© University Press 2023
ISSN 2719-6550
ISSN 2719-7417 online

"Journal of Education, Technology and Computer Science"
No. 4(34)/2023
www.eti.ur.edu.pl

Received: 21.01.2023
Accepted for printing: 15.12.23
Published: 29.12.2023
License: CC BY-SA 4.0

DOI: 10.15584/jetacomps.2023.4.19

MILAN KLEMENT [1], **LUCIE BRYNDOVÁ** [2]

Possibilities of Development of Informatics Thinking of Pupils and Students Using the Model of Subject-Didactic Competences of Teachers

[1] ORCID: 0000-0001-9964-4057, Professor, Head of the Center of the Information and Education Technology, Palacký University Olomouc, Department of Technical Education and Information Technology, Czech Republic

[2] ORCID: 0000-0003-3534-7621, Ph.D. student, Head of the Center of the Information and Education Technology, Palacký University Olomouc, Department of Technical Education and Information Technology, Czech Republic

Abstract

The development and incorporation of the concept of the development of computational thinking into the curriculum of computing subjects is currently one of the major challenges facing the Czech school system. However, such a concept of teaching presupposes a targeted development of content-subject and didactic- pedagogical competencies of teachers necessary for the development of computational thinking in their pupils and students. Thus, the present paper deals with the possibility of defining a general model of subject-matter and didactic competences of teachers supporting the development of computational thinking in pupils and students.

Keywords: computational thinking, content-subject competences, teachers, model of competences

Introduction

The accelerating development of technology has brought about many dramatic changes in all aspects of life over the last decades and has undoubtedly affected the functioning of our society. The expansion of the digital space, technological innovations leading to the modernisation of industry, commerce and households have given rise to a plethora of new concepts relating to digital and information technologies and their use.

One of them was Computational Thinking, introduced in 2006 by Jeannette Wing as an inevitable skill of a modern person who is able to fully use digital technologies and computer methods to solve everyday problems. According to Wing, Computational Thinking is a thought process that allows one to formulate a problem and describe its solution in a way that can be effectively processed by a computer, machine, or even a human (Wing, 2014). Thus, it can be said that in general terms, it is a way of solving a problem that focuses on describing, analyzing, and finding an effective way to solve it, emphasizing systematicity and the use of concepts known in the field of computer science. It is important to emphasize that the development of computational thinking does not only mean programming, the related competences can be applied by anyone, not only by a professional computer scientist, and thus contributes to the holistic development of the pupil or student with an overlap into the development of his/her digital competences.

Informatics thinking and its implementation

Since the first introduction of the concept of Computational Thinking (hereafter abbreviated as CT), there has been a great deal of international discussion regarding its precise definition, the specification of its components, and also efforts to integrate the development of CT into curricula within school systems virtually worldwide. The introduction of the concept of computational thinking into academic discourse has fostered a pedagogical discourse regarding the role of digital technologies in education and the potential of introducing computer science and programming education into national curricula that has existed since almost the beginning of the millennium (Tran, 2017; Klement, 2018). Although computer literacy instruction and the targeted development of digital and communication competencies are still of considerable importance, there is a tendency to move the targeted development of these competencies to the cross--curricular domain as part of the modernization of the whole school system (Balanskat, 2018).

Practically since the beginning of the international discussion on the integration of the development of computational thinking into education, attempts have been made to define specific sub-areas of CT. The primary goal of this process is to specify an otherwise very general definition of the phenomenon of computational thinking, which is not suitable for the practical implementation of CT in the school system. Currently, most state curricular definitions of computational thinking are based on, or largely coincide with, the 2011 CSTA and ISTE definitions of the characteristics and competencies associated with CT use.

Also for pedagogical and didactic purposes, the specification of the areas defining CT is usually done by a detailed analysis of the formulations of the CSTA & ISTE document. The following comparative Table 1 lists the subcomponents of the informatics according to the CSTA & ISTE definition and the key words and phrases used in this definition according to Chen (2017), on the basis of which we define the corresponding CT skills (Angeli, Nicos, 2020; Bocconi, Chioccariello, Dettori, Ferrari, Engelhardt, 2016; Wing, 2014; Selby, 2012) that are related to these concepts and that the informatics-minded learner should acquire.

Table 1. Definition of areas for the development of computational thinking

CSTA & ISTE definition	Keywords	Matching CT skill
Problem formulation for machine solutions	Formulation	Syntax, programming
Logically organize and analyze data	Data	Data processing
Represent data using abstractions	Representation	Modelling
Automating solutions using algorithmic thinking	Algorithmic thinking	Algorithmization, automation

At present, the vast majority of countries are still in the process of implementing the development of computational thinking in school curricula, and for many of them this is a very challenging to radical change. Implementation of the revision is particularly problematic in states that do not have an established tradition of teaching programming in schools and thus must revise virtually all areas of instructional management. The practical integration of CT development into school curricula usually depends on individual schools, their technological and economic background, resources, time availability, the qualifications of individual teachers, and other circumstances. It is these circumstances that often make implementation difficult because schools do not have sufficient support to facilitate curriculum revision.

