***Developing Teachers' Computational Skills through a Systematic Professional Development Course in Malaysia***

Mashitoh Hashim, Universiti Pendidikan Sultan Idris, Malaysia
Hew Mee Cheah, Malaysian Digital Economy Corporation, Malaysia
Aslina Saad, Universiti Pendidikan Sultan Idris, Malaysia
Mohd Hishamuddin Abdul Rahman, Universiti Pendidikan Sultan Idris, Malaysia

The Asian Conference on Education 2019
Official Conference Proceedings

**Abstract**
In this paper, we describe a 3-day workshop as part of a school teacher professional development course in introducing teachers to computational thinking (CT) concepts in Malaysia. The workshop emphasizes problem solving that utilizes teachers' prior knowledge in their subject area, together with CT skills to help them understand the nature and scope of problem. A total of 54 school teachers, with the majority being non-computer science major graduates were involved in this workshop. Two survey questionnaires have been used, a pre training survey- to observe teachers' perceptions of their knowledge and skills and a post training survey – to view the teachers' perceptions of the training. The survey results showed that the teachers' perceived their knowledge and skills to be poor in the pre training survey while they commented that the workshop was satisfying in the post training survey. This study contributes to the teachers' professional development through the introducing of CT across many discipline subjects.

Keywords: Teachers training, computational thinking, teacher professional development

## 1. Introduction

In August 2016, the Prime Minister of Malaysia announced that Computational thinking (CT) would be integrated into the country's school curriculum in January 2017 (Abas, 2016). Since 2017, the new standard based curriculum for primary and secondary schools has adopted the CT skills in phases which started with standard one pupils (aged 7 years old), form one (aged 13 years old) and form four students (aged 16). The implementation of CT in Malaysia can be divided into two forms: being directly taught in computer science (CS) classes or through the use of CT concepts in other subjects such as in Science, Technology, Engineering, and Mathematics (STEM) subjects. There are four CT concepts stated in the standard curriculum, namely abstraction, generalization, pattern recognition and decomposition, which need to be exposed to students. The plan to embed the CT concept in all subjects are closer as it is essential to build a good foundation for the young generation in building their competitive edge in the fast-paced digital era. As CT is an unfamiliar discipline to many teachers (Bower et al., 2017), training teachers to understand CT before transferring the skills to students is essential (Yadav, Gretter, Good, & McLean, 2017).

For that purpose, the Ministry of Education (MOE) through Malaysian Digital Economy Corporation (MDEC) have appointed Master trainer (MT) among selected university lecturers in Malaysia to give training for school teachers throughout the country. To select MTs, candidates were selected by their qualifications and working experiences. MTs also needed to undergo a few workshops and trainings as well as successfully complete systematic task evaluations before they were endorsed as a MT. The strict selection process was held to ensure that the selected MTs are qualified to train teachers nationwide as achieving the goal of CT for all requires a tight collaboration between teachers and experts (Yadav, Hong, & Stephenson, 2016). The successful selected MTs were then asked to give training to teachers throughout the country. Training for teachers are ongoing trainings that the MOE has held to ensure that teachers are aware of applicable CT concepts in their teaching and learning process.

The idea for the workshop for all teachers evolved from perceived needs in primary and secondary schools when the CT skills have been embedded in the curriculum. With this in mind, the workshop has been developed with a three-fold agenda: synthesize CT into teaching, transfer CT skills to students and have the technical skills and problem solving ability.

In this paper, we give a brief overview of the workshop and survey results from our 2018 teachers training held at Universiti Pendidikan Sultan Idris (UPSI), one of the Centre of Professional Developments (CPD) appointed by MDEC. CT has become very popular as the concept of it has been integrated in many countries as part of their new national curriculum standards (Balanskat, & Engelhardt, 2015).

