INTERNATIONAL JOURNAL OF ACADEMIC RESEARCH IN BUSINESS & SOCIAL SCIENCES

Vol 13, Issue 9, (2023) E-ISSN: 2222-6990

Coding and Computational Thinking Learning for Vocational Students: Issues and Challenges

Nurbaya Mohd Rosli, Mohd Effendi @ Ewan Mohd Matore

Faculty of Education, National University of Malaysia, Selangor, Malaysia

Email: p121104@siswa.ukm.edu.my, effendi@ukm.edu.my

To Link this Article: http://dx.doi.org/10.6007/IJARBSS/v13-i9/17766 DOI:10.6007/IJARBSS/v13-i9/17766

Published Date: *08* September 2023

Abstract

Coding is the act of writing code, whereas programming is the act of developing functional software or computer programs. A set of instructions or a set of rules expressed in a specific programming language are referred to as "computer code" in computer programming. Students pursuing certificates in vocational studies are now starting to learn about programming. At the lower secondary level, these students received preparation in computational thinking (CT). This concept paper identifies three primary problems with coding and computational thinking among vocational students: (i) CT's inadequate educational foundation; (ii) lack of interest in programming, and (iii) absence of measurement of CT literacy level in programming for vocational students. The uniqueness of this concept paper is in its ability to elaborate on coding challenges in programming and computational thinking abilities. The results of this study imply that screening should be done before bringing on new pupils. Attention should be taken to this article so that novice learners are better equipped to continue learning in a motivated way because they get the fundamentals of coding in programming. The aim of this research is to contribute to a better understanding of the problems and difficulties in teaching computational thinking and programming skills. Vocational colleges and Lead Trainers may engage collaboratively to create this proposed solution with the aim to provide interventions for students who are still novice to programming.

Keywords: Coding, Programming, Vocational, Measurement, Computational Thinking

Introduction

The Malaysian Ministry of Education (MOE) launched the Malaysia Education Blueprint 2013-2025 as part of its attempts to improve student performance and accomplishment The new ideas and strategies used in this change are intended to facilitate the way students learn skills for the twenty-first century.

The Standard Secondary School Curriculum (KSSM), which introduces the fundamentals of programming at the lower secondary level in the Basic Computer Science (ASK) subject and is further strengthened by the application of elements of Information Technology and

Communication (TMK), is one of the MOE's initiatives in this transformation. ASK subject is presented to help students learn the fundamentals of how digital technology is built, enabling them to develop computational thinking (CT) skills and the ability to solve problems.

At the upper secondary level, transformation is still being strengthened. In addition to the KSSM, MOE has also developed a Vocational Education Transformation Plan to improve training for skilled graduates. The demand for graduates with a vocational diploma is significant and will keep rising each year. The MOE transform is the transition of Vocational Secondary Schools (SMV) into Vocational Colleges (KV). The Malaysian Vocational Diploma (DVM) is a restructuring curriculum offered by KV at the accredited certificate and diploma levels. A six-month period of industrial training is also included in this programme. Information Technology is one of the subspecialties of the subject that KV offers. It includes a variety of chosen programmes, such as Network and Communication Systems, Database Management System and Technology, and Web Applications.

Computational thinking (CT) was first introduced by Seymour Papert in the early 1980s and is better known as procedural thinking (Papert, 1980). With an understanding of procedural thinking, Papert aims to teach students how to develop algorithmic answers that are subsequently carried out by computers. According to (Wing, 2006), CT is a process that involves formulating a problem and establishing a workable solution between a computer and a human or machine. According to (Yadav et al., 2017), the definition of CT also includes abstracting problems and formulating automatable solutions in the realm of the mind. Next, (Cansu & Cansu, 2019; Fuber, 2012) defines CT as the process of recognising the surrounding computational aspects and employing computer science tools and techniques to comprehend and justify both natural and artificial systems and processes. A solution that can be implemented by a computer is what Karl Beecher (2018) recently described as CT, an approach to problem solving that uses several computer science practises and ideas. It is not limited to programmers. However, this may be used to various kinds of fields.

