Preparing Primary School Teachers for Teaching Computational Thinking: A Systematic Review

Maria Ausiku(✉) and Machdel Matthee

Department of Informatics, University of Pretoria, Pretoria, South Africa
kamweshi@gmail.com, machdel.matthee@up.ac.za

Abstract. The purpose of this study is to systematically examine the existing literature on the teaching approaches and tools used to prepare primary school teachers to integrate computational thinking (CT) in their offering. In addition, the study considers perceptions of teachers towards teaching approaches for CT. Thirty (30) journal articles and conference proceedings that met the selection criteria were reviewed and thematically analysed. Teaching approaches and tools that have been used to train teachers on teaching CT in primary schools include unplugged computing, robotics, programming and game-based learning activities. Teachers' perceptions toward the teaching approaches for integration of CT were positive and progressive after interventions. To support teachers, most studies introduced modules within teacher-education curriculum or a professional development course for in-service teachers. The results indicated that most of the integrations are done within mathematics and science classes. Also, coding remains a useful way to teach CT. To prepare teachers to teach CT concepts and skills, both pre-service and in-service teachers need to practice teaching those concepts in authentic contexts. The review highlights the need for research that addresses the developing country context and working context of teachers as teaching strategies would differ from the developed countries context which has dominated the current research done.

Keywords: Computational thinking · Teacher education · Primary schools · Teachers' perceptions · Teaching approaches

1 Introduction

Computational thinking (CT) is a term that has emerged widely in the last decade. Although first used in 1980 by Papert (1980), it is only in the last decade that the term computational thinking has gained traction. Selby (2014) has inferred a comprehensive definition of CT and portrayed it as a problem-solving approach that uses decomposition, algorithms, abstractions, evaluation and pattern recognition by incorporating thought processes. In her seminal article, Wing (2006) stated that "computational thinking is a fundamental skill for everyone, not just for computer scientists". "To reading, writing, and arithmetic, we should add computational thinking to every child's analytical ability" (Wing 2006, p. 33). For this to happen, school curricula need redesigning and teachers have to be trained (Collins et al. 2011). Literature reviews on

C. Pang et al. (Eds.): SETE 2020/ICWL 2020, LNCS 12511, pp. 202–213, 2021.
https://doi.org/10.1007/978-3-030-66906-5_19

teachers' training for teaching computing are scarce, especially in primary education (elementary (K–6) education). This study presents a systematic review of existing literature on teaching strategies used to train primary school teachers for teaching computational thinking skills.

2 Background

For computational thinking to be infused in compulsory education, teachers need to be educated of what CT skills are and how they relate to their existing curriculum and what they do already on a day-to-day basis. Unfortunately, most primary school teachers lack the knowledge in content and pedagogy of computer science (CS) (Ng 2017; Rich et al. 2017b; Stanton et al. 2017). CT does not equate CS, but they are related in the sense that CS offers unique opportunities for developing computational thinking and that CT's practices can be applied to different domains besides CS. CT involves a set of problem-solving skills and techniques that software engineers use to write programs that underlie computer applications (Wing 2006). CT education, especially at primary level, is not about creating programmers or computer scientists, but it is enabling learners to solve problems using this powerful strategy and CS concepts (ISTE© - International Society for Technology in Education 2014).

Numerous related studies exist that suggest ways to prepare computational thinking teachers effectively. For example, several literature reviews have been published about teacher professional development (PD) (Desimone et al. 2002; Guskey and Yoon 2009; IAEL 2004), technology integration (e.g., (Ertmer and Ottenbreit-Leftwich 2010; Hew and Brush 2007; Lawless and Pellegrino 2007), computing education (Crick 2017; Garneli et al. 2015; Kallia 2017; Rich et al. 2017a); Waite 2017), and CT in education (Grover and Pea 2013; Heintz et al. 2016); Ilic et al. 2018; Lockwood and Mooney 2018; Lye and Koh 2014).

While wide-ranging research has been published on teachers (PD) and integration of technology in teaching, little research exists on preparing primary school teachers to teach CT skills (Yadav et al. 2019). An exception is the work of Mason and Rich (2019) who did a systematic review of literature describing ways to prepare primary school teachers (in-service and pre-service) to teach computing, coding and computational thinking. The findings suggest that programs that involve the active participation of teachers can improve teachers' computing self-efficacy, attitudes, and knowledge. This study is similar to Mason and Rich (2019)'s work in that it systematically reviews the literature on preparing primary school teachers to teach computational thinking, but the focus is on teachers' perceptions of CT teaching strategies and the teaching approaches and software tools that are used in training teachers to teach CT in primary schools.