## 2. Literature Overview

Obtaining the skills for 21$^{st}$ century skills that are broader than digital skills, knowledge and attitudes are essential for a student (Laar, Deursen, Dijk, & Haan, 2017) who should be able to work collaboratively, independently and creatively in future. Jobs that have emerged today such as programmers, data scientists, mobile app developers, software engineers are some of the jobs that have never existed before. Nevertheless, many existing jobs today such as basic-level medical practitioners, junior lawyers, factory workers, fast food workers, cashiers, posties, and many more are facing extinction. As technology moves very fast especially in the Artificial Intelligent (AI) field, there is a huge opportunity for creating new jobs via technology when some of the existing jobs are eliminated (Levy, 2018). Therefore, it is vitally important to impart appropriate education to the future workforce (Mohaghegh, & McCauley, 2016). But, how can we equip students with these skills for the better of their future? This is how the idea of CT takes places where it involves software-integrated education exposed to students at young age.

There are many definitions of CT. However, The International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) have collaborated with leaders from higher education ministries, industries, and K–12 education to develop an operational definition of computational thinking. They collectively define CT as "a problem-solving process that includes (but is not limited to) the following characteristics: Formulating problems in a way that enables us to use a computer and other tools to help solve them. logically organizing and analyzing data, representing data through abstractions such as models and simulations, automating solutions through algorithmic thinking (a series of ordered steps), identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources, generalizing and transferring this problem solving process to a wide variety of problems". They added that "these skills are supported and enhanced by a number of dispositions or attitudes that are essential dimensions of CT. These dispositions or attitudes include: confidence in dealing with complexity, persistence in working with difficult problems, tolerance for ambiguity, the ability to deal with open ended problems, and the ability to communicate and work with others to achieve a common goal or solution" (ISTE & CSTA, 2011).

Embedding CT across curriculum might be a promising way, however, it would be unreasonable to expect teachers to incorporate the CT concepts into their practice without adequate training to support them in teaching the skills. The vast majority of teachers perceived that they are not prepared in developing CT competences in their students (Corradini, Lodi, & Nardelli, 2017). A proper workshop should be designed to teach the new skills to teachers so that they can deliver and transfer the skills effectively to their students.

When the new curriculum is introduced, the majority of teachers are more cautious and have more concerns (Howard, & Mozejko, 2015) as they may be required to apply different teaching approaches (Hsu, Chang, & Hung, 2018). Mostly, they have lack of confidence in delivering the new teaching materials (Bower et al., 2017) as more time is needed to prepare themselves (Israel, Pearson, Tapia, Whefel, & Reese, 2015) limited technological skills and knowledge (Bower et al., 2017) as well as insufficient resources and supports for both contents and pedagogical (Yadav, Gretter, Hambrusch, & Sands, 2016). Thus, the fact that many teachers try to avoid teaching the course resulted in a shortage of qualified teachers to deliver the new curriculum (Peng et al., 2014). Providing teachers with adequate training is essential to increase their confidence level as well as to achieve the desired students' outcomes.

## 3. Methodology

### 3.1. Course Organization

The 3-day workshop consisted of 54 participants that were split into 4 batches (2 Primary Schools, 1 Secondary School and 1 Matriculation Centre). The first three batches had their workshops in April 2018 while the last batch attended the workshop in July 2018. The systematic workshop schedules prepared by MDEC consisted of understanding CT through a theoretical introductory session, unplugged activities, scratch programming practical sessions and also a Teaching demo session. The CT concepts used in this workshop were abstraction, decomposition, pattern recognition, logical thinking, algorithm and evaluation.

#### 3.1.1. Unplugged activities

As CS unplugged can be used by anyone regardless age, and the materials can make people interested to learn more about CS (Thies, & Vahrenhold, 2016), these activities are included in the training. There were three unplugged activities that were carried out during the training. The first one was "CT Realization" which was developing algorithm through an "act like a computer" and "act like a programmer" game. In this activity, two groups worked in pairs: one group acted as a programmer who wrote algorithm, and the other group acted as a computer who executed the algorithm. The programmer group was given a black and white picture as shown in Figure 1. Then, they were given the grid paper and coloured cutting shapes. With these materials, they wrote an algorithm to produce the same picture as given earlier. The computer group was given only the grid paper and coloured cutting shapes. The programmer group then read all steps written in the algorithm loudly and the computer group started following the steps. If they managed to obtain the same result, then, the algorithm was said to be a good algorithm. The group switched their roles with a different picture. This activity was performed to build awareness of the CT skills that they should use in solving the problem. It was also meant to show that executing an algorithm is a strict process where it should be understandable for a computer to carry out the task. Figure 1 shows an example of the materials used for the activity.

