Comparing Computational Thinking Competencies Across Undergraduate Engineering Majors: A Qualitative Analysis

Miss Na Zhao, Nanyang Technological University

Zhao Na is an undergraduate student in the Bachelor of Accounting program at Nanyang Technological University (NTU) in Singapore. She is involved in the Undergraduate Research on Campus (URECA) program and is working on computational thinking projects as part of Dr. Yeter's Research Team at NTU.

Dr. Ibrahim H. Yeter, Nanyang Technological University

Ibrahim H. Yeter, Ph.D., is an Assistant Professor at the National Institute of Education (NIE) at Nanyang Technological University (NTU) in Singapore. He is an affiliated faculty member of the NTU Centre for Research and Development in Learning (CRADLE) and the NTU Institute for Science and Technology for Humanity (NISTH). Additionally, he is the Director of the World MOON Project, the Associate Editor of the IEEE Transactions on Education, and the upcoming Program Chair-Elect of the PCEE Division at ASEE. His current research interests include STEM+C education, specifically artificial intelligence literacy, computational thinking, and engineering.

Dr. Cristina Diordieva, Nanyang Technological University

Cristina Diordieva is currently the Project Coordinator for the World MOON Project. She was a Post-doctoral Research Fellow in the joint medical school (LKCMedicine) at Imperial College London in the UK and Nanyang Technological University in Singapore. She is a co-author of a report published by the World Health Organization (WHO) in Switzerland. Her research interests include educational technology, online learning, digital health, and language massive open online courses (L-MOOCs).

Comparing Computational Thinking Competencies Across Undergraduate Engineering Majors: A Qualitative Analysis

Abstract

In this paper, we seek to investigate the ways and circumstances in which undergraduate engineering students engage in computational thinking (CT). As technology advances, we move towards a new industrial landscape where engineers face increasingly complex problems. For engineering students of the 21st century to thrive in their future careers, it is crucial for interdisciplinary education to equip them with the necessary tools and support required to solve problems effectively and think more comprehensively. CT and the engineering way of thinking enhance conceptualization and critical thinking skills, and their processes complement each other. Thus, promoting CT in engineering education is essential. However, research on the interpretation and development of CT is conducted to a limited extent at the undergraduate level. A semi-structured interview protocol was developed to gather insights on the five main pillars of computational thinking practices (i.e., abstraction, algorithms, problem decomposition, pattern recognition, troubleshooting/debugging) from eight undergraduate students (four female and four male). The participants were evenly distributed across four disciplines, including computer science, electrical and electronic engineering, civil engineering, and computer engineering, in a Southeast Asian research-focused institution. The students discussed their perceptions of CT within their respective disciplines and provided examples of when they thought CT might be helpful. Using thematic analysis, the results are used to understand how CT competencies organically arise in students' problem-solving processes. Student responses suggested that incorporating more use cases in the structuring of coursework may facilitate the integration of CT into the engineering curriculum by improving the recognition of CT concepts.

Background

In today's technology-driven world, computers are integral in expanding our capabilities across various sectors. Computing technologies are transforming sectors, and in the new industrial landscape, solving complex engineering problems calls for the use of computer systems as well as cross-functional teams [1]. Given that computer-based solutions are becoming increasingly integral to the engineering problem-solving and design process, computational thinking (CT) should be a fundamental skill for engineering students so that they can effectively leverage these tools.

Engineers are also presented with challenges that demand a broader scope of considerations due to digitalization in the field [2]. Thus, engineering has taken on a new dimension that requires a multifaceted approach to the analysis of problems and a sensitivity to the interaction between people and technical infrastructure. As a universally applicable problem-solving approach, CT can help engineering students navigate through complex situations. The practice of CT competencies (i.e., abstraction, algorithms, problem decomposition, pattern recognition, troubleshooting/debugging) is advantageous for developing analytical ability [3]. Additionally, CT is comprehensive rather than restrictive, as it builds upon computing processes [4]. As such, CT skills are essential for developing an agile and adaptable mind in the 21st century, where technology is ubiquitous. The importance of CT is reflected in the growing interest in exploring its potential role in various fields, including engineering. While CT in engineering education has been discussed in previous research, there needs to be more understanding of how CT may differ in the context of different engineering disciplines. Rich qualitative research on how students engage in CT and engineering can show how they can support each other [5]. Research has been conducted to investigate the implementation of CT in middle school education internationally. The studies emphasize the importance of CT in interdisciplinary education to foster students' critical thinking and problem-solving abilities [6]. However, there needs to be more such studies done at the undergraduate level. As there is a need for an operational scope when it comes to defining CT, a better understanding of students' perspectives and behaviors will help to set the context for how CT may add to their abilities [7]. Thus, in this study, CT is considered more of a natural reasoning process than a taught framework. The students' responses are inspired by their own ideas and instinctive approach to CT. By contextualizing their answers in terms of five aspects — abstraction, algorithms, problem decomposition, pattern recognition, and troubleshooting/debugging, the study aims to reveal further use cases and specific examples of how and when engineering students may practice computational thinking. The expected outcome is to provide a more precise direction regarding enhancing different engineering majors' computational thinking abilities.

