Polemics about Computational Thinking: Digital Competence in Digital Zeitgeist – Continued Search for Answers

Lúcia Helena Martins-Pacheco[a], Christiane Annelise Gresse von Wangenheim[b] and Nathalia da Cruz Alves[c]
Department of Informatics and Statistics, Federal University of Santa Catarina, Campus Universitário s/n, Trindade, Florianópolis, Brazil

Keywords: Digital Zeitgeist, Computational Thinking, ICT Competences, K-12 Education.

Abstract: Human society is immersed in digital zeitgeist and information, and communications technology competences became fundamental for each citizen around the world in order to participate in the great benefits that this technological development promotes. In this zeitgeist, several questions appear related to the educational context because the way to deal with information, knowledge, and communications has changed drastically. For that reason, several proposals emerged to develop additional competences in youths. Among these possibilities, computational thinking – CT - arose as an approach to encourage ICT competence. However, the initial proposal appeared in a simple viewpoint article; that was just an opinion without any reference. Surprisingly, such an article reverberated around the world, and several proposals appeared trying to define "what is computational thinking?". Parallel to this, some critics showed controversial aspects of CT, especially based on computer science (CS) educational, historical trajectory. But, many solid approaches of CT based on robust researches became popular, and several educational practices were successful. Thus, we analyze some criticism and show other points of view, aiming to clarify some questions and to give some answers. We believe that CT approaches are valuable inside of myriads of possibilities to promote ICT competences. Polemics and controversies used to make advances in each field, but fruitful initiatives must be acknowledged. K-12 educational system could not wait until there is a proper and unique consensus between researchers to start to teach ICT competences to our youths.

1 INTRODUCTION

The last decade of the 20th century was featured by a great impact of information and communications technology (ICT), especially the popularization of the internet, massive dissemination of digital devices, and globalization. This time and also the beginning of the 21st century was named as digital convergence (Iosifidis, 2002), information society (Webster, 2007), knowledge society (Delanty, 2003), and digital zeitgeist[1] (Baptista and Bertolli Fo., 2012). This transformation in global society communications calls the attention of several international organizations (Delors, (1996); Gordon et al. (2009); P21 (2015), Binkley et al. (2011)) concerned with the future of education and workforce to attend economic needs. In this global context, at the end of the 20th century and the beginning of the 21st century, some studies identified that ICT literacy is an essential skill for youth development. We believe that this zeitgeist was very definitive to the massive resonance of Wing´s Computation Thinking – CT – ideas (Wing, 2006).

According to Jeannette Wing (Wing, 2006), *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* (p.33). And she added: *Computational thinking involves solving*

[a] https://orcid.org/0000-0003-3552-4421
[b] https://orcid.org/0000-0002-6566-1606
[c] https://orcid.org/0000-0002-3523-9199

[1] *The general intellectual, the moral, and cultural climate of an era* (https://www.merriam-webster.com/dictionary/zeitgeist) and https://www.psd.gov.sg/challenge/ideas/deep-dive/digital-zeitgeist

Martins-Pacheco, L., von Wangenheim, C. and Alves, N.
Polemics about Computational Thinking: Digital Competence in Digital Zeitgeist – Continued Search for Answers.
DOI: 10.5220/0009797104990506
In *Proceedings of the 12th International Conference on Computer Supported Education (CSEDU 2020) - Volume 2*, pages 499-506
ISBN: 978-989-758-417-6

problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science. Computational thinking includes a range of mental tools that reflect the breadth of the field of computer science (p.33). These ideas have profoundly reverberated in K-12 educational communities around the world. But, if we look at her article, trying to comprehend what her intention was at that time, several issues call our attention. A question is: why an article in a section Viewpoint – ACM, march, 2006, with only four pages and no references got 5892 citations (Scholar Google in 02.24.2020)? Why does just a viewpoint of a researcher achieve such an impact in K-12 related groups and researchers? This simple article significantly elicits several studies trying to define, specify, and to implement CT ideas in educational environments. In Parallel, another movement, not so strong, criticizing her ideas came alongside. Critics used to question her idea foundations, especially around computer science (CS) educational history. In this context, in this present article, we show our viewpoint about this scenario, trying to clarify that nowadays, several solid research and experiments deepened the initial simple ideas by Wing, constituting a substantial corpus of knowledge. We believe that when we presently talk about CT, the question is: what approach are you talking about? Along with this present article, we bring back some discussion to explain our vision of this situation.