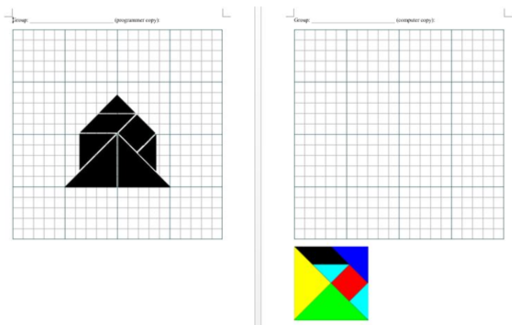

Figure 1: Materials used for the "act like a computer and a programmer" activity.

The monster face was the second unplugged activity where the participants tried to execute a given algorithm to see whether they could get the correct result. Compared to the first activity, this activity used simpler pre-defined steps as shown in Figure 2. The participants were also asked to produce a similar algorithm with pre-defined steps for their friends to try.

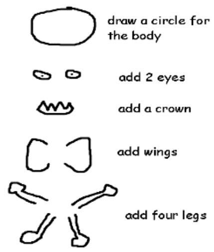


Figure 2: An example of pre-defined steps prepared to help in writing the algorithm.

The last unplugged activity was writing an algorithm with very limited instructions: use arrow symbols (left, right, up and down) and the keyword “fill in” only. Assuming that a cursor starts from the first box at the top-left corner, the participants were asked to write an algorithm to produce the given output as shown in Figure 3.

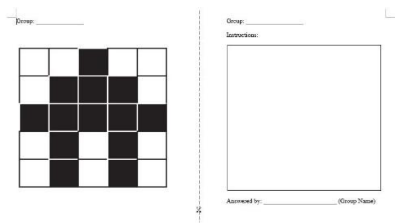

Figure 3: A limited instruction activity.

Exploring CT skills with the unplugged activities were very good exercises as it gives opportunities to participants in exploring CS in a meaningful and engaging ways (Bell, & Vahrenhold, 2018).

### 3.1.2. Scratch Session

A practical session was needed to expose participants with the CT concept. In this practical session, participants were taught about the scratch program and ways to write an algorithm using the program. Along the way, the participants were always asked to really understand what they were doing, drafted the solution by writing an algorithm first before putting them on the Scratch program. All Scripts in Scratch such as Motion, Looks, Sound, Data, Events (broadcast), Control (sequences, selection and iteration control structures), Sensing, Operators and More Blocked were used. There were many projects of different difficulties given to the participants, from easy to the difficult ones by using the appropriate scripts.

### 3.1.3. Teaching Demo session

A walkthrough to the existing curriculum of the national standard textbook was done to see how CT was embedded and how teachers can relate the content to student’s daily life experience. Discussions on how to improve the existing contents were also held. On the last day, participants were required to use the improved version for teaching contents that they have discussed earlier for their teaching demo. Here the concept of CT used in the teaching demo presented was discussed among all participants.

### 3.2. Surveys

The Pre-survey questionnaire was sent to the participants before attending the workshop to get each participants' professional background, prior knowledge on CT and expectations on the workshop. The post-survey questionnaire was completed on the last day of the workshop. Based on the preliminary investigations, we clarified research questions as:

RQ1 : How the teachers perceived their competence to teach CT?
RQ2 : How the teachers perceived the CT training?
RQ3 : Are teachers aware of what they have learnt during the workshop?