Literature Review

In recent years, the growing relevance of computational skills has created a demand for higher competency in CT. The term "computational thinking" gained attention beyond the computer science community after Wing's 2006 article, which outlined how CT benefits everyone, not just computer scientists. CT shares many common practices with other STEM fields, making it widely applicable in integrated STEM. CT is widely described as the thought processes involved in defining problems and coming up with solutions such that the solutions take on a form that can be interpreted and performed by an information-processing agent [8]. However, there is still no unanimous and concrete definition of CT. Despite this, some consensus on aspects of CT can still be perceived in existing research. Studies that provide definitions of CT commonly reference Wing's descriptions [7].

Additionally, these definitions tend to be centered around cognitive abilities [9]. Abstraction, algorithms, problem decomposition, generalization/pattern recognition, and troubleshooting/debugging as five core competencies of CT were proposed in a framework for K-6 education by Angeli et al. [10]. Abstraction involves recognizing critical information and filtering out irrelevant details. Algorithmic thinking requires developing a logical sequence of steps to solve a problem. Problem decomposition is breaking down a problem into smaller and more manageable segments. Pattern recognition involves finding generalizable features across different problems and the potential for cross-applicability. Lastly, debugging is testing for errors and correcting them. While abstraction and decomposition are terms consistently appearing across studies, the other components used to define CT may vary [7]. However, the central idea maintained is that CT equips individuals with a framework for solving problems in a structured and systematic way, regardless of the domain in which the problem arises. CT is closely related to engineering principles in planning and assessing complicated systems meant to function within real-world limitations [11]. Thus, CT is embedded in engineering design, making it a critical skill for engineers in their education journey and professional practice.

There are frequent discussions on effective CT development methods, model-eliciting, and simulation activities [12]. It is also common for robotics and programming to be used as a means for the teaching of CT skills. Hands-on approaches appear consistently across CT research because they evoke experiences that make the application of CT observable, thus making the concept more tangible [13]. Studies have also highlighted the importance of incorporating CT into formal and informal learning settings to enhance students' CT skills and engagement [14]. While engineering has always focused heavily on technical education, according to cognitive theory, experience based on soft skills can build a foundation for better understanding and instincts in the field and its practice [12].
Research Questions:

(1) How do undergraduate engineering students perceive computational thinking practices?
(2) How do undergraduate engineering students infuse computational thinking into their professional fields?
(3) How does computational thinking infusion differ among male and female engineering students?

Methodology

To address the research question, a qualitative approach was taken in this study to collate various use cases and interpretations of CT's applicability within the frame of reference of engineering students studying in their respective disciplines.

Procedure

A semi-structured interview protocol was designed to encourage students to give examples of CT experiences and their opinions on situations when CT may be helpful. For instance, questions on the background (e.g., what made you choose your major?), CT (e.g., what do you know about computational thinking?), and gender (e.g., would you say there is a gender differentiation in your field? Probe to "how?"). Before the interview, participants were informed about confidentiality measures and signed a consent form that the university's Institutional Review Board reviewed. Each interview lasted approximately 30 minutes and was audio-recorded. The audio recordings were used to identify critical parts of the conversation, which were then converted into clean-read transcripts for analysis. The only personal details noted are their year of study, major and gender identity. In this paper, all participants will be referred to by pseudonyms.

Participants

Eight undergraduate students (four female, four male) evenly distributed across computer science (CS), electrical and electronic engineering (EEE), civil engineering (CE), and computer engineering (CE) were interviewed. All participants are part of a research-focused institution in the Southeast Asia region. Participants have prior exposure to the concept of CT from a core module, which was an inclusion criterion for participation in the study. This module includes an introductory lecture that discusses CT in terms of four aspects (i.e., abstraction, algorithms, problem decomposition, and pattern recognition) and its application in computational biology.

