Systematic Literature Review of Integrating Computational Thinking into Mathematics Education

Herman Yu-Hin LEUNG
Panel Head and Curriculum Developer of ICT & STEM Education, Hong Kong;
Head of Teacher Professional Development, Hong Kong;
Po Leung Kuk Hong Kong Taoist Association Yuen Yuen Primary School, Hong Kong

Coding education is one of the best ways to develop learners’ computational thinking skills, and significant positive correlation coefficients can be found between computational thinking education and mathematics education. However, computational thinking skills have not yet been emphasized in mathematics education. Through this systematic literature review, insights about the connection among computational thinking education, mathematics education, and coding education, and the impact of integrating computational thinking into mathematics education through coding education will be examined.

Keywords: computational thinking, mathematics, education, integrated, coding

Introduction

In the fast-changing world in the 21st century, students are required to equip different essential skills to become innovators and problem-solvers, such as the “4C’s of 21st century skills”—critical thinking, creativity, collaboration, and communication (Singh, 2021). Recently, many educators and educational psychologists found that there are strong needs to include the 5th “C” as one of the most important 21st century skills, which is, computational thinking skills. Computational thinking skills can be defined by four major components—decomposition, pattern recognition, abstraction, and algorithms, which are closely related to some other important skills in the 21st century, such as creativity, critical thinking, meta-cognition, and higher order thinking skills (Andrian & Hikmawan, 2021).

The easiest way to develop students’ computational thinking skills is to implement coding education, which allows students to experience the process of programming different digital tools with algorithms (Ho & Ang, 2015). When coding programs, algorithms, such as scheduling, route-finding, compression algorithms, and creating new algorithms, are essential for coders to complete programs. Similarly, pattern recognition skills are important for coders to simplify the programming process and the program itself. Also, coders are required to communicate clearly and logically through an algorithm with the skills of decomposition and abstraction. In short, coding education helps develop students’ computational skills efficiently and effectively (Ho & Ang, 2015). Therefore, many countries have recently revised their K-12 education’s national curriculum, to cover coding education as one of the core subjects (Vico et al., 2019).
There is no doubt that coding education should be taught through information technology (IT), information and communication technology (ICT), or computer science (CS) curriculum. However, Heil, Pearson, and Burger (2013) suggested that the concept of integrated studies mentioned in or before the 1930s, was again being emphasized in the 21st century. Educators and researchers started to re-emphasize the importance of implementing interdisciplinary studies, and design integrated curriculum, such as STEM education. With the above, Gadanidis, Brodie, Minniti, and Silver (2017) found that coding education may be integrated into mathematics curriculum, to develop students’ essential computational thinking skills effectively.

In terms of mathematics education in primary schools, the aim is to enable students to become problem-solvers who are confident in using mathematics to analyse and solve problems in real-life situations (Ramaley, 2007). In addition, the role of mathematics education is to develop students’ abstract, logical, and critical thinking skills through decomposition, pattern recognition, abstraction, and algorithm (Seidametova, 2020). Therefore, there is a relationship between mathematics education and computational thinking education.

In short, integrating computational thinking education into mathematics education through a coding curriculum seems to help students develop computational thinking skills and enhance their understanding of mathematical concepts and performance in mathematics. However, there is a lack of studies on the above research areas. This study seeks to identify the connection between computational thinking and mathematics education, and the impact of integrating computational thinking education into mathematics education through coding activities.

In this study, a systematic literature review will be conducted with four databases, which are, British Education Index, Eric, Academic Search Ultimate, and Education Research Complete, to present a summary of literature by following the seven steps of review suggested by Cooper et al. (2018).

**Literature Review**

**Computational Thinking Education**

There are hundreds of thinking skills that were named in the past, such as analytical, divergent, convergent, critical, creative, abstract, concrete, sequential, and holistic thinking skills (Moseley, Elliott, Gregson, & Higgins, 2005). In the 1950s, the term “computational thinking” was introduced, but the concepts and the ideas of computational thinking are much older (Tedre & Denning, 2016). Tedre and Denning (2016) claimed that computational thinking is also a kind of thinking skill, and shares similar ideas with engineering, systematic, scientific, and innovative thinking skills. Djambong et al. (2018) advised that there are four core components of computational thinking skills, which are decomposition, pattern recognition, abstraction, and algorithms.