3 Methodology

3.1 Research Questions

In an attempt to understand the current state of CT within primary schools in terms of teachers' readiness to teach and teaching approaches, the following questions were used to guide the review:

1. What teaching approaches and tools have been used in training teachers to teach CT in primary school?
2. What are the primary school teachers' perceptions of CT teaching strategies?

3.2 Review Methods

This systematic review followed Kitchenham and Charters (2007)'s guidelines for systematic reviews in software engineering research. Following these guidelines, our research methodology included three main phases: planning the review, conducting the review and reporting the results.

3.3 Data Sources

Conducting the review started with the identification of the relevant primary studies. Databases searched were electronic and concerned about the areas of education and technology/computer science. Overall, our search spanned over six digital libraries that are the most commonly used in similar studies, namely: ACM Digital Library, Springer Link, IEEE Xplore, Science Direct, Taylor & Francis Journals and ERIC. Our study collected data from conference proceedings and journals articles only.

3.4 Search Strategy

The following search string was used to search each database on the 08/03/2020: ("teachers" OR "educators") AND ("computational thinking") AND ("perceptions") AND ("teaching approaches" OR "pedagogy" OR "teaching" OR "teacher education" OR "teacher training"). All searches were made against article title and abstracts. Search filters were used on some databases to align with the screening criteria. Results of the initial search were ACM Digital Library (483), Springer Link (326), IEEE Xplore (16), Science Direct (139), Taylor & Francis Journals (226) and ERIC (25).

3.5 Study Selection

The selection process started by scanning through the title, keywords and abstract to ensure that the article is focused on our study before a paper was downloaded for a full read. The inclusion and exclusion criteria are presented in Table 1.

Table 1. Selection criteria

Inclusion criteria	Exclusion criteria
Studies that focus on teaching/integrating CT, teacher training	Studies that do not focus on the keywords or not written in the English Language
Studies that focus on CT in primary education	Studies that focus on CT at high school or university level
Studies that are published in peer-reviewed journals or conference proceedings	Materials that are not peer-reviewed (audio/video files, PPT, etc.)
Studies that presented teaching approaches or tools for teaching CT at primary level	Studies that did not answer the research questions

3.6 Study Quality Assessment

To assess the quality of the included articles, each was judged on the quality criteria of objectives, methods, results and conclusions with answer scores for the items as No = 0; Partially = 0.5; Yes = 1. A total of 30 papers were retained for the study.

3.7 Data Extraction

The extraction protocol which guided data extraction from the retained articles included research purpose, participant characteristics, description of context or setting, research design used in the research, and key findings. Extracted data were stored in an Excel spreadsheet.

4 Results

4.1 Descriptive Statistics

This section presents the quantitative results regarding the distribution of studies reviewed by publication year, what teaching approaches and tools were used. The majority of studies reviewed were carried out in developed countries with the USA dominating, only one study was done in a developing country which was Colombia and none was done in Africa.

Frequency of Publication. Figure 1 below shows the distribution of studies according to the publication year. While we didn't define a date range, data gathered revealed when publications on teachers training on CT gained attention.

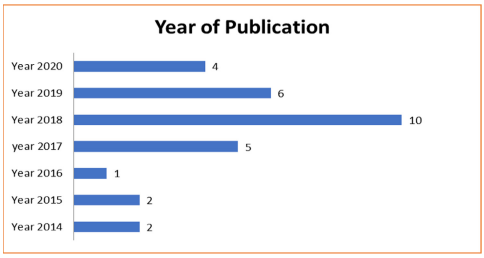

Fig. 1. Distribution of studies by year of publication

Teaching Approach Used. Figure 2 below indicates which teaching approaches were used in the studies to implement CT in the lessons, with programming being the popular approach.