Participants are either in their first or second year of study and are still taking foundational courses. Computer science, while not generally regarded as an engineering discipline, was included in the study as a control because computer science majors are assumed to have greater familiarity with CT practice. Data was collected over the second semester of 2022–2023 through convenience sampling.

Data Analysis

The data collected were analyzed using a qualitative inductive approach [15]. All interviews were transcribed using manual coding. An inductive approach was used to code the data from interviews and helped in identifying patterns, themes, and categories within the data as they naturally occur. This open approach is well-suited, given the objective of this study to explore new insights into the use of CT from students' perspectives. Descriptive codes were created to unveil themes in participants' responses, which were then developed to evaluate their reported CT experiences further.

Multiple rounds of line coding were performed to analyze and extract the essence of the data [16]. The author divided the codes into semantically associated groups based on spotted trends in the codes. An iterative technique was used to group codes into prospective themes with the goal of improving categories and switching from specific to more abstract categories [16]. Overall, two major themes evolved during the analysis: (1) CT's extensive applicability in STEM fields and (2) the impact of male dominance in the engineering education environment. Findings regarding each of the emerged themes are summarized in the following section.

Findings

CT's extensive applicability in STEM fields

Table 1 below shows the participants' responses to the CT definition and how they perceive it. When faced with the concept of CT, three participants were generally able to recall four competencies (i.e., abstraction, algorithms, problem decomposition, and pattern recognition) as taught in the core module. As it is a first-year module, the two participants from the first year of study remembered more of it. Half of the participants associated the definition of CT with coding and thinking like a computer. This association could be because all first-year engineering students also had to take an introductory module about CT, which mainly covered Python and C.

Additionally, combined with students' strong association of CT with computer science, this gave rise to inconsistency in their descriptions of CT's applicability: "*For example, there's this question right, and if you manage to solve it and if you see another similar question, you actually can use the same concept to apply and solve the code*." (Selena, Computer Science). Opinion was skewed towards it being universally applicable regarding daily life but limited in the contexts of other fields (i.e., non-STEM disciplines): *"I'm sure computational thinking applies to everybody in their daily life, so not just for Comp science students."* (Ivory, Computer Science). Although participants generally viewed their majors as demanding and requiring adaptability and critical thinking for reasons such as complexity and versatility, they feel they cope by picking up

problem-solving skills independently rather than through their classes: "... *you need to go and practice yourself to find out how to actually solve the code.*" (Selena, Computer Science)
When asked to provide specific instances where they think they have demonstrated CT, the scenarios mainly were about studying, timetabling, correcting mistakes when doing practice papers, recognizing exam questions, and following the steps learned to solve common question types. However, there were other unique examples, including working section-by-section while drawing a picture, categorizing people by behavioral patterns, finding the football team's weaknesses by reviewing past matches, making inferences for history essays, and planning a time-efficient travel itinerary. Interestingly, two participants deemed CT relevant to philosophy.

Table 1: Examples of CT Competencies from the participants' responses

Category	**Examples of CT Competencies**
Decomposition	"*The first thing is to analyze the problem, break the problem into different parts, and then you need to think about how to solve it*." Cheng Yang, Male, CE, Y2 *"You need to decompose the question into easier parts for you to solve it. I think when a person applies computational thinking, it can be to any question actually, it doesn't have to be like coding. So if it is a math question, a pattern math question, all you need to do is just find the pattern, then you can solve it, yeah."* Selena, Female, CS, Y2
Pattern Recognition	*"The first time I see a question on, I don't know, like differentiation context for example. Then you're like, what? Then after a while, you just get used to it. Then you're like oh I now know how to do, because I've gained exposure to it.*" Celine, CE, Y1 "*... But there is a pattern in some people's behavior. Then you can kind of generalize, and kind of categorize them*." Sean, Male, CS, Y2
Algorithm	"*Famous algorithm called the Merge sort. It's like there's a really complex problem, but you can divide it into different small parts, and divide the small parts into smaller, smaller parts and solve the smallest and do a recursive call to solve the whole problem.*" Cheng Yang, CE, Y2 "*... must be systematic like the steps we take are quite systematic when solving questions.*" Ivory, Female, CEE, Y2
Abstraction	"*Abstraction, where you just take the key points from a problem after you decompose it or something.*" Yi Xuan, Female, EEE, Y1 "*For most of the classes, they don't teach us a specific language or specific skill to solve the problem. They just teach you the idea of this technology.*" Cheng Yang, Male, CE, Y2

Impact of male dominance in the engineering education environment

While most participants noted no difference in how students of different genders approached computational problems, the male participant from computer science observed a glaring ability gap between outstanding female students and those who could have been better in ability, stating that he rarely found someone in between. The following statement suggests a perceived gender difference in empathy and decision-making styles.