#### 3.2.1. Survey participants

The participants for this study were 54 multi-disciplinary school teachers who had attended the workshops. The Primary school teachers teach students aged 7 to 12 years-old, and the Secondary teachers teach students aged 13 to 17 years-old. The Matriculation centre basically provides pre-university courses that allows students (aged 18 to 19) to pursue a degree upon successful completion of a 1 or 2 years program at the centre. The participants were chosen by the State Education Department (JPN) of Perak, Boarding school Management Division of MOE and Perak Matriculation centre. Table 1 provides details of the participants.

Table 1. Demographic information of participants (n = 54)

| | |
|---|---|
| Gender | Female : n= 30, Male: n= 24 |
| School type | Primary Schools: n= 29,<br>Secondary Schools : n=12,<br>Matriculation: n=13 |
| Teaching experiences | >10 years: n=24,<br>8-10 years: n=6,<br>4-7 years: n=10,<br>1-3 years: n=10,<br><1 year : n=4 |
| Highest qualifications | Degree: n= 37, Diploma: n= 4, Master: n= 13 |
| Majoring | Language (Malay/English/Tamil) : n= 14,<br>Maths and Science (Science/Maths/Biology/PE): n=15,<br>Music and Art : 6,<br>Technology (IT/CS/Multimedia/Engineering): n=19 |
| Main Teaching Subjects | Arabic language : 3<br>Basic Computer Science: 8<br>Computer Science: 13<br>English: 10<br>Islamic Study: 1<br>Malay language: 4<br>Maths: 3<br>Physical Education:3<br>Science: 7<br>Tamil language: 1<br>Music: 1 |

### 3.2.2. Instrument

The pre-survey questions were designed using a three point Likert scale with teachers responding to “poor”, “fair” and “good” for their perceptions of certain knowledge and skills, while the post-survey questions used a five point Likert scale with teachers responding to each question by choosing from “strongly disagree” to “strongly agree” options. The open ended questions were also posted in both pre and post-survey questionnaires.

### 3.2.3. Survey data analysis

For this research, only frequencies, mean and standard deviations were used. The data gathered from an electronic survey was transformed to IBM SPSS Statistics.

### 3.3. Certification process

After the training ended, participants were required to submit an assignment consisted of one scratch project (with a few checklist to be followed), teaching proposal and a video that showed how they had integrated CT in their teaching. The participants were required to submit the assignment within 2 months after the training, partially phase by phase. The submission was made via Learning Management System (LMS) given by MDEC. To be certified with CT skills, participants must not only attend all the workshop sessions, but also need to submit the assignment within the given time frame.

## 4. Survey results

### 4.1. Survey results

Table 2: Perceived competencies of the teachers (n=54)

| | Mean | Std. Deviation |
|---|---|---|
| CS and Programming Skills<br>4 questions | 1.5972 | .47438 |
| Mathematical Reasoning<br>5 questions | 1.6698 | .62821 |
| Teaching Assessments<br>2 questions | 1.5093 | .57042 |

Scale: 1 = poor, 2 = fair, 3 = good

For the RQ1, the three sets of self-assessment questionnaire consisted of 3 Likert scales from 1 to 3 (“poor”, “fair”, and “good”). The first set was accustomed from Computer Science (CS) and Programming Skills, which examines teachers’ knowledge and skills in CS and programming in general, Scratch and Python programming languages. The other set examined the teachers’ perceived capability in Mathematical Reasoning related concepts such as statements, quantifiers, operations, implications, argument, deduction and induction. The last set asked about teachers’ perception of using and designing their teaching assessments. The means and standard deviations for each of the constructs, are summarized in Table 2.

As seen in Table 1, 24 teachers had more than ten years working experience and only four teachers had less than 1 year of teaching experience. Majority (n=37) had a Bachelor degree as their highest qualification, 13 teachers had Master degree and only 4 teachers had Diploma. The teachers’ majoring area and their teaching subjects seemed to have not much of a difference, however, only 19 teachers had qualifications majoring in computer science related (IT, CS, Multimedia, and Engineering), while 35 teachers came from non-computer science backgrounds.