CT skills are essential for novice programmers to solve problems involving coding a computer system. Students' ability to develop programme code and solve problems is aided by their prior CT knowledge (Wing, 2006) Abstraction, which is the process of locating and gathering pertinent information to ascertain the essential idea and omit irrelevant information, is one of the fundamental abilities in CT.

Studies related to CT actively started researching Wing's (2006) article regarding related articles. Following that, many academics in the area of computers in education began to look into CT and the role that these abilities play in the creation of teaching pedagogy for both programming and non-programming subjects. CT has been utilised not only in the field of computer science but also in STEM (Science, Technology, Engineering, Mathematics). Studies in the areas of CT and coding include those from Totan & Korucu (2023); Lai & Ellefson (2022); Kutay & Oner (2022); Fang et al (2022) Moreira Oliveira, 2022; Li et al., (2021); de Ruiter & Bers (2021); Palts & Pedaste (2020) dan (De Jong & Jeuring, 2020). Studies from (Moreira Oliveira, 2022) dan (Palts & Pedaste, 2020) focusing on the curriculum, development of models, and how to develop and carry out activities in introducing CT at the university first-year level as well as at other levels of education. Next, Totan & Korucu (2023) dan Kutay & Oner (2022) investigate coding research employing CT for elementary school kids. A study on CT assessment and coding for elementary school children was conducted by (Fang et al., 2022; Li et al., 2021; de Ruiter & Bers, 2021; Lai & Ellefson, 2022) carried out research on the CT competency structure using psychometric assessment and measurement. An overview study was done by (De Jong & Jeuring, 2020) to see CT interventions in the context of higher education.

CT studies and computer programming are not emphasised for vocational students in Malaysia. There is only one new study by (Daud, 2023), and it focuses on creating a programming literacy course for students in the vocational field. In addition, studies by (Azhar & Adnan, 2022) and (Yusoff et al., 2021) focused on the creation of matriculation-level teaching and learning modules as well as diplomas involving CT and programming, while (Sidek, 2022) researched a CT teaching and learning model based on gamification. (Sovey et al., 2022) assessed the level of CT disposition among secondary school pupils, while (Napiah & Hashim, 2021) analysed the level of teacher preparedness for CT.

Thus, the discussion of this subject creates a space for identifying issues with CT and coding that vocational students encounter. It is unfortunate that, despite being in the vocational stream, the issues of CT and coding seem to be ignored in this group. Further research will explore into various issues with CT and coding among students in the vocational field.

Issues and Discussion

First Issue: Inadequate Basic Understanding in CT

The initial introduction was provided by MOE to students in primary schools, where teachers applied CT skills during their learning of subject areas in STEM once a week via Information Technology and Telecommunications (TMK) subjects. Students are then directly introduced to these abilities at the secondary school level in the Basic Computer Science (ASK) course. The issue involves two types of students: those who lack a foundation in computer science and have no experience writing code, and those who have encountered these concepts but have not fully grasped them. CT skills consist of four basic pillars, namely decomposition, pattern recognition, abstraction, and algorithms (Krauss & Prottsman, 2017). According to preliminary research by (Rosman & Hamid, 2020), junior secondary students struggle with the concepts of algorithms and programming code in ASK topics. Students can employ CT skill strategies to assist them address programming-related issues (Nor & Khalid, 2020). When students begin learning programming languages in the first semester and are introduced to generating pseudocode, building flowcharts, and writing source code for programming modules, this problem can be identified early on. If students already possess CT skills that describe the methods to solve problems using the fundamental CT concepts, these processes can be learnt and comprehended successfully. On the other hand, inexperienced programmers tend to write code without considering the issue at hand. The situation can be viewed from the standpoint of both the educational approach used to teach programming and the perspective of students who lack the cognitive and strategic problem-solving skills needed.

Additionally, IT vocational students need to study intensively from the start to comprehend and acquire CT abilities and be able to develop better and full programming code as a software programme. Students' chances of advancing to a diploma programme are exceptionally slim if they are weak and fail to master these skills, and they will later face a challenging route at a higher level.