> *"I think that girls use more empathy. That's right. As in, they're more towards the feeling side. So when they try to like, when making decisions or like something, they tend to put themselves into other people's shoes and stuff. But I think when I see most of the time, a lot of guys are quite systematic."* (Elijah, boy, Electrical and Electronic Engineering)

In particular, the computer science and computer engineering participants felt strongly that they had a high male-to-female ratio. As a result of the male-dominated environment, some effects on female students that were mentioned included feelings of inferiority, feeling restricted by their own ability, and reluctance to attend tutorials or participate in competitions. The following statement highlights the importance of gender balance in the classroom for female students.

> "*Yeah, because I'm like the only female right. I sometimes find it awkward because it's all guys, and I prefer that if there's, like, more girls in my class then. I would be more willing to attend the lessons because right now I usually skip all my tutorials*." (Selena, girl, Computer Science, Y2)

Further, one of the students testifies how the predominance of males in engineering disciplines affects the self-assurance and engagement of female students. Furthermore, the student claims that female students can feel constrained by their abilities and intimidated by the prevalence of males in the engineering field. This stresses the importance of creating supportive environments that encourage female students to undertake risks and get over restricting and hesitant attitudes.

> *"Sometimes you feel very scared because you feel like, oh, you might not be able to do well and you feel very restricted by your ability. When they see it as male dominated and they're like, OK, maybe I don't want to do this because I don't think I can."* (Yi Xuan, girl, Electrical and Electronic Engineering, Y1)

Overall, the gender differences in empathy and decision-making styles, the desire for greater gender balance in classes, and the impact of male dominance in the engineering field were mostly discussed by the participants. Also, the data suggests that there are significant challenges that female students face in the academic engineering field.

Limitations

While the study aims to uncover insights into the use of CT from students' perspectives, it is essential to acknowledge that biases of the participants cannot be removed from their answers and that more generalizability is required due to the small sample size. Thus, implications of the results and the conclusions made based on the results may need to be recontextualized if they are to be applied in different settings. Additionally, there may be slight inconsistency due to the flexibility of semi-structured interviews, making it challenging to test the reliability of findings. On top of data collection limitations, subjectivity may also be present in the interpretation of data. Unconscious biases could also affect how data is analyzed and reported. Measures will be taken to reduce these limitations as much as possible, including peer review, reflection on possible biases to eliminate them, and data cleaning to ensure the evaluation of interview responses is meaningful.

Implications and Discussion

The first research question explores the perceptions of undergraduate engineering students on CT. The consensus on its use was positive, and most were able to recognize its role in problem-solving. However, it was reflected that most participants needed help articulating their problem-solving processes clearly. In the words of a participant, Elijah, being aware of CT helps because it gives "*a sense of security*." He also said that knowing the steps provides clarity and direction, eliminating the need for thinking on the spot, thus reasoning that CT education would benefit those less natural.

The second research question explores the infusion of CT in engineering. Besides the general competencies that align, CT in engineering must also account for the discipline-specific contexts. Engaging students in real-world applications is a practical approach to promoting students' interest in engineering. Engineering students retain CT skills better in model-eliciting activities than lecture-based methods [17]. CT is naturally embedded within plugged (e.g., 3D modeling, simulation software, programming [18]) and unplugged (e.g., puppy playground [14], origami) activities and can be seen in interdisciplinary subjects, including STEM and non-STEM (e.g., art, history, music). However, students do not seem to recognize the universal applicability of CT this way. The unique illustrations provided by participants in this study can be leveraged as other opportunities to enhance CT skills.