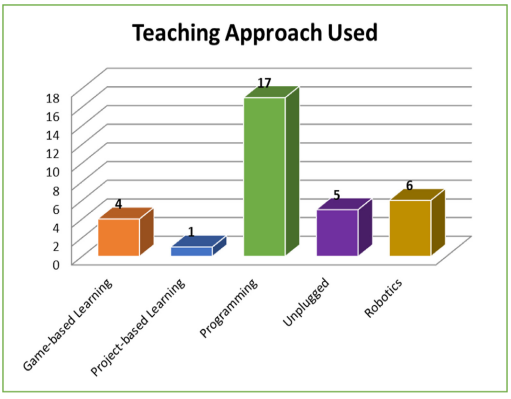

Fig. 2. Teaching approach used

Software Tools Used. Figure 3 below displays the teaching tools used in the studies to teach or integrate CT into the classrooms with Scratch activities being the most popular ones.

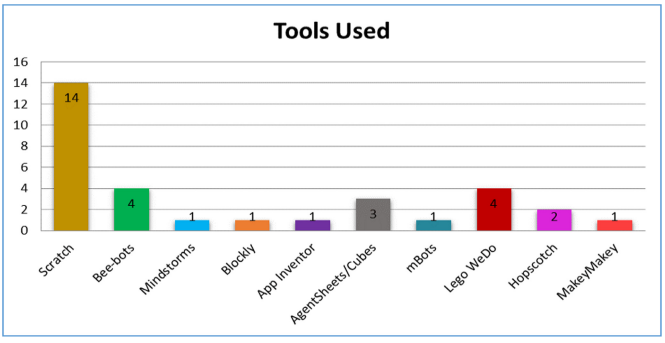

Fig. 3. Tools used

4.2 Thematic Analysis

First, extracted data were examined for patterns or trends across each extracted variable and over time. Next, the Findings or Results, Discussion, and Conclusions sections of the articles were analysed using thematic analysis, a qualitative data reduction method for identifying patterns or themes within data (Braun and Clarke 2006). The themes are grouped according to teaching approaches.

Unplugged Approach: These refer to CT activities without digital devices. Regardless of the teachers' socio-economic status, teachers can apply these concepts in different domains. Ouyang et al. (2018) and Rich et al. (2020) helped teachers to use non-programming/unplugged science and mathematics activities to present CT concepts and improve students' CT skills. Curzon et al. (2014) presented a comprehensive set of workshops for teaching primary school teachers CT concepts through unplugged activities and indicated that it works to build confidence and fill teachers' knowledge gaps about CT. Mannila et al. (2014) showed how unplugged activities and digital story-telling can be used in a variety of approaches.

Game-Based Learning and Robotics: Leonard et al. (2017) and Leonard et al. (2018) learned that game design enabled learners to take familiar concepts in STEM and apply them to a range of complex tasks to create representations and models using LEGO as a tool. Chalmers (2018) examined how primary school teachers infused robotics and coding in their classrooms in Australia using LEGO robot kits and the results determined that teachers built their confidence and knowledge. Jaipal-Jamani and Angeli (2017) and Esteve-mon et al. (2019) studied the understanding of science concepts, self-efficacy, and CT of pre-service teachers as they use robotics in a science methods course using LEGO and MakeyMakey as tools and the findings suggest that the robotics activities have increased the interest in robotics and were an effective teaching strategy to enhance CT skills and increase self-efficacy to instruct with robotics. Nickerson et al. (2015) described a framework for learning different sets of computational thinking concepts and explored how teachers employing a CT curriculum known as Scalable Game Design are using these contexts in their teaching. AgentSheets/Cubes were used as tools and it was shown that CT can be effectively taught if scaffolding is provided for learning specific skills. Game design and robotics afford teachers an opportunity to collaborate with others and engage computer scientists to deepen their understanding of computer science concepts.

Programming as Teaching Approach: A few studies show a positive response of teachers when trained to use Scratch (block-based programming) in CT. These positive responses include a greater understanding and appreciation for its usefulness (Gleasman and Kim 2020; Cetin 2016; Kong et al. 2020). Adler and Kim (2018) examined how CT can be introduced through simulations and modelling within a science method class for preservice teachers using web-based simulation and Scratch. Results showed that after the intervention, teachers had a better understanding of the topic and realized how beneficial CT is in education, and wanted to integrate it into their future classrooms. Linde-koomen (2019) and Bean et al. (2015) ran an intervention module for pre-service teachers on how to use programming and CT as a teaching strategy within