Second Issue : Lack of interest in programming

When new students are short listed into the information technology programme, they go through a screening procedure that includes interviews with lecturers and industry professionals. Students who are not familiar with programming still have no idea how to create programme code. Some students discover that programming languages are challenging to learn after learning them in the lab (Siti Suhaina Mohamed Daud, 2023), at which point they give up and conclude that it is challenging and complicated (Ummi et al., 2020). Students lose interest in programming as a result of this predicament and the challenging learning they had to endure. There is a need for learning modules to be established as a reference for lecturers, as well as for lecturers to investigate methodologies and modules that can be employed during the learning process (Yusoff et al., 2021). To assist students in producing programming code, it is necessary to investigate CT skill techniques (Nor & Khalid, 2020). In addition, according to (Yağci, 2016)the traditional learning method and a dearth of practise exercises cause students to struggle with the programming concept. This problem is frequently discovered at the start of the semester when programming-related courses are offered.

Third Issue : Absence measurement of CT literacy in programming.

The measurement of CT programming literacy has not yet received much attention. CT might be considered a brand-new aspect of computer literacy that ought to be taught in schools (Tsai et al., 2021; Wing, 2011). There is a study measuring the level of CT disposition among high school students (Sovey et al., 2022). and a psychometric assessment to investigate the structure of PK competence (Lai & Ellefson, 2022). To screen students for admission to the information technology area in the vocational stream, no special study has been carried out to determine the level of CT literacy in coding and programming. De Ruiter & Bers (2021); Yusoff (2021) both state that the majority of studies are more focused on creating CT assessments and programming for kids as well as on creating teaching modules for programming subjects at the matriculation level.

Since no attempt has been made to ascertain the students' interest in and proficiency with the fundamental CT programming skills, the study of the development of instruments to measure the level of CT literacy in programming has not yet gained attention in Malaysia. Due to the lack of interest and increased regret if students fail their studies, this gap needs to be addressed in order to ensure that students' interest is driven by the chosen flow path and to reduce learning halfway.

It is possible to determine whether students are interested in coding and programming by using an instrument to assess their CT literacy in that subject. Hence, the results can be used to screen applicants for admission at the certificate level in the vocational stream and serve as a benchmark for lecturers as they develop more in-depth learning materials and interventions to spark students' interest in learning programming and aid them to develop their programming skills.

Suggestions for Improvement on the Issues

There are several suggestions for improvement for CT, coding, and programming issues for vocational students.

The first issue (inadequate basic understanding in CT) might be improved by giving students a clearer and more engaging explanation to spark their interest in learning CT ideas and skills, followed by writing computer code. Additionally, sufficient training must be provided to teachers who teach lower secondary students on the fundamentals of CT so they can effectively express this idea. Students will lose interest in coding in their potential professional careers if this subject is not taught successfully(Thorat & Kshirsagar, 2021). When studying CT skills, students must have an established basis in problem-solving skills before they can create basic programme code. As a result, these students can concentrate on the new programming course by using the previously acquired CT abilities. On the other hand, students without CT foundational abilities can still learn to write programme code, but it takes time to compile and finish the programme code sequence. It is appropriate for college lecturers to provide a comprehensive session to introduce CT skills and then adapt them to coding and programming for the group of students who lack those skills.

Suggested improvements to the second issue (lack of interest in programming) are to foster students' interest in produce code and programming. Tasks that are simple and basic in their basis must be presented to students. They must be confident in their ability to write programming code and have an optimistic mindset. If students struggle with compiling the programme code, lecturers must also assist and encourage them. Additionally, group assignments can give each group member the chance to offer suggestions for coding problems that need to be solved, which makes students more eager and open to seeing the solution in coding. The methodology of pair programming, which involves two programmers working at one workstation in software development, must be taught in class. If students use such a method to write their own code, it can help them feel less anxious since pair programming makes the code easier to read (Kapoor et al., 2023).

Basically, lecturers are crucial in encouraging students' interest in programming. To present the content of CT skills and programming, lecturers must be prepared with the necessary knowledge, abilities, and effective teaching techniques. The all-encompassing toolkit that lecturers have may undoubtedly influence the way they teach, and the ease with which topics are understood by students will further spur interest in writing programming code.