2 MAIN POLEMICS ABOUT CT

Polemics, divergences, criticisms, and emphatic argumentations are common in scientific fields, and such "social activism" is crucial for the development of science in every field. Thomas Kuhn (1962), a great American philosopher of science, considered that such processes make scientific advances and the appearance of new paradigms possible. And in CS, it is not different, for example, structured programming versus object-oriented programming, which high-level language should be introduced to a beginner: Java? C++? Phyton?; which computer architecture must be taught – RISC or CISC?; free software or commercial software in educational practices?; and so on. These discussions are beneficial and sometimes do not reach a consensus. Each choice that someone makes, consequently has implicit compromises in terms of limitatios and possibilities. So, we should ponder the benefits and difficulties of each decision considering the situations and material resources that are available. To listen to different opinion and viewpoints allow us to see a broader vision of the involved scenarios. And it is no different when we talk about CT.

To analyze the author's criticisms, we consider an exploratory approach through a qualitative method. Among qualitative methods, Lenberg et al. (2017) suggest three qualitative methods: interpretative phenomenological analysis, narrative analysis, and discourse analysis. To understand the criticism about CT ideas, we chose discourse analysis because it clarifies *the essence of the problem and the underlying assumptions that enable its existence* (Lenberg et al., 2017). Sometimes it reveals implicit and unacknowledges aspects, and it can be used to any type of text. Qualitative methods are interesting to clarify complex issues that involve behaviors, emotions, values, hidden social factors as bias, prejudice, and social norms. So that, based on documents produces by critics about CT approaches, we try to analyze their arguments through reflexivity. *Reflexivity involves thinking about how our thinking came to be and how pre-existing understanding is constantly revised in the light of new insights* (p.14) (Lenberg et al., 2017).

To identify papers with arguments against CT approaches, we define the followed strategy: in the beginning, we took Peter Denning's paper titles (Denning (2009) and Denning (2017)) as a search string in Scholar Google. Denning is an acknowledged critic of CT. After the search was performed, we opened the citations of the two articles. At first, we analyze their title and exclude the ones that do not express any kind of criticisms or doubts about CT. Next, we read the abstract (if it exists) and exclude those who do not have strong arguments against CT. From the selected ones, we unfolded the references to find others. We do not intend to exhaust this subject, but just to gather some more usual opinions against CT. Our initial hypothesis is that critics did not consider several successful initiatives that use the CT approach in education, as such we found before (Martins-Pacheco et al., 2019).

Table 1 shows a summary of the main critiques. We selected 11 references with strong arguments against the CT approach. Then, in the following paragraphs, we expose some of the main criticisms. We are considering the critiques of Denning (2009), (2010), (2017), (2017a); Tedre and Denning (2016), Hemmendiger (2010), Armoni (2016), Corradini et al. (2017), Yaşar (2018), Guzdial et al. (2019), and Nardelli (2019).

Table 1: Main references with arguments against CT approaches.