The teachers perceived that their competency as "poor" in all aspects that had been asked (mean < 1.68). Mathematical reasoning skills scored almost fair as the mean was closed to 1.68 which was in range "fair" (1.68 – 2.35). In the programming competent questions, almost all (n=54) teachers answered that they were poor at Scratch and Python programming language. On the other hand, majority teachers who were majors in Technology (n=19) had a broad concept of CS in general, but not in programming languages.

The open ended question asked how teachers defined CT before they came to the workshop. Majority of the teachers gave a very simple sentences for example 20 teachers defined CT as "a new problem solving technique", 5 teachers said CT was "logical thinking", 3 teachers defined CT as "the way to think like a computer", 2 teachers described it as "a new skill", 2 teachers called it "creative thinking" and 2 teachers suggested CT as "coding". There were 20 teachers who left the questions unanswered.

For the RQ2, the self-assessment questionnaire asked the teachers' perceptions of the workshop, with an answering module consisting of a 5 Likert scale from 1 to 5 ("strongly disagree" to "strongly agree"). The questions were divided into 5 parts. The first part was general questions about the training such as the organization of the contents, course materials, allocation of time and the workshop control. The second part examined the teachers' perceived capability of the Master Trainers who had conducted the training which consisted of questions about how the trainer demonstrated expertise in explaining the CT concept, verbal and non-verbal teaching methods, professionalism in answering questions, time allocation for the Q&A and how MTs stimulated the learning process. The third part was on their views of CT after joining the training, followed by the last part, the BIC model. The last part was about the facilities provided during the training. The means and standard deviations for each of the constructs, are summarized in Table 3.

Table 3: Perceived of the workshop

| | Mean | Std. Deviation |
|---|---|---|
| About the training<br>9 questions | 4.4609 | .66331 |
| About the Master Trainer (MT)<br>8 questions | 4.5787 | .48764 |
| About the CT<br>5 questions | 4.4556 | .52903 |
| About the BIC model<br>8 questions | 4.4861 | .62696 |
| Facilities<br>2 questions | 4.5741 | .74230 |

### 4.2. Post-course teachers' feedback

For the RQ3, teachers were asked "what they had learnt from the training" to see their awareness of the training contents. Table 4 summarizes the feedback result.

Table 4: What teachers have learned from the training (n=54)

| | Frequency | Percentage |
|---|---|---|
| CT concepts | 51 | 94% |
| Scratch Programming | 51 | 94% |
| Integrating CT in the classrooms | 43 | 80% |
| Python Programming | 12 | 22% |
| Micro teaching | 5 | 9% |
| Classroom Management | 2 | 4% |
| No feedback | 3 | 6% |

An open question asking the teachers, "How do you hope to change your practice as a result of this training," gives the results as shown in Table 5.

Table 5: how do you hope to change your practice as a result of this training (n=54)

| | Frequency | Percentage |
|---|---|---|
| Pupils can gain a lot of benefit | 1 | 2% |
| I need to practice more | 7 | 13% |
| I will use CT in my class | 28 | 52% |
| I will plan lessons based on CT and CS skills | 6 | 11% |
| I need more time to learn | 2 | 4% |
| I will use more Inquiry-based and control | 2 | 4% |
| I will ensure more student centred activity in class | 3 | 6% |
| No feedback | 5 | 9% |

### 4.3. Certification results

Out of 54 participants, only 26 (48%) were able to complete the assignments. Even though some of the participants had submitted a partial of the assignment such as the Scratch project, they were not certified as CT Trainers as all tasks should be completed, as per the requirements stated earlier.