For teaching the skills of decomposition to young learners, a scenario that required problem-solving techniques should be created. As mentioned by the four stages of cognitive development, learners in K-6 education are generally not able to solve complicated problems, until the learners reach the last stage of the cognitive development, formal operational stage in ages 12 or above (Piaget, 1964). Therefore, the skills of decomposition are extremely important in K-6 education. Teachers are required to facilitate students’ learning on how to solve complicated problems that involve multiple steps, such that, to teach students on how to break down the larger task into several small tasks or multi-step, which is important for them in the future when they need to carry out larger tasks.

After decomposing complicated problems into mini-tasks, the next step of problem-solving is to be an effective problem-solver, such that, to solve a problem using the simplest and easiest way in the shortest time.
The next two core components of computational thinking facilitate learners to understand how to solve problems more effectively are, pattern recognition and abstraction (Grover & Pea, 2018). The skills of pattern recognition allow students to analyze similar objects, steps, or experiences, and then identify their commonalities and patterns. By practicing these, students are able to use the patterns to solve the future problems effectively and make predictions on the questions easily, such as building a piece of IKEA furniture.

Another important skill to enhance the efficiency of solving problems in computational thinking skills is named as abstraction, also known as pattern generalization (Ho & Aug, 2015). Abstract action skill focuses on the process of identifying relevant and important information, which allows problem-solvers to filter out irrelevant or unnecessary information and steps of a problem.

Lastly, teaching algorithms as creating sequential instructions in response to the problem. Students are required to learn the order of operations as well as the skills and importance of debugging an algorithm, to represent a solution to a problem (Grover, 2018).

In short, computational thinking skills, as the skills of formulating and solving problems effectively, are an essential skill for the 21st century. There is a strong need for implementing computational thinking education in primary education.

**Computation Thinking Education**

Coding as a method of communicating with a computer through specific programming languages, is not simply about computer studies and computer programming, but it aims at allowing students to apply the skills that they have learned through coding, to different contexts and environments (Kanbul & Uzunboylu, 2017).

Nowadays, coding education is identified as one of the most effective ways to develop and enhance students’ computational thinking skills (Seo & Kim, 2016). Seo and Kim (2016) pointed out that the skills of decomposition, pattern recognition, abstraction, and algorithm are all essential in computing coding and programming activities. For example, after developing the objectives and targets, coders are required to break down the program into many small codes and programs—decomposition; then, coders have to identify the patterns of the codes and the coding experience, to simplify or duplicate the codes; also, identify and apply the useful codes, to effectively develop a set of algorithms to build a program.

Therefore, many countries started to implement computational thinking education through coding education recently (Vico et al., 2019). For example, the Curriculum Development Council of the Education Bureau of Hong Kong, has developed an official document, named “Computational Thinking—Coding Education: Supplement to Primary Curriculum” in 2020. All local primary schools are highly recommended to adopt the coding education curriculum on a school-based basis (The Education Bureau of Hong Kong, 2020).

**Integrated Studies**

Interdisciplinary studies has been promoted since the 1930s, researchers believe that integrated studies may help learners to build a better understanding of general concepts and also provide better learning opportunities and experiences for learners, in comparison with teaching in isolated subjects (Newell, Wentworth, & Sebberson, 2001). More importantly, integrated studies enhance students’ various generic skills, such as creativity, problem-solving, and critical thinking skills, which are extremely important for students in the 21st century (Newell et al., 2001).

Even though computational thinking education can be implemented through coding activities effectively, computational thinking education is not taught as an isolated subject in most of the schools and countries too
Swaid (2015) pointed out that many schools in many countries integrated computational thinking into different learning areas, such as science education, ICT education, and especially STEM education. However, there is a lack of studies about integrating computational thinking into mathematics education.

Mathematics Education

The aims of mathematics education in primary schools are not only teaching students mathematical knowledge and concepts, but more importantly, enabling students to solve problems through the process of decomposition, investigating and recognizing patterns, abstraction, and creating algorithms, which shares the same core elements of the computational thinking education (Ramaley, 2007).