REFERENCE AND ARGUMENTS (some original citations)
DENNING (2009) – Viewpoints (ACM) – 3 pages – 9 refs.
. *If we are not careful, our fascination with "computational thinking" may lead us back into the trap we are trying to escape.* (p.28) . *I am concerned that the computational thinking movement reinforces a narrow view of the field and will not sell well with the other sciences or with the people we want to attract.* (p.28) . CT movement *ignores the venerable history of computational thinking in computer science and in all the sciences*. (p.29) . *Alone, computational thinking seems like an inadequate characterization of computer science.* (p.30) . CT *seen as a repackaging—a change of appearance but not of substance. Do we really want to replace that older notion with "CS = computational thinking"?* (p. 30)
DENNING (2010) – Opening Statement Conference (ACM) – 11 pages – 32 refs.
. *The term "computational thinking" has recently become popular* (Wing, 2006)*, after hibernating many years in the jargon of our field. We are discovering that neither we in the field nor our friends outside agree on what this term means. Future education and research policies depend on the answer. We need a better answer*. (p.3)
DENNING (2017) – Viewpoints (ACM) – 7 pages – 37 refs.
. Definitions of CT *made fuzzy and overreaching claims*. (p.34) . *Unsubstantiated claims undermine the effort by overselling computer science, raising expectations that cannot be met, and leaving teachers in the awkward position of not knowing exactly what they are supposed to teach or how to assess whether they are successful*. (p.34) . *The boldest claim of all is that* CT *enhances general cognitive skills that will transfer to other domains where they will manifest as superior problem-solving skills.* (p.38) . CT *primarily benefits people who design computations and that the claims of benefit to non-designers are not substantiated*. (p.38)
DENNING (2017A) – American Scientist – 5 pages – 12 refs.
. *They launched a political movement to secure funding for computational science.* (p.14) . I*t quickly opens the door to the false belief that step-by-step procedures followed by human beings are necessarily algorithms*. (p.15) . F*uzzy definitions have made it difficult for educators to know what they are supposed to teach and how to assess whether students have learned it*. . T*he word "thinking" is not what we are really interested in—we want the ability to design computations. (…) computational design is a more accurate term.* (p.15)
TEDRE AND DENNING, (2016) – Conference – 10 pages – 80 refs.
. Seven threats: *lack of ambition, dogmatism, knowing versus doing, exaggerated claims, narrow views of computing, overemphasis on formulation, and lost sight of computational models*. (p.120) . R*isk exaggerated claims of applicability of CT*. (p.126) . *Ignoring the history and the work of the field's pioneers diminishes the computational thinking movement rather than strengthening it*. (p.127)
HEMMENDINGER (2010) – Critical Perspective (ACM Inroads) – 4 pages – 11 refs.
. The original components of CT (Wing, 2006) are not exclusive of it. . All knowledge domains use problem-solving as strategies, for example, heuristics, decomposition, recursively, and modeling, are common practices to reformulating complex problems. . It is not reasonable to decree to think like a computer scientist for people of other disciplines because each one has a proper way of thinking. Thinking well is not the province of any one discipline. (p.7)
ARMONI (2016) – Opinion (ACM Inroads) – 4 pages – 11 refs.
. CS in K-12 education has undergone two different processes: *rationalized extraction, stemming from a meaningful view of CS*. (p.24) . I*t depicts more of a reduced version of CS, just a pale image of it, deemphasizing the challenges and the thinking patterns*. (p.24) . T*he hi-tech industry, bypasses college and university CS education, and goes directly into K–12 education*. (p.27) . Block-based educational environments *use the term 'coding' instead of 'programming' or 'solving;' and promise quick learning*. (p.27) . *CT will mostly be in the hands of elementary school teachers, who are not knowledgeable in this field, we will promote a false public image of CS, far from the problem-solving discipline that it actually it is.* (…) *we will delete 30 years from the maturity of CS and its image, back to its early days, and ten from the age of CT, eliminating it altogether* (p.27). . CS is broader than programming, and programming is broader than coding.

Table 1: Main references with arguments against CT approaches (cont.).

CORRADINI ET AL. (2017) – Conference – 9 pages – 24 refs.
. CT term has a *lack of a widely accepted definition - has become a "buzzword." We are convinced this approach is wrong and misleading: in the long run it will do more harm than benefit to informatics.* (p.136) . I*n schools they do not teach "linguistic thinking" or "mathematical thinking," with specific "body of knowledge" or "assessment methods*." (p.136) . to think like a computer scientist *is required since informatics is the science underlying the digital technology pervading all aspects of contemporary society* (p.136). . In order to teach CT or informatics subjects, the teachers' conception concerning this issue is essential. They found that *the vast majority of Italian primary school teachers has not a sound and complete conception about CT.* (p.143)
YASAR (2018) – Viewpoints (ACM) – 7 pages – 34 refs.
. *Some of the remaining trouble spots include definition, methods of measurement, cognitive aspects, and universal value of CT*. (p.33) . He proposes an interdisciplinary perspective *to address both cognitive and curricular aspects of CT by merging CS education research with concepts from epistemology, cognitive and neurosciences*. (p.34) . CT educational approaches should put more emphasis on modeling and simulation (M&S).
GUZDIAL ET AL. (2019) – Viewpoints (ACM) – 3 pages – 5 refs.
. CT *movement puts the onus on the student and on the education system* the onus should be put *back on the computer scientists and other computationalists*. (p.28) *If we want better thinking and problem-solving. to improve the computing and use that to change our teaching.* (p.28)
NARDELLI (2019) – Viewpoints (ACM) – 4 pages – 21 refs.
. W*e probably need the expression as an instrument, as a shorthand reference to a well-structured concept, but it might be dangerous to insist too much on it and to try to characterize it* (…) *precisely*. (p.32) . *what is important is stressing the educational value of informatics for all students*. (p.32) . C*onsidering CT as something new and different is misleading: in the long run it will do more harm than benefit to informatics*. (p.33) . W*e should discuss what to teach and how to evaluate competences regarding informatics in primary/middle/secondary schools, and forget about teaching and evaluating competences in CT*. (p.33) . CT is not a kind of thinking *better than others*, *just that it offers a complementary and useful conceptual paradigm to describe reality*. (pp.33-4)