The third issue (absence measurement of CT literacy in programming) could be improved by gauging students' interest in programming that uses the right measurement theory and their level of CT literacy. A study or solution (Gouws et al., 2013), through interest screening or intervention, can be developed to deal with students' difficulties in learning the fundamentals of coding and programming at the basic level considering their inadequacies in basic CT abilities.

A pathway for student interest and early intervention by lecturers after accepting new students requires the development of a measurement of the level of literacy and interest of vocational students, particularly in Malaysia. This measurement should be done in addition to the evaluation and development of models made by other researchers. Additionally, testing of measurement items is required to establish validity and reliability and to demonstrate the high calibre of the generated items.

CT and Programming Learning Challenges for Vocational Students : Expectations for Future Momentum

Issues in coding, programming, and CT skills continues (Melro et al., 2023; Daud, 2023) and will be studied and analysed for improvement, either through curriculum revision or learning delivery strategies in lessons and labs. A thorough intervention must be put in place over the next ten years to guarantee that vocational students have the fundamental CT problem-solving skills they need to use coding and programming to build a programme. Therefore, students with middling ability will lag and have a tough time college if these concerns are not considered and handled. In addition to the industrial training that students will receive during their last semester of study, the college may also host a programme, brief workshop, or competition involving the industry that presents job chances concurrently. Students will be exposed to the real world of work and might add motivation and inspiration to succeed further in this field with industry interaction. In addition, students need to be highly resilient when writing programme code as they study computers. Students must make sure that their understanding of computer technology is current with the changes that take place every day and every week by staying up to date with the newest advancements in coding and information technology. This is a good added value if the student is highly enthusiastic and able to explore independently.

Students in grades K–12 are exposed to CT skills on a global scale. Most of them concentrate on teaching primary students and first-time programmers the fundamentals of programming using the Scratch programming language. Numerous research as been conducted to improve the way that coding skills are taught at the beginning level in higher education institutions. However, it has not yet received specific attention in the vocational field. The study (Van der Linde-Koomen et al., 2023) is directed at pre-vocational students in the Netherlands who focus on learning in an unplugged mode involving CT skills with the setting of a bakery course. The study's findings demonstrate the applicability and interest of the idea of CT skills. As a result, students should be able to use CT skills as a foundation for problem-solving. In the context of coding and programming, it is crucial to direct students through the steps of developing CT skills, writing pseudocode, creating flowcharts, and then writing programming code for a construction software.

Why not strengthen CT skills if it can kickstart coding learning? If this issue is ignored, it may partially skip a process of student understanding in learning programming (which is synonymous with problem solving). Students' interest in generating good programming code will undoubtedly increase if they acquire CT skills since they will be able to organise issues based on previously learnt concepts. This gives lecturers the opportunity to focus specifically on teaching students new programming languages and improving their programming skills.

Conclusion

This concept paper addresses issues as can be seen in Figure 1. There are also some proposed solutions to the problems highlighted. In particular, the management of MOE and Technical and Vocational Education Division should offer a support structure in terms of finances and training. The support provided can raise students' technical proficiency, help them all achieve passing grades, and almost certainly increase their employability after graduation. By giving incoming students in the vocational stream an intervention module for learning fundamental CT skills, MOE can plan deeper research. Also, more efforts should be made to expose and highlight CT abilities that can aid learning in computer science, STEM, and even non-technology subjects.

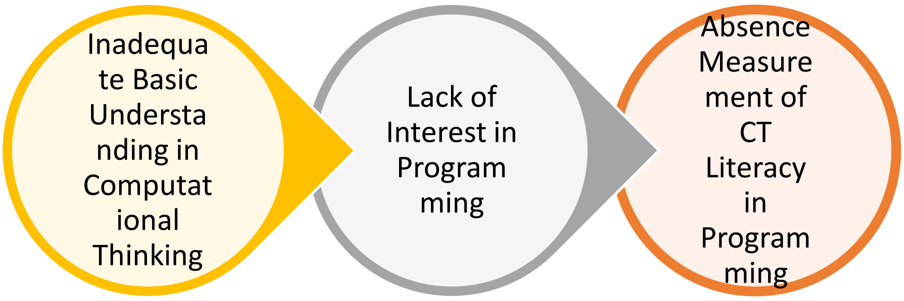
Inadequate Basic Understanding in Computational Thinking
Lack of Interest in Programming
Absence Measurement of CT Literacy in Programming