Firstly, Ling and Loh (2021) observed that pattern recognition will be taught in mathematics education in various ways, such as recognizing relations of data, patterns, and trends on the changes of data from statistical charts, sequences of numbers patterns, using triangles or quadrilaterals to design patterns based on tessellations of a plane, create and explore geometric patterns by using suitable application software.

Secondly, decomposition helps to learn mathematics and solve mathematical problems easily. One of the most frequently used practices of decomposition is called “partial sums addition as an alternative addition method”, which is based on decomposition. Besides applying the concept of decomposition into solving mathematical problems in basic operations, complicated mathematical problems also require students to solve the problems through decomposition (Adomian, 1988). For example, in the question about calculating perimeter and area of a heart shape below (see Figure 1); learners are required to decompose the question into two parts by dissecting the heart shape into a square and two semicircles.

Thirdly, one of the important aims of mathematics education is to enhance students’ problem-solving skills with real-life situations. Therefore, the concept of pattern generalization and abstraction is also extremely important in mathematics education. There are no hints and guidance for a problem-solver to find the answer in real life. Problem-solver is required to identify useful information in response to the problem, and to enhance the efficiency and effectiveness of solving the problem.

Fourthly, algorithms represent procedures in mathematics education, showing the steps of solving a mathematical problem (Lockwood, DeJarnette, Asay, & Thomas, 2016). For example, in a problem of “What is 52 divided by 3” can be solved by the following algorithm:

Step 1. Question: How many times do 3 go into 5?
Step 2. Answer: The answer is 1 time.
Step 3. Question: How many are left over?
Step 4. Answer: The answer is 2 tens. Then, put 2 tens in front of the 2.
Step 5. Question: How many times do 3 go into 22?
Step 6: Answer: The answer is 7 times.
Step 7: Question: How many are left over?
Step 8. Answer: The answer is 1 unit.
Step 9. Since the value of the answer is less than the divisor, 1 will be the remainder
Step 10. This is concluded that the answer is 17 with a remainder of 1.

The example above shows the importance of algorithmic thinking skills in mathematics. When solving mathematical problems, the sequential instructions and order of operations matters a lot in solving problems.

From the above, there may exist a positive correlation between computational thinking and mathematics education, however, there is a lack of studies about integrating computational thinking and mathematics education.

Research Questions

This study will first identify the connection between computational thinking and mathematics education through qualitative research data collection. Then, a systematic literature review with the collection of quantitative research data will be conducted to examine the impact of integrating computational thinking into mathematics education. Lastly, the existing coding curriculum for teaching computational thinking will be studied.

RQ 1: Connection between computational thinking and mathematics education;
RQ 2: Impact of integrating computational thinking into mathematics education.

The Way of Retrieving Computational Thinking Studies

To conduct the literature review on the selected topic, procedures for performing systematic reviews suggested by Kitchenham (2004) has been adopted, which are planning the review, conducting the review, and reporting the review, as shown below:

Firstly, a preliminary research was conducted in October 2020, using the database of “Web of Science Core Collection” with search string (TS = (Math* AND Computational Thinking AND (Primary OR Elementary OR Junior))) AND LANGUAGE: (English) AND DOCUMENT TYPES: (Article) Indexes = SSCI, A & HCI Timespan = All years.

After a title and abstract screening, there are 33 articles are related to computational thinking in mathematics education, whereas, 27 of the articles are released in the last five years, which indicated that, the number of published CT in ME studies has been increasing significantly since 2015. Six studies are selected for the literature review, because these studies involved teaching and learning implementation, as shown in the Table 1 below.

Methodology

Research Methods

In order to study the three research questions about integrating computational thinking education into mathematics education, a systematic literature review has been conducted by following the procedures for performing systematic literature review suggested by Cooper et al. (2018). The steps are as follows:

1. Deciding who should undertake the literature search;
2. Determining the aim and purpose of a literature search;
3. Preparing for the literature search;
4. Designing the search strategy;
5. Determining the process of literature searching and deciding where to search (Bibliographic database search);
6. Determining the process of literature searching and deciding where to search (Supplementary search);
7. Managing the references;
8. Documenting the search.