We found seven papers that are the point of view or opinions of just one author. In the gathered papers, the amount of references varies between five to 80. Denning (2009), (2010), (2017), (2017a); Tedre and Denning (2016), Hemmendiger (2010), and Armoni (2016) show worries about CS educational legacy and also CS reputation in the K-12 context. They consider that CT diminishes CS to programming or coding. Many of them consider that CT claims are exaggerated, *raising expectations that cannot be met* (Denning, 2017) and decreasing other knowledge domains. Besides, some of them affirm that no evidence of developing CT skills allows them to transfer them to other knowledge domain skills.

Most of the authors argue that CT conception based on Wings (2006) is vague and a repackage of older ideas that belonged to CS educational context. Corradini et al. (2017) and Nardelli (2019) consider that informatics is a more adequate term to use instead of CT.

It is important to highlight that some of these critics, despite some disagreements about CT as proposed by Wing (2006), show their definition of CT. For example, Nardelli (2019) affirms: *forget about teaching and evaluating competences in CT.* But, he shows his definition*: "Computational thinking is the thought processes involved in modeling a situation and specifying the ways an information-processing agent can effectively operate within it to reach an externally specified (set of) goal(s).*" (p. 34).

Besides the before-mentioned authors, the study accomplished by Cansu and Cansu (2019) aimed to define the concept of CT. They performed an analysis of some criticisms and contemporary related studies concerning CT in K-12. According to them, Denning (2009) and Hemmendinger (2011) showed their thesis that original definition (Wing, 2006) *of computational thinking could give the impression that computational thinking is only relevant to the field of computer science and is largely inapplicable to everyday situations in would-be computational thinking learners* (Cansu and Cansu (2019), p. 6). Cansu and Cansu (2019) examined such criticism and deducted that in *ascribing undeserved importance to certain fields – whether they are* *deemed coding, computer science, or computational thinking – would be inappropriate* (p.9). For them, many types of research hold CT *as a potential method of transforming education, as long as they also hold the criticisms applied to the field in equal regard* (p.9). According to Cansu and Cansu (2019), it is essential to ponder that an important sub-field of CS is programming, and *while primarily conducted to educate learners in the best practices of computer programming, one of its goals is being conducive to the creation of high-quality computer programs*. (p.6).

3 ISSUES ABOUT CT CRITIQUES IN DIGITAL ZEITGEIST

When we see these critiques and observe what happened from 2006 until now (2020), several contradictions and advances emerge in researches that are unconsidered in the discussion by the critics. Then we return to our initial questions: why does an article in a section Viewpoint – ACM, march, 2006, with only four pages, and no references get 5892 citations? Why just a viewpoint of a researcher achieves such an impact in K-12 related groups and researchers? Why have people been looking forward to an answer to "what is computational thinking?" Why even hard critics also try to define it?

Our understanding of such phenomenon is as follows. We dare to hypothesize some possible answers. Hu (2011) said: *Often, fruitful discussions can be more valuable than finding definitive answers* (p. 223). In philosophy, many times, to have a good question is more useful than giving precise answers. When we have open questions or an ill-defined problem about a crucial issue, which is very common in social sciences, we open several possibilities to create cognitive dynamism to find good responses for such a conundrum. It is what is called in the Gestalt the "closure principle" (Todorovic, 2008).