other subjects. The post-survey indicated that students intend to use Scratch in their future lessons. Falkner et al. (2018) and Bower et al. (2017) improved the CT skills of in-service teachers through workshops using programming as a teaching strategy and Hopscotch, Blockly, Scratch and Beebots as software tools. After the interventions the teachers had a detailed understanding of CT and its sub-components and have divergent ideas of various strategies they can use to teach CT in their classrooms. Marcelino et al. (2018) and Haduong and Brennan (2019) showed that Scratch can be taught and learnt through distance education effectively as teachers were able to develop quality teaching material that is useful for their classrooms. The literature shows that programming is mostly taught within a programming context, which does not help to dispel the perception that CT is about coding. Geldreich et al. (2018) and Zha et al. (2020) did a flipped module of coding activities to examine how it contributed to the pre-service teachers' knowledge, self-efficacy and attitudes towards CT. The teachers had an overall positive perception of the CT learning experience.

Project-Based Learning: Ozturk et al. (2018) has explored the use of Project-Based Learning (PBL) to integrate CT and results indicated that during collaborative project planning sessions, teachers worked closely to incorporate standards with grade-level team members. A meta-review by Hsu et al. (2018), shows that most CT training (in general), implement project-based learning, problem-based learning, cooperative learning, and game-based learning in the CT activities.

5 Discussions

The purpose of this review was to systematically examine what has been done to prepare primary school teacher to integrate CT into their lessons, explore what approaches and tools have been used and recognised research gaps.

The context of developing countries is insufficiently researched. Muñoz del Castillo et al. (2019) observed that while the teachers agree that CT skills needs to be incorporated into their training, there is a long way to go before well-organized training programs for teachers are put in place. Hence, the initial step towards preparing teachers to teach CT in developing countries is to have educational policies which promote professional development where CT takes center stage. Accordingly, teacher training curricular should be adapted to include CT skills teaching.

Fewer studies investigated the unplugged approach which can provide a way to CT skills in non-computing environments. Given Wing (2006)'s idea of CT being a fundamental skill that everyone should possess, this approach presents an opportunity to reach a large number of teachers and learners who don't have access to computing devices in developing countries. This is in agreement with Ouyang et al. (2018) and Rich et al. (2020) who indicated that using unplugged activities works to build confidence and fill teachers' knowledge gaps about CT. Unplugged activities not only introduce teachers to the CT concepts without overwhelming them with technologies and devices, but regardless of the teachers' socio-economic status, teachers can apply these concepts in different domains. This argument points in the same direction as the literature that although the subjects where CT can be integrated with the least effort are

mathematics and science, some teachers saw the possibilities of integrating within content areas such as Arts and Social studies (Lamprou and Repenning 2018; Muñoz del Castillo et al. 2019; Gadanidis et al. 2017).

The literature has revealed that programming has been the most used approach to teach CT followed by robotics. However, the current dependence on programming as a CT teaching approach may deter teachers who don't have a computing background or from schools without access to computers and programming platforms to apply CT skills in other subject domains. Schools in developing countries are not only faced with lack of computers, but also the shortage of computing skills among the teachers, which makes it hard to use programming and robotics as teaching approaches.

While every study in this review described the research context, most interventions were provided to teachers who represented different contexts. One key strategy to ensure effective CT training in developing countries is to consider the context for each teacher when training them. Teachers should be able to reflect on the teaching approach and decide if it's suitable for their classroom. Accordingly, researchers should involve teachers in the design and development of the professional development programmes or interventions to incorporate different contexts.

Practice activities that teachers did through programming mostly used Scratch as a tool and teachers' perceptions toward such CT teaching strategies were positive and improved over time. However, teachers' perceptions in developing countries are likely to differ from what the literature has revealed as the tools and approaches may not be applicable to their context.

6 Conclusions

The absence of studies based in developing countries' context is clear and more research needs to be done in developing countries especially in an African context where the teaching strategies and tools should vary accordingly. Teachers' working contexts should be considered in the choice of teaching approach and tools used. Future research should involve teachers in designing and developing interventions. Teachers need to be well prepared to help learners develop and enhance their CT capabilities so they can create technology and not just consume it.

Acknowledgements. Funding for this research was made possible by the National Research Foundation, CSIR, South Africa.