**Databases**

A systematic literature review will be conducted with four databases, which are British Education Index, Eric, Academic Search Ultimate, and Education Research Complete.

**Search Strings**

The search string is set to be TI (computational thinking or algorithmic thinking or programming or coding) and (math or mathematics or maths or math education) with limiters of the Published Date: 20110101-20210630; Publication Type: Academic Journal; Document Type: Journal Article; Language: English.

**Table 1**

**Selected Articles for Literature Review**

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Title</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bråting, K. &amp; Kilhamn, C.</td>
<td>The integration of programming in Swedish school mathematics: Investigating elementary mathematics textbooks</td>
<td>2021</td>
</tr>
<tr>
<td>Daher, W.; Baya’a, N.; Jaber, O.; &amp; Awawdeh S. J.</td>
<td>A trajectory for advancing the meta-cognitive solving of mathematics-based programming problems with Scratch</td>
<td>2020</td>
</tr>
<tr>
<td>Dohn, N.</td>
<td>Students’ interest in Scratch coding in lower secondary mathematics</td>
<td>2020</td>
</tr>
<tr>
<td>Falloon, G.</td>
<td>An analysis of young students’ thinking when completing basic coding tasks using Scratch Jnr. On the iPad</td>
<td>2016</td>
</tr>
<tr>
<td>Gökçe, S.; Yenmez, A.; &amp; Özpinar, I.</td>
<td>An analysis of mathematics education students’ skills in the process of programming and their practices of integrating it into their teaching</td>
<td>2017</td>
</tr>
<tr>
<td>Israel, M. &amp; Lash, T.</td>
<td>From classroom lessons to exploratory learning progressions: Mathematics plus computational thinking</td>
<td>2020</td>
</tr>
<tr>
<td>Lambie, D.</td>
<td>Presenting practical application of mathematics by the use of programming software with easily available visual components</td>
<td>2011</td>
</tr>
<tr>
<td>Miller, J.</td>
<td>STEM education in the primary years to support mathematical thinking: using coding to identify mathematical structures and patterns</td>
<td>2019</td>
</tr>
<tr>
<td>Monga, M.</td>
<td>Problem posing and programming as a general approach to foster the learning of mathematics</td>
<td>2019</td>
</tr>
<tr>
<td>Solin, P. &amp; Roanes-Lozano, E.</td>
<td>Using computer programming as an effective complement to mathematics education: Experimenting with the standards for mathematics practice in a multidisciplinary environment for teaching and learning with technology in the 21st century</td>
<td>2020</td>
</tr>
<tr>
<td>Suters, L. &amp; Suters, H.</td>
<td>Coding for the core: Computational thinking and middle grades mathematics</td>
<td>2020</td>
</tr>
<tr>
<td>Innervik, P.</td>
<td>Implementing programming in school mathematics and technology: Teachers’ intrinsic and extrinsic challenges</td>
<td>2020</td>
</tr>
<tr>
<td>Winters, J. J.; Winters, K. E.; &amp; Kimmins, D. L.</td>
<td>The nuts and Bot's of math and coding in the lower grades</td>
<td>2020</td>
</tr>
</tbody>
</table>
Data Handling

The Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines mentioned by Selçuk (2019) have been adopted with the procedures for performing systematic literature review. Eight hundred and twenty-five academic journal articles were recorded through the four databases searching, 340 duplicated results were removed and 466 records were excluded. Hence, 19 articles were selected for full-text study. However, three full-text articles were unavailable online, and so excluded from the studies in the qualitative synthesis. Summary of the research result and PRISMA flow diagram are as shown on Figure 2 and Table 1 above.

Result and Analyses

RQ 1: Connection Between Computational Thinking and Mathematics Education

Miller (2019) claimed that the conceptual development between computational thinking and mathematics education are connected. There is a relationship between computational thinking and thinking practices of several disciplines, especially mathematics education (Rich, Spaepen, Strickland, & Moran, 2020). Therefore, English (2016) suggested that learning opportunities should be created through trans-disciplinary learning, such
as integrating science, technology, engineering, and mathematics, as known as STEM education, which is important and effective in developing students’ mathematics skills and concepts in meaningful ways (English, 2016).