Considering the digital zeitgeist, there is a strong appeal of several international organizations since the end of the 20th century, which had acknowledged that digital and ICT competence would be as essential to citizens in the 21st century (Pischetola, 2019) as reading and writing. The idea of the “information society” and “knowledge society” become a reality due to the internet, popularization of digital devices, and, lately, especially due to the "omnipresence" of the smartphones. Digital inclusion has to be achieved. This idea is aligned with public governmental policies, democratic practices, and also with commercial interests. Educational systems have been concerned about how to teach and how to learn in a digital age (Bates, 2016).

Armoni (2016), concerned with a false image of CS, as mentioned before, said: *teaching CT will mostly be in the hands of elementary school teachers, who are not knowledgeable in this field* (p.27). In this sense, Wing´s article probably called the attention of the K-12 public because the language that she used was sufficiently accessible for reaching people beyond the CS expert community. Wing expressed, maybe, some aspects in an exaggerated way, but in fact, it motivated people to believe that computational thinking is important for everyone. It was a strong generalization, as we are used to doing when we are just talking, but it deeply resounds in the educational system. For everyone, we could interpret that it does not means that it has to be compulsory but a possibility available to promote digital inclusion.

The long history of computer science comes alongside digital technology advances in the availability of electronic devices and connectivity. For example, a smartphone, today present in the hand of millions of people around the world in everyday life, has about one million times more memory capacity than NASA's Apollo 11 guidance computer in 1972[2].

Thus, since the beginning of CS, a lot of things in the digital age change drastically in terms of availability for every people. There are two questions that critics of CT should be concerned about to certify the reputation of CS: What must be kept? What must be changed? They frequently return to Wing´s primordial article, but, in the interim, many other proposals to define, implement, assess CT in schools have been done, and they ignore them. For example, Kalelioglu et al. (2016), Martins-Pacheco et al. (2019), Zhang and Nouri (2019) and Moreno-León et al. (2019) did literature reviews, and they found several advances in terms of educational practices based on one of the several approaches for CT and also a myriad of different solid definitions. Recently, Moreno-León et al. (2019) found in their network of textual analysis that *neither programming nor coding emerges among the most influential words of the main CT definitions* (p.32). In this sense, the concerns of the critics are that CT approaches lead to reducing CS to programming or coding, which does not seem real.

[2] https://www.realclearscience.com/articles/2019/07/02/your_mobile_phone_vs_apollo_11s_guidance_computer_111026.html - accessed in 02.24.2020

Serious researchers that use the CT term frequently cite Wing (2006) just as a historical view, but they choose an approach better defined as ISTE and CSTA (2011) (2016), Brennan, and Resnick (2012), or other. When someone is talking about CT, the question is, "what CT approach are you considering?". Wing´s article is just an initial idea. A lot of other researches constructed frameworks, defining and specifying details of how to make CT in educational practices feasible in K-12. Some critics try to frame CT into CS, and they limit the possibilities and exclude the reality of the digital age that encompass many non-major people much beyond the CS community experts. Some authors (e.g., Mühling et al., 2015) have a proposal for CS education in K-12 and do not use CT approaches since it is just another possibility. Some critics claim to change the term CT to computational design, modeling, and simulation, informatics, or modeling of a dynamic system. Probably these terms are better adjusted to the CS or engineering community, but will non-specialized people understand them?

Other aspects that critics comment on are about vagueness and ambiguous definitions or that some concepts or aspects do not belong exclusively to CT. But, these characteristics used to appear in CS and other areas of knowledge. For example, problem-solving is a very general concept that belongs to each field of knowledge and also belongs to the every day individual/group life. Decomposition is part of the cartesian method[3] proposed in the 17th century. This old method is important in a lot of scientific fields. Another concept is the 'algorithm' that seems exclusive of CS. Nonetheless, according to the Merriam-Webster dictionary[4]: *It was formed from algorism "the system of Arabic numerals," a word that goes back to Middle English and ultimately stems from the name of a 9th-century Persian mathematician, abu-Ja'far Mohammed ibn-Mūsa al-Khuwārizmi, who did important work in the fields of algebra and numeric systems*.