References

Adler, R.F., Kim, H.: Enhancing future K-8 teachers' computational thinking skills through modeling and simulations. Educ. Inf. Technol. **23**(4), 1501–1514 (2018). https://doi.org/10.1007/s10639-017-9675-1

Bean, N., Weese, J., Feldhausen, R., Bell, R.S.: Starting from scratch: developing a pre-service teacher training program in computational thinking. In: Proceedings - Frontiers in Education Conference, FIE 2014, pp. 1–8 (2015). https://doi.org/10.1109/FIE.2015.7344237

Bower, M., Wood, L.N., Howe, C., Lister, R.: Improving the computational thinking teaching capabilities of school teachers. **42**(3) (2017)

Braun, V., Clarke, V.: Using thematic analysis in psychology. Qual. Res. Psychol. **3**(2), 77–101 (2006). https://doi.org/10.1191/1478088706qp063oa

Cetin, I.: Preservice teachers' introduction to computing: exploring utilization of scratch. J. Educ. Comput. Res. **54**(7), 997–1021 (2016). https://doi.org/10.1177/0735633116642774

Chalmers, C.: Robotics and computational thinking in primary school. Int. J. Child-Comput. Interact. **17**, 93–100 (2018). https://doi.org/10.1016/J.IJCCI.2018.06.005

Collins, S., et al.: A model for piloting pathways for computational thinking in a general education curriculum (2011). https://doi.org/10.1145/1953163.1953243

Crick, T.: Computing Education: An Overview of Research in the Field, April, pp. 1–38 (2017). https://royalsociety.org/-/media/policy/projects/computing-education/literature-review-overview-research-field.pdf

Curzon, P., Dorling, M., Ng, T., Selby, C., Woollard, J.: Developing computational thinking in the classroom: a framework (2014). https://eprints.soton.ac.uk/369594/

Desimone, L.M., Porter, A.C., Garet, M.S., Yoon, K.S., Birman, B.F.: Effects of professional development on teachers' instruction: results from a three-year longitudinal study. Educ. Eval. Policy Anal. **24**(2), 81–112 (2002). https://doi.org/10.3102/01623737024002081

Ertmer, P.A., Ottenbreit-Leftwich, A.T.: Teacher technology change. J. Res. Technol. Educ. **42** (3), 255–284 (2010). https://doi.org/10.1080/15391523.2010.10782551

Esteve-mon, F.M., Adell-segura, J., Ángeles, M., Nebot, L., Novella, G.V., Aparicio, J.P.: The development of computational thinking in student teachers through an intervention with educational robotics. J. Inf. Technol. Educ. Innov. Pract. **18**, 139–152 (2019). https://doi.org/10.28945/4442

Falkner, K., Vivian, R., Falkner, N.: Supporting computational thinking development in K-6. In: Proceedings - 2018 6th International Conference on Learning and Teaching in Computing and Engineering, LaTiCE 2018, pp. 126–133 (2018). https://doi.org/10.1109/LaTICE.2018.00031

Gadanidis, G., Cendros, R., Floyd, L., Namukasa, I.: Computational thinking in MATHS teacher education.pdf. Contemp. Issues Technol. Teach. Educ. 17(4) (2017). https://citejournal.org/volume-17/issue-4-17/mathematics/computational-thinking-in-mathematics-teacher-education/

Garneli, V., Giannakos, M.N., Chorianopoulos, K.: Computing education in K-12 schools: a review of the literature. In: 2015 IEEE Global Engineering Education Conference (EDUCON), pp. 543–551 (2015)

Geldreich, K., Talbot, M., Hubwieser, P.: Off to new shores: preparing primary school teachers for teaching algorithmics and programming. In: ACM International Conference Proceeding Series (2018). https://doi.org/10.1145/3265757.3265783

Gleasman, C., Kim, C.: Pre-service teacher's use of block-based programming and computational thinking to teach elementary mathematics. Digit. Exp. Math. Educ. **6**(1), 52–90 (2020). https://doi.org/10.1007/s40751-019-00056-1

Grover, S., Pea, R.: Computational thinking in K–12: a review of the state of the field. Educ. Res. **42**(1), 38–43 (2013). https://doi.org/10.3102/0013189X12463051

Guskey, T.R., Yoon, K.S.: What works in professional development? Phi Delta Kappan **90**(7), 495–500 (2009)