The third research question explores the difference between male and female engineering students' approaches to CT. Visibility plays a part in lowering female students' confidence in male-dominated engineering majors. When asked about her confidence in her major, Selena responded negatively because "the guys are way smarter than girls." She explained that when they returned the results, she would see that the male students "always score[d] better" than her. Due to the high mass of male students, getting better results could have made gender identity stand out as a differentiating factor to Selena, though it may not be. Selena suggested that practice was the only way to learn CT effectively and that projects can help because people learn from others who think differently. Another study integrating collaboration within CT learning settings can assist female students in learning to apply CT practices [19]. Thus, this could be something to work towards, given that most participants stated that currently, they hardly have any projects in their core modules apart from the coding ones.

Moreover, one of the students believes that females are more prone to apply empathy in their decision-making process while males make more logical decisions. While this idea may be based on stereotypes and may not apply to all students, it is interesting to consider how these decisions may affect the dynamics in the classroom. Another student complained that she felt uncomfortable being the only female in the class and claimed she would be prone to attend lessons if there were other girls. This clearly shows that there may be issues with inclusivity in engineering fields and that more efforts need to be made to ensure that female students feel acknowledged and valued. Overall, the data indicate that female students still face major obstacles in the engineering field. It is important for teachers to be aware of these issues and make efforts to develop welcoming learning environments that promote diversity, confidence and eliminate gender stereotypes. Further research may assist all students in reaching their full abilities and excelling in their academic endeavors.

Conclusion and Recommendations

There is a need for CT to be explicitly integrated into the curriculum so that engineering students can consciously draw on CT competencies to increase their propensity for effective problem-solving. However, more than simple discussions of CT concepts are required to make students recognize CT as a trainable skill or realize the universal applicability of CT when it is implicit in their technical classes. This supports adding a dimension to CT unrelated to technology or coding in the classroom [20]. The fact that second-year students recall less of the first-year CT module is a reminder that reinforcing CT skills is a crucial aspect. Distributed and generative practice, rather than blocked practice, can be considered to improve CT [21].

Additionally, it has also been pointed out by some interview participants that they felt that their teachers were occupied with finishing course content. CT integration should thus facilitate the delivery of course content rather than be merely adjacent to it. To promote the integration of CT across various disciplines, it is crucial to provide support in teacher education for CT instruction [22]. Computational modeling and simulation processes, which help teachers teach with technology in a "permanent, constructive, and tool-independent" manner, can be more effectively leveraged to emphasize CT skills [23]. Achieving a meaningful infusion of CT for different engineering disciplines should allow students to interact with technology proactively instead of merely being passive users. This prepares them for future engineering careers where it is necessary to adapt quickly to innovation.

The results of this study act as a stepping stone for gaining insights into how engineering students develop their understanding of CT by determining the most evident CT applications. The use cases presented in this study and the insights gained from student responses can guide the structuring of coursework. However, it is essential to note that implications may be limited by specific educational contexts and the student's experiences and backgrounds, as highlighted in previous studies, emphasizing the need for engineering education to be more inclusive and fairer towards students from diverse backgrounds [24]. Further design-based research is needed to form ability measures and assessments, and supplementary quantitative research may provide more robust and generalizable evidence. In conclusion, the findings of this study set a direction for the investigation of how engineering students view and apply CT, with the desired outcome of maximizing the synergy between engineering and CT in engineering education.

Acknowledgment

This material is based upon work supported by the Undergraduate Research Experience on Campus (URECA) at Nanyang Technological University in Singapore. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the URECA program. We would like to acknowledge all the researchers, data collectors, and students who participated in the study.