So, the term 'algorithm' has hundreds of years of age, and it was taken to CS from mathematics. Even the word computation[5] had first known use in the 15th century, meaning the act or action of calculation. That sometimes is pointed out the historical use of the term CT. Some authors (for example, Moreno-Leon et al. (2019) and Denning (2017) state that Papert (1980), is the first one to use the term computational thinking. After that, it seems that only Wing (2006) took the use of this term. But, we had searched in Scholar Google (02.23.2020). In the period between 1990 and 2005 articles that use CT in the title, we found eight occurrences. Therefore the term CT was used in other scientific and technical fields without associated with CS formal concept or definition.

Concerning concepts, definitions, semantics, and meaning of words or terms, they are human cognitive and historical constructs, as CT is. The meaning of words is dynamic and changes, or it gets broader along with human history. The comprehension of Otte and Barros (2016) about concepts and definitions is: *definitions are formulated to draw conclusions and to solve technical problems*. (…) *Concepts, in contrast, are like continua relations and visions of possibilities. For them, conceptual meanings are much more ambiguous and infinitely more versatile than tools* (p.159). Concepts and definitions are developed through interactions among individuals and changes in the environment, along with the history.

Bringing these ideas back to our reflections, the initial concept of CT proposed by Wing (2006) is ambiguous, versatile, and also very ambitious. However, as it was mentioned before, in subsequent years, several serious researchers formulated and reformulated CT definitions, even the critics. It was done to solve technical (or educational) problems aiming to operationalize scholar practices along with all stages of K-12. This process has created a substantial corpus of knowledge, and it also allowed different approaches. Nevertheless, it is under development, and there are several challenges to overcome related to pedagogy and childhood development issues, teacher formation, material resources for schools, etc. As a result, there is an important contribution of CT approaches to provide ICT competence in the digital zeitgeist.

Concerning knowledge transfer from a different domain, recently, Scherer et al. (2019) found that *learning how to program a computer improves cognitive skills even beyond programming* (p.764). So, it put in check some critiques and promote new possibilities to teach and learning CT.

It is out of the scope of this present work to make an in-depth review of CT definition evolvement. But, in a recent study, Zhang and Nouri (2019) have done a systematic review related to Scratch, and also Moreno-León et al. (2019) made an in-depth general review of CT concepts. Moreno-León et al. (2019) performed extensive textual analysis and collected new CT definitions, that is: *The ability to formulate and represent problems to solve them by making use of tools, concepts, and practices from the computer*

[3] https://plato.stanford.edu/entries/descartes-epistemology/
[4] https://www.merriam-webster.com/dictionary/algorithm
[5] https://www.merriam-webster.com/dictionary/computati on #h1

science discipline, such as abstraction, decomposition, or the use of simulations. (Moreno-León et al., (2019), p. 33)

When we look back, it is possible to realize that the CT concept meaningfully evolved in comparison to the initial vision of Wing (2006). So, CT approaches are becoming more mature and certainly can make part of the multitude of possibilities to teach and to learn ICT skills.

4 FINAL CONSIDERATIONS

We presented the main controversies concerning CT, and we attempted to present other perspectives around these issues. In the present digital zeitgeist, it is necessary to cross the frontiers between CS and CT and include them inside the "computers and society"[6] field. This study could be deepened in the future to understand critics' believes and attitudes, the kind of references that they take into account aiming to delimited their epistemological foundations.

CT can be considered as a social practice. *Rather than seeking conceptual unity in computational thinking, we highlighted the different ontological commitments that cognitive, situated and critical framings bring to computational thinking and illustrated how these contextualize research with programming tools, design of applications, and classroom implementations* (p.51) (Kafai et al., 2020). For them, multiple framings and interdisciplinary perspectives for CT are desirable. We agree with them because, especially when we have K-12 as a scenario concerning pedagogy, educational/ developmental/ cognitive psychology, public policies, material resources, and so on, the diversity of possibilities must be taken into account. There is a great variety of approaches and methods, and, in general, governments define some common rules, and schools and teachers choose what is more adequate to the context where they are inserted.