L. Suters and H. Suters (2020) also claimed that computational thinking education can be easily embedded into the mathematics curriculum, instruction, and even assessment, to provide more and better learning opportunities in mathematics for students.

At the same time, Falloon (2016) believed that computational thinking skills align with the development of mathematics education, especially in enhancing students’ decomposition skills, critical thinking, and synthetic thinking skills, which are also the core components of the computational thinking skills. Sáez-López, Sevillano-García, and Vazquez-Cano (2019) advised that computational thinking education helps students to enhance their learning of mathematics concepts and logic substantially. Specifically, Israel and Lash (2020) suggested that, there are three different relationships of integrating computational thinking into mathematics education, which are:

1. No integration between the content areas of computational thinking education and mathematics education;
2. Computational thinking education and mathematics education are taught separately but under common themes simultaneously;
3. Computational thinking education and mathematics education are taught in an integrated subject, such as STEM education.

In addition, Israel and Lash (2020) believed that there is a supportive relationship between computational thinking education and mathematics education, such that one learning area supports another learning area’s curriculum or activity. However, Gökçe, Yenmez, and Özpınar (2017) have done an experimental pre-test-post-test control group design to examine the impact of implementing computational thinking education through coding education. Results showed that learning to code helps to enhance learners’ critical thinking significantly with \( t(41) = 7.01, p < 0.05 \). But, there was no significant effect on enhancing problem-solving skills, such that \( t(41) = 1.28, p > 0.05 \), which is different from most of the existing research (Gökçe et al., 2017). Therefore, there is a strong need to further study the impact of integrating computational thinking into mathematics education, which will be presented in the RQ 2 below.

**RQ 2: Impact of Integrating Computational Thinking Into Mathematics Education**

L. Suters and H. Suters (2020) emphasized that computational thinking is essential for students’ futures in the 21st century, which is shown in the USA National standards and frameworks for mathematics education. To study the impact of the computational thinking in mathematics, Solin and Roanes-Lozano (2020) studied the Common Core’s eight Mathematical Practice Standards by Common Core State Standards Initiative, to show the impact and importance of enhancing students’ mathematical concepts and knowledge in those eight mathematical practices through coding education in computational thinking education. It is found that coding education can help students to study, improve, practice, and consolidate their learning in mathematics (Solin & Roanes-Lozano, 2020).

At the same time, Israel and Lash (2020) declared that computational thinking education can be used as a tool for constructing deep and long-lasting memories of mathematics learning, as well as keeping students engage in class to enhance their decomposition, pattern recognition, abstraction, and algorithmic skills through both programming and unplugged activities.
Bråting and Kilhamn (2021) also answered the question of how programming or coding education creates a connection between computational thinking and mathematics education. Qualitative content analysis has been conducted that programming requires learners to reformulate different computational thinking skills by using mathematical concepts and knowledge (Bråting & Kilhamn, 2021). Also, programming helps students to discover and develop new mathematical concepts, specifically, pattern recognition, decomposition, and algorithmic skills (Bråting & Kilhamn, 2021). However, Bråting and Kilhamn (2021) mentioned that most of the existing programming curriculums or tasks that enhance students’ pattern recognition skills are in key stage 1, but significantly less in key stage 2. At the same time, the development of the skills of decomposition and algorithmic thinking are being emphasized in key stage 2 through programming tasks (Bråting & Kilhamn, 2021).

Daher, Baya’a, Jaber, and Awawdeh (2020) also agreed that coding has a positive impact on learning mathematical concepts, especially on the topic of symmetry and pattern recognition. Scratch is one of the most famous coding and programming platforms that can be used for integrating computational thinking education into mathematics education via coding activities (Daher et al., 2020).

In addition to the positive impacts of integrating computational thinking education into mathematics, Monga (2019) recognized that coding and programming is an effective way and as a language for learning mathematics naturally. Specifically, Daher et al. (2020) suggested that using coding, technology, and programming as tools and strategies can help to improve students’ problem-solving abilities in their mathematics activities and problems.