Haduong, P., Brennan, K.: Helping K–12 teachers get unstuck with scratch: the design of an online professional learning experience. In: Proceedings of the 50th ACM Technical Symposium on Computer Science Education, pp. 1095–1101. Association for Computing Machinery, New York (2019). https://doi.org/10.1145/3287324.3287479

Heintz, F., Mannila, L., Farnqvist, T.: A review of models for introducing computational thinking, computer science and computing in K-12 education. In: Proceedings - Frontiers in Education Conference, FIE, vol. 2016-November. Institute of Electrical and Electronics Engineers Inc. (2016). https://doi.org/10.1109/FIE.2016.7757410

Hew, K.F., Brush, T.: Integrating technology into K-12 teaching and learning: Current knowledge gaps and recommendations for future research. Educ. Tech. Res. Dev. **55**(3), 223–252 (2007). https://doi.org/10.1007/s11423-006-9022-5

Hsu, T.C., Chang, S.C., Hung, Y.T.: How to learn and how to teach computational thinking: suggestions based on a review of the literature. Comput. Educ. **126** (2018). https://doi.org/10.1016/j.compedu.2018.07.004

Ilic, U., Haseski, H.İ., Tugtekin, U.: Publication trends over 10 years of computational thinking research. Contemp. Educ. Technol. **9**(2), 131–153 (2018). https://doi.org/10.30935/cet.414798

ISTE© - International Society for Technology in Education. Computational Thinking Competencies (2014). https://www.iste.org/standards/computational-thinking. Accessed 18 June 2019

Jaipal-Jamani, K., Angeli, C.: Effect of robotics on elementary preservice teachers' self-efficacy, science learning, and computational thinking. J. Sci. Educ. Technol. **26**(2), 175–192 (2017). https://doi.org/10.1007/s10956-016-9663-z

Kallia, M.: Assessment in Computer Science Courses: A Literature Review (2017)

Kitchenham, B., Charters, S.: Guidelines for Performing Systematic Literature Reviews in Software Engineering (2007). https://doi.org/10.1145/1134285.1134500

Kong, S.C., Lai, M., Sun, D.: Teacher development in computational thinking: design and learning outcomes of programming concepts, practices and pedagogy. Comput. Educ. **151** (March), 103872 (2020). https://doi.org/10.1016/j.compedu.2020.103872

Lamprou, A., Repenning, A.: Teaching how to teach computational thinking. In: Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE, pp. 69–74. Association for Computing Machinery (ACM) (2018). https://doi.org/10.1145/3197091.3197120

Lawless, K.A., Pellegrino, J.W.: Professional development in integrating technology into teaching and learning: knowns, unknowns, and ways to pursue better questions and answers. Rev. Educ. Res. **77**(4), 575–614 (2007)

Leonard, J., Barnes-Johnson, J., Mitchell, M., Unertl, A., Stubbe, C.R., Ingraham, L.: Developing teachers' computational thinking beliefs and engineering practices through game design and robotics. In: North American Chapter of the International Group for the Psychology of Mathematics Education (2017)

Leonard, J., et al.: Preparing teachers to engage rural students in computational thinking through robotics, game design, and culturally responsive teaching. J. Teach. Educ. **69**(4), 386–407 (2018). https://doi.org/10.1177/0022487117732317

Van Der Linde-koomen, D.: Computational thinking on primary education teacher education. In: EdMedia + Innovate Learning 2019 - Amsterdam, Netherlands, 24–28 June 2019 Computational, pp. 684–691 (2019)

Lockwood, J., Mooney, A.: Computational thinking in secondary education: where does it fit? A systematic literary review. Int. J. Comput. Sci. Educ. Sch. **2**(1) (2018). https://doi.org/10.21585/ijcses.v2i1.26

Lye, S.Y., Koh, J.H.L.: Review on teaching and learning of computational thinking through programming: what is next for K-12? Comput. Hum. Behav. **41**, 51–61 (2014). https://doi.org/10.1016/j.chb.2014.09.012

Mannila, L., et al.: Computational thinking in K-9 education. In: Proceedings of the Working Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference, pp. 1–29. Association for Computing Machinery, New York (2014). https://doi.org/10.1145/2713609.2713610

