Mayer, Richard E. (2009). *Multimedia Learning* (2nd ed). Cambridge University Press.

Mayer, Richard E. (2021). Evidence-based principles for how to design effective instructional videos. *Journal of Applied Research in Memory and Cognition*, *10*(2), 229–240. https://doi.org/10.1016/j.jarmac.2021.03.007

Mohmad Fuzi, M. Z., & Wan Yahaya, W. A. J. (2024). Empowering Problem-solving in Computer Science: A Need Analysis for a Computational Thinking Mobile Learning Application. *Asian Journal of Research in Education and Social Sciences*. https://doi.org/10.55057/ajress.2024.6.1.37

Rosita, Ekohariadi, Anifah, L., Rijanto, T., Munoto, & Nurlaela, L. (2020). Learning Strategies to Improve Computational Thinking in Problem Solving. *Proceedings of the 2nd International Conference Innovation in Education (ICoIE 2020)*. https://doi.org/10.2991/assehr.k.201209.233

Sambe, G., Drame, K., & Basse, A. (2021). Towards a Framework to Scaffold Problem-solving Skills in Learning Computer Programming. *Proceedings of the 13th International Conference on Computer Supported Education*, 323–330. https://doi.org/10.5220/0010446503230330

Shi, Y., & Qu, S. (2021). Cognitive Ability and Self-Control's Influence on High School Students' Comprehensive Academic Performance. *Frontiers in Psychology*, *12*. https://doi.org/10.3389/fpsyg.2021.783673

Sobral, S., & Oliveira, C. (2021). *Predicting students' performance in introductory programming courses: A literature review*. 7402–7412. https://doi.org/10.21125/inted.2021.1485

Taqiyuddin, H., Al Idrus, S., & Wijarnaka, S. (2022). Mobile-Based Learning Media Using Open Ended Learning Approaches to Increasing Students' Interest in Basic Programming Subjects of Vocational Schools. *Letters in Information Technology Education (LITE)*, *5*(2), 44. https://doi.org/10.17977/um010v5i22022p44-50

Wing, J. M. (2017). Computational thinking's influence on research and education for all Influenza del pensiero computazionale nella ricerca e nell'educazione per tutti. *Italian Journal of Educational Technology*, *25*(2), 7–14. https://doi.org/10.17471/2499-4324/922

Zha, S., Jin, Y., Moore, P., & Gaston, J. (2020). Hopscotch into Coding: Introducing Pre-Service Teachers Computational Thinking. *TechTrends*, *64*(1), 17–28. https://doi.org/10.1007/s11528-019-00423-0