Our exploratory analysis allowed us to confirm the initial hypothesis that several successful initiatives that use the CT approach in education were not taken into account by critics. Consequently, their opinions seem partial and attached to the context of experts in CS education or traditional practices.

To promote digital competences in K-12 is a broader space than just the CS field. Probably, the rules and more rigid definitions are essential for CS experts, but in K-12, the way to cope with teaching and learning has to be more flexible. We do not believe that CT is a panacea, but certainly, it is a valuable one. Polemics and controversies used to make advances in each field, but successful initiatives have to be acknowledged. K-12 educational system can not wait until there is a proper and unique consensus between researchers to start to teach ICT competencies to our youths.

[6] https://dl.acm.org/newsletter/sigcas

ACKNOWLEDGMENTS

The authors would like to thank Renata Martins Pacheco for her help with formatting and reviewing the English version of the final text.

REFERENCES

Armoni, M., 2015. Computer science, computational thinking, programming, coding: The anomalies of transitivity in K–12 computer science education. ACM Inroads 7, 4, pp. 24–27.

Baptista, J.; Bertolli Fo., C., 2012. Cultura, Identidade e o Zeitgeist Digital. IJ 8 – Estudos Multidisciplinares da Comunicação do XVII - Congresso de Ciências da Comunicação na Região Sudeste.

Bates, A.W., .2015. Teaching in a digital age: Guidelines for designing teaching and learning. Available at: https://openlibrary-repo.ecampusontario.ca/jspui/handle/123456789/276

Binkley, M., Ola, E., Herman, J. Raizen, S., Ripley, M., Rumble, M., 2011. Defining 21st Century Skills. In: Assessment and teaching of 21st-century skills. Nova York: Springer, p. 17-66.

Brennan, K., Resnick, M., 2012. New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada.

Cansu, S. K.; Cansu, F. K., 2019. An Overview of Computational Thinking. International Journal of Computer Science Education in Schools, April 2019, Vol. 3, No. 1 ISSN 2513-8359. 1-11.

Corradini, I., Lodi, M., Nardelli, E., 2017. Conceptions and Misconceptions about Computational Thinking among Italian Primary School Teachers. ICER'17, Tacoma, WA, USA, pp. 136-144

CSTA 2011. (The CSTA Standards Task Force), CSTA K–12 Computer Science Standards – Revised 2011, New York: ACM.

CSTA 2016. (The CSTA Standards Task Force), [INTERIM] CSTA-12 Computer Science Standards: Revised 2016. New York: ACM,.

Delanty, G., 2001. Challenging knowledge: the university in the knowledge society. Open University Press, Buckingham, 2001, ISBN 9780335205783

Delors, J., Mufti, I. A., Amagi, I., Carneiro, R., Chung, F., Geremek, B., Gorham, W., Kornhauser, A., Manley, M., Quero, M. P., Savané, M., Singh, K., Stavenhagen, R., Suhr, M. W. & Nanzhao, Z., 1996. Learning: The treasure within. Paris: UNESCO. Retrieved from http://unesdoc.unesco.org/images/0010/001095/109590eo.pdf.

Denning, P. J., 2009. The Profession of IT Beyond Computational Thinking. Commun. ACM. 52. 28-30. 10.1145/1516046.1516054.

Denning, P. J., 2010. What is Computation? Opening Statement. Ubiquity, an ACM publication. http://ubiquity.acm.org.

Denning, P. J., 2017. Remaining trouble spots with computational thinking. Commun. ACM 60, 6, pp.33–39.

Denning, P. J., 2017a. Computational Thinking in Science The computer revolution has profoundly affected how we think about science, experimentation, and research. American Scientist, Volume 105, pp. 13-17

Gordon, J.; Halasz, G.; Krawczyk, M.; Leney, T.; Michel, A.; Pepper, D.; Putkiewicz, E.; Wisniewski, J., 2009. Key competences in Europe: Opening doors for lifelong learners across the school curriculum and teacher education, CASE Network Reports, No. 87, ISBN 978-83-7178-497-2.

Guzdial, M., Kay, A., Norris, C., Soloway, E., 2019. Computational thinking should just be good thinking. Communications of the ACM, 28-30.