## 5. Discussion

In summary, the workshop was successfully well received and the objectives of the workshop were achieved based on the survey questions and feedbacks we have obtained. Our findings show that the teachers perceived they were incompetent in CS in general, mathematical reasoning and Programming language before attending the workshop. As majority (n=35) of the participants came from non-computer science major, the result is predicted. The teachers also had lack of understanding of the CT concepts based on their own definitions of CT. However, this research did not measure the teachers' technology competencies when the training ended. Even though they were aware that they have learned many new skills (such as the CT concepts, Scratch language, how can CT be integrated in the classroom and etc.), throughout the workshop, it cannot be said that they were competent as no knowledge and skills test has been done after the training. Certification process is one of the ideas initiated by MDEC to measure teachers' competencies as an initiative for continuing the assessment, however, many of the teachers failed to submit the assignment as requested. This may be due to the workload they have when the teachers return to their schools. To see whether the workshop that has been conducted gives real benefits to teachers, a continuing assessment is essential. However, it is beyond our current resources to explore the continuing assessment in further details. The findings also show that teachers were highly motivated to use the skills that they have obtained in their class, however, how far they can integrate CT in their classroom is also beyond our research area. There were three teachers who simultaneously did not give any feedback to the post-course teachers' questions. They were also identified as given low options scale for every constructs on how they perceived of the workshop resulted in quite high standard deviations as shown in Table 3. As the participants attended the workshop involuntarily, it may contribute to this result. Questions that focused on teachers' readiness in attending the workshop as well as readiness in integrating CT in the classroom should have been included in the questionnaires as attitudes toward technology is a key success for technology acceptance and integration in classrooms (Scherer, Tondeur, Siddiq, & Baran, 2018).

## 6. Conclusion

We identify the following shortcomings in our study: Measuring teachers' competencies is essential to see in what level they are at before and after enrolling in the workshop. CT test such as proposed by Román-González, Pérez-González, Moreno-León, and Robles (2018) or the Beaver International Contest on informatics and computer fluency (haberman, Cohen, & Dagiene, 2011) can be used to evaluate the CT skills. Pre-survey and post-survey for teachers' self-assessment should be done using the same constructs to see whether it shows any differences. We also have doubt whether two months given period to all teachers to submit their assignment is a good period or not as teachers usually have lots of commitments at school. Evaluating scratch projects should also be carried out using a free and open source web application, such as Dr. Scratch (Moreno-León, & Robles, 2015), was designed to analyse projects programmed with Scratch. This may be used to avoid any bias while judging the scratch project. The teachers that have been chosen by MOE to attend the workshop ready or not to embed or teach CT without prior experiences or any compulsory education in CT before. As the workshop continues to evolve, revising and refining the content is very important to ensure quality of the teachers' professional development. It is our hope to continue to have a broad impact throughout Malaysia that can help increase understanding of CT among the school teachers. As future work, we intend to develop an online module that can be used by teachers earlier in the workshop to expose them with CT beforehand.


## Acknowledgements

We want to thank the Malaysia Ministry of Education, Malaysia Digital Economic Corporation and, Universiti Pendidikan Sultan Idris, especially, the teachers who participated in the study.


## Conflict of interest

The authors declare that there is no conflict of interest in this paper.

## References


Abas, A. (2016, August 11). Computational thinking skills to be introduced in school curriculum next year. Retrieved June 18, 2019, from https://www.nst.com.my/news/2016/08/164732/computational-thinking-skills-be-introduced-school-curriculum-next-year

Balanskat, A., & Engelhardt, K. (2015). *Computing our future: Computer programming and coding - Priorities, school curricula and initiatives across Europe*(pp. 1-87, Rep.). Brussels, Belgium: European Schoolnet.

Bell, T., & Vahrenhold, J. (2018). CS Unplugged—How Is It Used, and Does It Work? *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications Lecture Notes in Computer Science,*497-521. doi:10.1007/978-3-319-98355-4_29

Bower, M., Wood, L., Lai, J., Howe, C., Lister, R., Mason, R,., . . . Veal, J. (2017). Improving the Computational Thinking Pedagogical Capabilities of School Teachers. *Australian Journal of Teacher Education,42*(3), 53-72. doi:10.14221/ajte.2017v42n3.4

Corradini, I., Lodi, M., & Nardelli, E. (2017). Conceptions and Misconceptions about Computational Thinking among Italian Primary School Teachers. In *Proceedings of the 2017 ACM Conference on International Computing Education Research*(pp. 136-144). Tacoma, WA: ACM.