Figure 1 : Issues in Coding and Computational Thinking Learning for Vocational Students

Corresponding Author
Mohd Effendi @ Ewan Mohd Matore (Ph.D)
Faculty of Education, National University of Malaysia, 43600 UKM Bangi, Selangor, Malaysia.
Email: effendi@ukm.edu.my

Acknowledgment
I am profoundly grateful to my supervisor, Associate Professor Ts. Dr. Mohd Effendi @ Ewan Mohd Matore for his unwavering support and guidance throughout the course of writing. I also extend my deepest gratitude to my family, whose served as a constant source of strength and inspiration during writing this review paper.

References

Azhar, N., & Adnan, N. H. (2022). Mengkaji Kelemahan dan Kekuatan Dalam PdP Pengaturcaraan C #: Satu Kajian Kes. *Jurnal Dunia Pendidikan*, *4*(2), 280–293. https://doi.org/10.55057/jdpd.2022.4.2.25

Cansu, F. K., & Cansu, S. K. (2019). An Overview of Computational Thinking. *International Journal of Computer Science Education in Schools*, *3*(1), 17–30. https://doi.org/10.21585/ijcses.v3i1.53

De Jong, I., & Jeuring, J. (2020). Computational Thinking Interventions in Higher Education: A Scoping Literature Review of Interventions Used to Teach Computational Thinking. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3428029.3428055

de Ruiter, L. E., & Bers, M. U. (2021). The Coding Stages Assessment: development and validation of an instrument for assessing young children's proficiency in the ScratchJr programming language. *Computer Science Education*, *00*(00), 1–30. https://doi.org/10.1080/08993408.2021.1956216

Fang, J. W., Shao, D., Hwang, G. J., & Chang, S. C. (2022). From Critique to Computational Thinking: A Peer-Assessment-Supported Problem Identification, Flow Definition, Coding, and Testing Approach for Computer Programming Instruction. *Journal of Educational Computing Research*, *60*(5), 1301–1324. https://doi.org/10.1177/07356331211060470

Gouws, L., Bradshaw, K., & Wentworth, P. (2013). First year student performance in a test for computational thinking. *ACM International Conference Proceeding Series*, 271–277. https://doi.org/10.1145/2513456.2513484

Kapoor, A., Xie, T., Kwan, L., & Gardner-Mccune, C. (2023). *Logistics, Affordances, and Evaluation of Build Programming*. 179–185. https://doi.org/10.1145/3545945.3569756

Krauss, J., & Prottsman, K. (2017). *Computational Thinking and Coding for Every Student*. Corwin.

Kutay, E., & Oner, D. (2022). Coding with Minecraft: The Development of Middle School Students' Computational Thinking. *ACM Transactions on Computing Education*, *22*(2), 1–19. https://doi.org/10.1145/3471573

Lai, R. P. Y., & Ellefson, M. R. (2022). How Multidimensional is Computational Thinking Competency? A Bi-Factor Model of the Computational Thinking Challenge. *Journal of Educational Computing Research*, *0*(0), 1–24. https://doi.org/10.1177/07356331221121052

Li, Y., Xu, S., & Liu, J. (2021). Development and Validation of Computational Thinking Assessment of Chinese Elementary School Students. *Journal of Pacific Rim Psychology*, *15*, 1–22. https://doi.org/10.1177/18344909211010240

Melro, A., Tarling, G., Fujita, T., & Kleine Staarman, J. (2023). What Else Can Be Learned When Coding? A Configurative Literature Review of Learning Opportunities Through Computational Thinking. *Journal of Educational Computing Research*, *February*. https://doi.org/10.1177/07356331221133822

Oliveira, M. C. (2022). The Development of Computational Thinking in Computing Higher Education. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V. 2 (SIGCSE 2022), March 2022, Providence, RI, USA* (Vol. 1, Issue 1). Association for Computing Machinery. https://doi.org/10.1145/3478432.3499217

Napiah, A. S. M., & Hashim, M. (2021). Tahap Kesediaan Guru Pelatih terhadap Pelaksanaan Pemikiran Komputasional. *Journal of ICT in Education*, *8*(4), 81–103.