Hemmendinger, D., 2010. A plea for modesty. Acm Inroads, 1(2), 4-7.

Hu, C., 2011.Computational Thinking – What It Might Mean and What We Might Do About It. ITiCSE'11, Darmstadt, Germany.

Iosifidis, P., 2002. Digital Convergence: Challenges for European Regulation, Javnost - The Public, 9:3, 27-47, DOI: 10.1080/13183222.2002.11008805

Kafai, Y.; Proctor, C; Lui, D., 2020. From theory bias to theory dialogue: embracing cognitive, situated, and critical framings of computational thinking in K-12 CS education. ACM Inroads. 11. 44-53. 10.1145/3381887.

Kalelioglu, F., Yasemin, G., Kukul, V., 2016. A Framework for Computational Thinking Based on a Systematic Research Review. Baltic Journal of Modern Computing, 4(3), pp. 583-596.

Kuhn, T. S., 1962, The Structure of Scientific Revolutions, University of Chicago Press, Chicago.

Lenberg, P.; Feldt, R.; Wallgren, L.; Tidefors, I. & Graziotin, D.; 2017. Behavioral software engineering - guidelines for qualitative studies. Available at: arXiv preprint arXiv:1712.08341

Martins-Pacheco, L. H.., von Wangenheim, C.A. G., Alves, N. C., 2019. Assessment of Computational Thinking in K-12 Context: Educational Practices, Limits and Possibilities - A Systematic Mapping Study. Proceedings of the 11th International Conference on Computer Supported Education (pp. 292-303). CSEDU 2019.

Moreno-León, J., Robles, G., Román-González, M. y Rodríguez, J. D., 2019. Not the same: a text network analysis on computational thinking definitions to study its relationship with computer programming. RIITE. Revista Interuniversitaria de Investigación en Tecnología Educativa, 7, 26-35. Doi: http://dx.doi.org/10.6018/riite.397151

Mühling, A., Ruf, A., Hubwieser, P., 2015. Design and First Results of a Psychometric Test for Measuring Basic Programming Abilities. London, ACM

Nardelli, E., 2019. Do We Really Need Computational Thinking? Communications of the ACM | February 2019, Vol. 62, NO. 2

Otte, M.F., Luiz G. X. de Barros, L. G. X., 2016. What is the Difference Between a Definition and a Concept? Science Journal of Education. Vol. 4, No. 5, pp. 159-168. DOI: 10.11648/j.sjedu.20160405.14

P21 (Partnership for 21st Century Skills), 2019.Framework for 21st Century Learning Definitions, Available at http://www.battelleforkids.org/networks/p21/frameworks-resources.

Papert, S., 1980. Mindstorms: Children, Computers, and Powerful Ideas. Basic Books.

Pischetola, M., 2019. Inclusão Digital e Educação: a nova cultura da sala de aula. Ed. Vozes; Rio de Janeiro, Brazil. ISBN 978-85-326-6196-8.

Scherer, R., Siddiq, F., Viveros, B. S., (2019) The Cognitive Benefits of Learning Computer Programming: A Meta-Analysis of Transfer Effects. Journal of Educational Psychology, Vol. 111, No. 5, pp. 764–792. https://doi.org/10.1037/edu0000314

Tedre, M. and Denning, P.J., 2016.The long quest for computational thinking. In Proceedings of the 16th Koli Calling Conference on Computing Education Research, pp. 120–129.

Todorovic, D., 2008. http://www.scholarpedia.org/article/Gestalt_principles">Gestalt principles. Scholarpedia, 3(12):5345. accessed Feb 2020

Webster, F., 2007. Theories of the Information Society. London: Routledge,
https://doi.org/10.4324/9780203962824

Wing, J. M., 2006. Computational thinking. Communications of the ACM, 49(3), 33-35.

Yaşar, O., 2018. A new perspective on computational thinking. Communications of the ACM. 61. 33-39. 10.1145/3214354.

Zhang, L., Nouri, J., 2019. A systematic review of learning computational thinking through Scratch in K-9. Computers & Education. https://authors.elsevier.com/a/1ZIfP1HucdHyVb. 103607. 10.1016/j.compedu.2019.103607.