The lack of support for educators is highlighted by virtually all states that have moved to implement CT in curricula in the last decade. Yet, the experience of many states that have already successfully implemented CT development in their public school systems suggests that an important aspect of implementing any form of IT curriculum revision is the preparation of detailed teacher support materials.

Difficulties accompanying the implementation of informatic thinking in educational practice

The main problem, apart from economic and organizational aspects, is the increased demands on the qualifications and competences of teachers brought by the integration of programming into schools. It is programming, regardless of the global conception and definition of the term computational thinking, that is considered the most effective tool for the development of computational thinking (Román-González, Pérez-González, Jiménez-Fernández, 2017).

From a didactic and pedagogical point of view, the practical application of its concepts such as algorithmization, decomposition, generalization, evaluation and abstraction is essential for the development of CT (Angeli et al., 2016). If the goal of curriculum revision and the development of computational thinking in schools is to prepare graduates to use information technology in life and in the labour market, it is crucial that they are able to use their skills and abilities. However, such a design of teaching presupposes a targeted development of content-subject and didactic-pedagogical competencies of teachers necessary for the development of computational thinking in their pupils and students. Thus, it is not only about content competences, focused on the knowledge of algorithmization, decomposition, generalization, evaluation and abstraction, but also about didactic competences, enabling the development of cognitive, affective and psychomotor components of students' personality.

Therefore, continuous research on material conditions, analysis of educational content, forms and methods of teaching, as well as the readiness of teachers, including the necessary competences for the development of computational thinking in their pupils and students, is a necessity. In Europe, for example, European Schoolnet has been mapping the problems of implementing CT development in schools. According to the results of this research, the most important shortcoming is precisely the insufficient qualification of teachers, especially for teachers of lower grades (Balanskat, 2018). Many research activities in this area can be noted, which deal with the definition of CT content (e.g. Brennan, 2012; Kanemune, 2017; Moller, Crick, 2018, etc.), methods of teaching CT (e.g. Rubio, Romero-Zaliz, Mañoso, de Madrid, 2015; So, Jong, Liu, 2020, etc.), forms of CT teaching (Román-Gonzáles et al., 2017; Tran, 2017; Tang, Yue, Lin, Hadad, Zhai, 2020, etc.). Even less research focuses on teachers themselves and their level of preparedness for teaching CT in terms of content-subject and didactic-pedagogical competencies (e.g. Rambousek, 2013; Cheng, 2019; Klement, Dragon, Bryndová, 2020).

Our contribution in the area of describing content-subject competences of teachers for the implementation of CT development teaching was research (Klement et al., 2020), which, at least in the case of the Czech Republic, analysed this issue. Our aim was to identify different groups of respondents within the research sample (a total of 123 teachers of computer science subjects at the second level of primary schools) who declared the same or similar level of evaluation of the importance of individual subject units of computer science, and to describe their characteristics and, if necessary, to correct the negative impact of certain groups of respondents on the results of the research. This was achieved by using cluster analysis, which in this case analysed the clusters within the group of teachers to see if there were groups of teachers who reported similar levels of importance attached to each thematic unit in ICT education.

In this way, the computer science teachers were also divided into groups that showed similar variance in ratings. Simply put, if there were several thematic units that the respondents evaluated mostly in the same way in terms of their importance, these teachers formed a separate cluster (the essence, course and results of the survey are described in detail in the publication: Klement et al., 2020).

Methodology. Teachers and theirreadiness to implement computational thinking in teaching

On the basis of the analyses conducted, it can be concluded that the assumption we made about the possibility of typology is confirmed and that there is evidence of common sorting variables that can divide different groups of computer science subject teachers into separate groups according to their preferences for specific subject units. That is, a model has been found that characterizes the different subgroups within the group of teachers of computer science subjects in primary 2. Based on the identified model, different groups of respondents in the research sample were identified that show the same or similar level of evaluation of each thematic unit of computer science according to their degree of importance for teaching, so that it is also possible to describe their characteristics in more detail, as shown in Table 2.

Table 2. Groups of teachers according to the level of importance they place on the selected IT thematic units

Group of teachers	Typical thematic units preferred by the group of teachers	Overall characteristics of the group
1 – teachers preferring the development of students' computational thinking	Algorithmization and programming Working with databases Robotics and el. kits Administration and operation of computer networks	A group of teachers is interested in implementing teaching "non-traditional" thematic units, focused on the most challenging tasks related to the operation of information systems.
2 – teachers preferring to develop students' interaction skills	Working with touch devices Working with sound and video Creating web pages	A group of teachers are interested in implementing training in web services and social networking, for communication or information sharing purposes.
3 – teachers preferring to develop students' digital literacy	Computer hardware and software Work with spreadsheet calculator Working with the text editor Searching for information on the Internet Work with presentation applications	The group of teachers is interested in teaching in purely "traditional" thematic units consisting mainly in the creation and editing of documents, presentations, tables or simple graphics.
4 – teachers preferring to develop students' visualisation skills	Work with computer graphics Work with technical graphic systems Manage files and folders	A group of teachers is interested in implementing training in the use of IT tools for presentation or self--presentation in graphic form.