Haberman, B., Cohen, A., & Dagiene, V. (2011). The beaver contest: Attracting youngsters to study computing. In *The 16th annual joint conference on Innovation and technology in computer science education (ITiCSE '11)*(p. 378). New York,NY: ACM. doi:https://doi.org/10.1145/1999747.1999891

Howard, S. K., & Mozejko, A. (2015). Teachers: Technology, change and resistance. In M. Henderson & G. Romeo (Authors), *Teaching and digital technologies: Big issues and critical questions*(pp. 307-317). Port Melbourne, VIC: Cambridge University Press.

Hsu, T., Chang, S., & Hung, Y. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education, 126*, 296-310. doi:10.1016/j.compedu.2018.07.004

Israel, M., Pearson, J. N., Tapia, T., Wherfel, Q. M., & Reese, G. (2015). Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. *Computers & Education,82*, 263-279. doi:10.1016/j.compedu.2014.11.022

ISTE & CSTA. (2011). Operational Definition of Computational Thinking - id.iste.org. Retrieved April 14, 2019, from https://id.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf

Laar, E. V., Deursen, A. J., Dijk, J. A., & Haan, J. D. (2017). The relation between 21st-century skills and digital skills: A systematic literature review. *Computers in Human Behavior,72*, 577-588. doi:10.1016/j.chb.2017.03.010

Levy, F. (2018). Computers and populism: Artificial intelligence, jobs, and politics in the near term. *Oxford Review of Economic Policy,34*(3), 393-417. doi:10.1093/oxrep/gry004

Mohaghegh, M., & McCauley, M. (2016). Computational Thinking: The Skill Set of the 21st Century. *International Journal of Computer Science and Information Technologies,7*(3), 1524-1530.

Moreno-León, J., & Robles, G. (2015). Dr. Scratch: A Web Tool to Automatically Evaluate Scratch Projects. In *Proceedings of the Workshop in Primary and Secondary Computing Education (WiPSCE '15)*(pp. 132-133). New York, NY: ACM. doi:10.1145/2818314.2818338

Peng, W. J., Mcness, E., Thomas, S., Wu, X. R., Zhang, C., Li, J. Z., & Tian, H. S. (2014). Emerging perceptions of teacher quality and teacher development in China. *International Journal of Educational Development,34*, 77-89. doi:10.1016/j.ijedudev.2013.04.005

Román-González, M., Pérez-González, J., Moreno-León, J., & Robles, G. (2018). Extending the nomological network of computational thinking with non-cognitive factors. *Computers in Human Behavior,80*, 441-459. doi:10.1016/j.chb.2017.09.030

Scherer, R., Tondeur, J., Siddiq, F., & Baran, E. (2018). The importance of attitudes toward technology for pre-service teachers technological, pedagogical, and content knowledge: Comparing structural equation modeling approaches. *Computers in Human Behavior,80*, 67-80. doi:10.1016/j.chb.2017.11.003

Thies, R., & Vahrenhold, J. (n.d.). Back to school: Computer science unplugged in the wild. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*(pp. 118-123). New York NY: ACM.

Yadav, A., Gretter, S., Good, J., & McLean, T. (2017). Computational Thinking in Teacher Education (P. J. Rich & C. B. Hodges, Eds.). In *Emerging research, practice, and policy on computational thinking*(pp. 205-220). Berlin: Springer. doi:10.1007/978-3-319-52691-1_13

Yadav, A., Gretter, S., Hambrusch, S., & Sands, P. (2016). Expanding computer science education in schools: Understanding teacher experiences and challenges. *Computer Science Education,26*(4), 235-254. doi:10.1080/08993408.2016.1257418

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational Thinking for All: Pedagogical Approaches to Embedding 21st Century Problem Solving in K-12 Classrooms. *TechTrends,60*(6), 565-568. doi:10.1007/s11528-016-0087-7

**Contact email:** mas287@yahoo.com