References

[1] L. D. Xu, E. L. Xu and L. Li, "Industry 4.0: state of the art and future trends," *International Journal of Production Research* 56, no. 8, pp. 2941-2962, 2018.
[2] R. Jiao, L. Luo, J. Malmqvist and J. Summers, "New Design: Opportunities for Engineering Design in an Era of Digital Transformation," *Journal of Engineering Design* 33, no. 10, pp. 685-690, 2022.
[3] J. M. Wing, "Computational Thinking," *Communications of the ACM* 49, no. 3, pp. 33-35, 2006.
[4] Y. Li, A. H. Schoenfeld, A.A. diSessa, A. C. Graesser, L. C. Benson, L. D. English and R. A. Duschl, "Computational Thinking Is More about Thinking than Computing," *Journal for STEM Educ Res* 3, pp. 1-18, 2020.
[5] H. Ehsan, A. P. Rehmat and M.E. Cardella, "Computational thinking embedded in engineering design: Capturing computational thinking of children in an informal engineering design activity," *Int J Technol Des Educ* 31, pp. 441-464, 2021.
[6] Diordieva, C., Yeter, I.H. and Smith, W.S., 2019, June. Middle school STEM teachers' understandings of computational thinking: A case study of Brazil and the USA. In *2019 ASEE Annual Conference & Exposition*. https://peer.asee.org/33107
[7] J. A. Lyon and A .J. Magana, "Computational thinking in higher education: A review of the literature", *Computer Applications in Engineering Education* 28, no. 5, pp. 1174-1189, 2020.
[8] J. Cuny, L. Snyder and J. M.Wing, "Demystifying computational thinking for non-computer scientists," unpublished.
[9] F. K. Cansu and S. K. Cansu, "An Overview of Computational Thinking," *International Journal of Computer Science Education in Schools* 3, no. 1, pp. 17-30, 2019.
[10] C. Angeli, J. Voogt, A. Fluck, M. Webb, M. Cox, J. Malyn-Smith and J. Zagami, "A K6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge," *Journal of Educational Technology and Society* 19, no. 3, pp. 47-57, 2016.
[11] J. M. Wing, "Computational thinking and thinking about computing," *Phil. Trans. R. Soc. A.* 366, no. 1881, pp. 3717-3725, 2008.
[12] V. Dolgopolovas, V. Dagienė, "Computational thinking: Enhancing STEAM and engineering education, from theory to practice," *Computer Applications in Engineering Education* 29, no. 1, pp. 5-11, 2021.
[13] D. Kotsopoulos, L. Floyd, S. Khan, I. K. Namukasa, S. Somanath, J. Weber and C. Yiu, "A Pedagogical Framework for Computational Thinking," *Digit Exp Math Educ* 3, pp. 154-171, 2017.
[14] H. Ehsan, T. Dandridge, I. Yeter and M. Cardella, "K-2 students' computational thinking engagement in formal and informal learning settings: A case study (fundamental)," in *2018 ASEE Annual Conference & Exposition Proceedings*. https://doi.org/10.18260/1-2--30743
[15] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative Research in Psychology*, *3*(2), pp. 77–101.
[16] J. Saldaña, *The Coding Manual for Qualitative Researchers*. Sage, 2015.
[17] M. Shahin, C. Gonsalvez, J. Whittle, C. Chen, L. Li and X. Xia, "How Secondary School Girls Perceive Computational Thinking Practices through Collaborative Programming with the Micro:Bit," *Journal of Systems and Software* 183, 2022.
[18] A. J. Magana and G. S. Coutinho, "Modeling and simulation practices for a computational thinking-enabled engineering workforce," *Comput Appl Eng Educ* 25, pp. 62-78, 2017.

[19] G. Ardito, B. Czerkawski and L. Scollins, “Learning Computational Thinking Together: Effects of Gender Differences in Collaborative Middle School Robotics Program,” *TechTrends* 64, pp. 373-387, 2020.
[20] M. Zapata-Cáceres, N. Fanchamps, I. H. Yeter, P. Marcelino and E. Martín- Barroso, “Understanding Teachers’ Attitudes and Self-Assessment Towards Computational Thinking,” *CTE-STEM 2022*, 2022.
[21] O. Yaşar, J. Maliekal, P. Veronesi, L. Little, M. Meise and I. H. Yeter, “Retrieval Practices Enhance Computational and Scientific Thinking Skills,” *Frontiers in Psychology* 13, pp. 142-154, 2022.
[22] Diordieva, C., & Yeter, I. H., & Smith, W. S. (2019, June), *Middle School STEM Teachers' Understandings of Computational Thinking: A Case Study of Brazil and the USA* Paper presented at 2019 ASEE Annual Conference & Exposition, Tampa, Florida. 10.18260/1-2--33107
[23] O. Yasar, P. Veronesi, J. Maliekal, J. Little, L. S. E. Vattana and I. H. Yeter, “Computational Pedagogy: Fostering a new method of teaching,” *Computers in Education Journal, 7*(3), pp. 51–72.
[24] V. Shamita, I. H. Yeter and E. Fong, “An Initial Investigation of Funds of Knowledge for First-Generation and Continuing-Generation Engineering Students in Singapore,” in *2022 ASEE Annual Conference & Exposition*. https://sftp.asee.org/41124