However, there is a lack of quantitative studies about integrating computational thinking into mathematics, one of the quantitative studies was conducted by Lewis and Shah (2012) to study the correlation between coding and mathematics performance, which represent the same result with Ke (2014), such that, positive correlation and attitudes towards integrating coding education with mathematics education are found significantly.

However, Solin and Roanes-Lozano (2020) pointed out that, most of the mathematics teachers are not aware of the significant positive connection among coding education in computational thinking education, with mathematics education. Also, J. Winters, K. Winters, and Kimmins (2020) claimed that mathematics teachers in primary schools tended to integrate coding programs and activities into lower grades. However, most of the teachers are implementing coding education with mathematics in an unsystematic manner and without a detailed planning and structure of the implementation (Winters et al., 2020). Therefore, there is a need to conduct more qualitative and quantitative research about integrating computational thinking in mathematics education, and there is a strong need for in-service teachers to learn about coding and programming, as well as learn how to teach mathematics through programming in a systematic way.

Discussion and Recommendation

As discovered, there are three relationships of computational thinking education and mathematics education integration, which are no integration, partial integration, and fully integration that implement curriculum as an integrated subject or interdisciplinary study. Much existing research supports the concept of integrated study, such as STEM education, which not only enhances students’ learning in science, technology, engineering, and mathematics, but also strengthens their ability to integrate and apply knowledge and skills that are required to be successful in the 21st century. Since it is found that, there is a significant positive correlation between computational thinking education and mathematics education, and computational thinking education is
normally conducted through science and technology lessons, this is strongly suggested, to further promote STEM education, and integrate coding activities into mathematics to enhance students’ performance in mathematics. To further promote integration of mathematics with coding activities, there are three recommendations in response to integrating computational thinking into mathematics education.

Firstly, conduct a systematic review and meta-analysis. There is a lack of studies about integrating computational thinking into primary mathematics education through coding education. Hence, a systematic review, and especially a formal quantitative analysis are needed to examine the effectiveness of teaching mathematics with coding activities.

Secondly, implementation of a mandatory continuing professional development (CPD). Many studies and reports have shown that primary teachers do not have enough content knowledge (CK) in coding education, technological knowledge (TK) in using coding activities as a tool to teach mathematics, and pedagogical content knowledge (PCK) of understanding the best practices for teaching mathematics through coding activities. Hence, each school should implement a school-based systematic teacher professional development plan for STEM education, coding education and teaching mathematics with coding activities.

Thirdly, promote integrated subject and trans-disciplinary learning systematically. Primary schools are encouraged to promote integrated study, such as STEM education. However, many schools are implementing STEM education in an unsystematic manner. There is a strong need for schools to plan for implementing STEM education in a systematic way, such as arranging regular lesson time for the STEM discovering activities.

**Limitation and Conclusion**

Existing studies have shown that there is a significant positive correlation between computational thinking and mathematics education, and integrating computational thinking into mathematics education through coding education has a positive impact on both enhancing learners’ computational thinking skills and improving their performance in mathematics. However, most of the existing research is focusing exclusively on secondary school education only (Rodríguez-Martínez, González-Calero, & Sáez-López, 2020). Also, the relationship between computational thinking and mathematics education has not yet been clarified. There are some suggestions for future research as suggested below.

Firstly, research and study about integrating computational thinking in mathematics education in primary school, especially studies with quantitative research methods with larger sample size and wider scope are in strong need.

Secondly, the relationship between computational thinking and mathematics education has to be studied, such as the correlation coefficient of the performance of coding education and mathematics education, to examine the impact of implementing integrated study with computational thinking education and mathematics education.

Thirdly, in addition to examining the impact of the students’ academic performance, this is also important to study students’ learning motivation and learning interest on learning mathematics through coding education.

Last but not least, teacher education should also be further studied, as it plays an important role and has a great impact on the implementation of integrated learning.
References


Gökçe, S., Yenmez, A. A., & Özpınar, İ. (2017). An analysis of mathematics education students’ skills in the process of programming and their practices of integrating it into their teaching. *International Education Studies, 10*(8), 60-76.