Nor, S. Z., & Khalid, F. (2020). Penggunaan Modul Berasaskan Pemikiran Komputasional dalam Meningkatkan Kemahiran Menyelesaikan Masalah Pengaturcaraan dalam Kalangan Pelajar Asas Sains Komputer. *Proceeding of the International Conference on Global Education IX*, *Kpm 2017*, 546–553.

Palts, T., & Pedaste, M. (2020). A model for developing computational thinking skills. *Informatics in Education*, *19*(1), 113–128. https://doi.org/10.15388/INFEDU.2020.06

Rosman, N., & Hamid, J. (2020). Penguasaan Konsep Pengaturcaraan Melalui Pembelajaran Teradun: Satu Kajian Kes. *Journal of ICT in Education*, *7*(1), 52–57. https://doi.org/10.37134/jictie.vol7.1.6.2020

Sidek, S. F. (2022). The Design and Validation of A Tool to Measure Content Validity of A Computational thinking Game-Based Learning Module for Tertiary Educational Students. *International Multidisciplinary Research Journal*, *4*(1), 1–9. https://doi.org/10.54476/iimrj01

Daud, S. S. M R. M. Z. (2023). Penilaian Pembangunan Modul Celik Programming Untuk Pelajar Kolej Vokasional Melalui Aspek Kesahan. *Sains Humanika*, *15*(1), 57–65. https://doi.org/10.11113/sh.v15n1.1993

Sovey, S., Osman, K., & Matore, M. E. E. M. (2022). Rasch Analysis for Disposition Levels of Computational Thinking Instrument Among Secondary School Students. *Eurasia Journal of Mathematics, Science and Technology Education*, *18*(3), 2–15. https://doi.org/10.29333/ejmste/11794

Thorat, S. A., & Kshirsagar, D. P. (2021). Developing logic building, problem solving, and debugging programming skills among students. *Journal of Engineering Education Transformations*, *34*(Special Issue), 402–406. https://doi.org/10.16920/jeet/2021/v34i0/157188

Totan, H. N., & Korucu, A. T. (2023). The Effect of Block Based Coding Education on the Students' Attitudes about the Secondary School Students' Computational Learning Skills and Coding Learning: Blocky Sample. *Participatory Educational Research*, *10*(1), 443–461. https://doi.org/10.17275/per.23.24.10.1

Ummi, H., Mohd, A., Fatimah, S., & Mohd, Y. (2020). Pengintegrasian pemikiran komputasional dalam aktiviti pengaturcaraan dan robotik. *International Journal of Education and Pedagogy*, *2*(2), 124–133. http://myjms.moe.gov.my/index.php/ijeapJournal

Van der Linde-Koomen, D., Jonker, H., & Voogt, J. (2023). Computational Thinking in Pre-vocational Education: A Focus on Coding Unplugged. *Teaching Coding in K-12 Schools*, 331–341. https://doi.org/10.1007/978-3-031-21970-2_22

Wing, J. M. (2006). Les enjeux de la clarification des règles. *Concurrences*, *2018*(3), 22–24.

Yadav, A., Good, J., Voogt, J., & Fisser, P. (2017). Computational thinking as an emerging competence domain. *Technical and Vocational Education and Training*, *23*(January), 1051–1067. https://doi.org/10.1007/978-3-319-41713-4_49

Yagci, M. (2016). Blended learning experience in a programming language course and the effect of the thinking styles of the students on success and motivation. *Turkish Online Journal of Educational Technology*, *15*(4), 32–45.

Yusoff, K. M. (2021). *MODEL PENYELESAIAN MASALAH DALAM PENGATURCARAAN BERASASKAN PEMIKIRAN KOMPUTASIONAL (C-Think)*. Universiti Kebangsaan Malaysia.

Yusoff, K. M., Ashaari, N. S., Wook, T. S. M. T., & Ali, N. M. (2021). Validation of the Components and Elements of Computational Thinking for Teaching and Learning Programming using the Fuzzy Delphi Method. *International Journal of Advanced Computer Science and Applications*, *12*(1). https://doi.org/10.14569/IJACSA.2021.0120111