Marcelino, M.J., Pessoa, T., Vieira, C., Salvador, T., Mendes, A.J.: Learning computational thinking and scratch at distance. Comput. Hum. Behav. **80**, 470–477 (2018). https://doi.org/10.1016/j.chb.2017.09.025

Mason, S.L., Rich, P.J.: Preparing elementary school teachers to teach computing, coding, and computational thinking. Contemp. Issues Technol. Teach. Educ. (CITE J.) **19**, 790–824 (2019)

Muñoz del Castillo, A., Muñoz, M.A., Acosta Huertas, L.C., Herrera, E., Jimenez Toledo, J., Ramos, D.X.: Developing a teacher training curriculum including computational thinking skills: early advances on a study focused on Colombia. In: 2019 XIV Latin American Conference on Learning Technologies (LACLO), pp. 8–11 (2019). https://doi.org/10.1109/LACLO49268.2019.00012

Ng, W.S.: Coding education for kids: what to learn? How to prepare teachers? In: Proceedings of ICICTE, pp. 195–205 (2017)

Nickerson, H., Brand, C., Repenning, A.: Grounding computational thinking skill acquisition through contextualized instruction. In: Proceedings of the 2015 ACM Conference on International Computing Education Research, ICER 2015, pp. 207–216 (2015). https://doi.org/10.1145/2787622.2787720

Ouyang, Y., Hayden, K.L., Remold, J.: Introducing computational thinking through non-programming science activities. In: Proceedings of the 49th ACM Technical Symposium on Computer Science Education, pp. 308–313. Association for Computing Machinery, New York (2018). https://doi.org/10.1145/3159450.3159520

Ozturk, Z., Dooley, C.M.M., Welch, M.: Finding the hook: computer science education in elementary contexts. J. Res. Technol. Educ. **50**(2), 149–163 (2018). https://doi.org/10.1080/15391523.2018.1431573

Papert, S.: Mindstorms: Children, Computers, and Powerful Ideas. Basic Books (1980). https://dl.acm.org/citation.cfm?id=1095592

Rich, K.M., Yadav, A., Larimore, R.A.: Teacher implementation profiles for integrating computational thinking into elementary mathematics and science instruction. Educ. Inf. Technol. **25**(4), 3161–3188 (2020). https://doi.org/10.1007/s10639-020-10115-5

Rich, K., Strickland, C., Franklin, D.: A literature review through the lens of computer science learning goals theorized and explored in research. In: Proceedings of the Conference on Integrating Technology into Computer Science Education, ITiCSE, pp. 495–500. Association for Computing Machinery, New York (2017a). https://doi.org/10.1145/3017680.3017772

Rich, P.J., Jones, B.L., Belikov, O., Yoshikawa, E., Perkins, M.: Computing and engineering in elementary school: the effect of yearlong training on elementary teacher self-efficacy and beliefs about teaching computing and engineering. Int. J. Comput. Sci. Educ. Sch. **1**(1), n1 (2017b)

Selby, C.C.: How can the teaching of programming be used to enhance computational thinking skills? University of Southampton (2014). https://eprints.soton.ac.uk/366256/

Stanton, J., et al.: State of the States Landscape Report: State-Level Policies Supporting Equitable K–12 Computer Science Education | EDC (2017). https://www.edc.org/state-states-landscape-report-state-level-policies-supporting-equitable-k-12-computer-science

The Institute for the Advancement of Research in Education at AEL: Review of the Research: Nine Components of Effective Professional Development Prepared for Texas Instruments Educational and Productivity Solutions Division by The Institute for the Advancement of Research in Education at AEL (2004). www.ael.org

Waite, J.: Pedagogy in teaching Computer Science in Schools: A Literature Review (2017)
Wing, J.M.: Computational thinking. Commun. ACM **49**(3), 33 (2006). https://doi.org/10.1145/1118178.1118215
Yadav, A., Larimore, R., Rich, K., Schwarz, C.: Integrating computational thinking in elementary classrooms: introducing a toolkit to support teachers. In: Society for Information Technology & Teacher Education International Conference, no. 1, pp. 347–350 (2019)
Zha, S., Jin, Y., Moore, P., Gaston, J.: Hopscotch into coding: introducing pre-service teachers computational thinking. TechTrends **64**(1), 17–28 (2020). https://doi.org/10.1007/s11528-019-00423-0