The above model of groups of teachers of IT subjects at the second level of primary schools can be interpreted in such a way that there is a significant group of teachers who prefer educational content focused on the development of digital literacy, i.e. on "traditional" thematic units (42.5% of teachers of IT subjects). There is also a group of pupils who prefer educational content focused on the development of computational thinking (26.8% of teachers of computing subjects). Thus, these two groups of teachers perceive the use of IT resources as a necessary condition for the further professional development of their pupils, as they attach the highest importance to those IT units that enable productive use to perform either purely "professional" tasks or tasks related to "user" use.

Furthermore, it is possible to identify a group of teachers who prefer to use IT resources more for personal development within the social interaction of their students, as they prefer thematic units, their knowledge can now also be used in the field of information sharing or establishing and maintaining personal contacts and connections using social networks or related web services (26.8% of teachers of computer science subjects). The relatively smallest group of teachers are those who prefer thematic units focused on static or dynamic graphics (4.9% of IT teachers).

The descriptions of the individual groups and their intentions may be interpreted in different ways, which we fully admit, and it would be necessary to obtain further data on the basis of which these facts could be examined more closely. It is also necessary to emphasize that the developed model dealt only with a part of the content-subject competences of teachers, for the implementation of teaching aimed at the development of CT, and did not deal with the issue of equally important didactic-pedagogical competencies of teachers. Again, this is also the intention of our further scientific work in this area.

Results. A theoretical model of teacher preparation for the development of computational thinking

In the Czech Republic alone, there are more than 7,000 teachers of informatics at the primary and secondary level, representing 8.9% of the total number of teachers (see the document of the Ministry of Education and Science, Zprava_MiS3, 2020), while according to the statistics available to us, the situation in other European countries is similar (see European Schoolnet). In the context of innovative trends in computer science education based on the development of computational thinking, which presuppose a different approach not only to educational content (hard skills) but also to the methods of teaching it (soft skills), there is a need for international comparative research on this issue. Considering the fact that only a quarter of existing IT teachers are implementing or are ready to implement teaching aimed at CT development (see Table 2), there is a clear society-wide demand for an analysis of the causes of this situation and the creation of a suitable model and tool for diagnosing their competences in this area. Such a model would not only allow for the diagnosis of the level of competences needed, but would also allow for the targeted development of the knowledge and pedagogical potential of CT teachers in the necessary areas in undergraduate, postgraduate or continuing education.

The proposed model is based on the basic five domains of CT, which generally describe the definition of CT and are most often found in pedagogical, didactic and legislative documents (Klement et al., 2020):

1. Abstraction is considered to be the most important component of computational thinking (Wing, 2014). In the context of CT, it is the ability to simplify a problem to its basic form so that essential information is not lost, and then to work with the schematic form of that problem. They work with defining patterns, generalization, representation, simulation, implementation, parameterization, optimization, and other skills necessary to successfully solve complex problems.

2. Algorithmic thinking is the ability and skill to find efficient and economical solutions to a problem and to formulate these solutions in an adequate way. It is a matter of debate whether, in the context of CT, it is the formulation of a solution in a formal programming language, or whether it is a concept completely independent of practical programming. Sometimes a separate component of Automation is defined from this area, i.e. simplifying the process to save time and energy. More often, however, it is understood as one of the principles of algorithmization (Angeli, 2016).

3. Decomposition, associated by some authors or directly identified with modularization, is the ability to divide the whole into subcomponents and to work with these components. It is closely related to abstraction because it involves solving the subparts of a problem. In the pedagogical area of CT development, it is typically associated with working with the subparts of a sequence and optimizing them through the use of functions.

4. Systematic evaluation, which includes concepts such as analysis, debugging, and analysis, is reasoning that allows predicting the outcome of a situation and the operation of an algorithm based on a critical analysis of the situation. It works with testing, analytical and logical thinking, variation and evaluation.

5. Generalization is the identification of similarities and connections between problems, which should result in the design of a universal solution to the problem, applicable to multiple situations. It is the ability to solve multiple problems based on similarities, as well as the use of learning based on the similarity of a problem to a previous one.

The theoretical design of the model includes not only the subject-subject dimension, i.e. the content component of competences, but also includes the didactic-pedagogical dimension, which allows the study and derivation of the relationship between knowledge and the ability of targeted CT development, which forms the integrating dimension of the model. Both dimensions contain identically five CT domains each, from the perspective of application within the content-subject focus and from the perspective of application from the didactic-pedagogical focus. The integrative dimension allows to understand the links between the content and the level of competences aimed at factual knowledge and orientation in CT issues on the one hand, and on the other hand the level of competences aimed at the ability to convey CT issues to pupils and students.

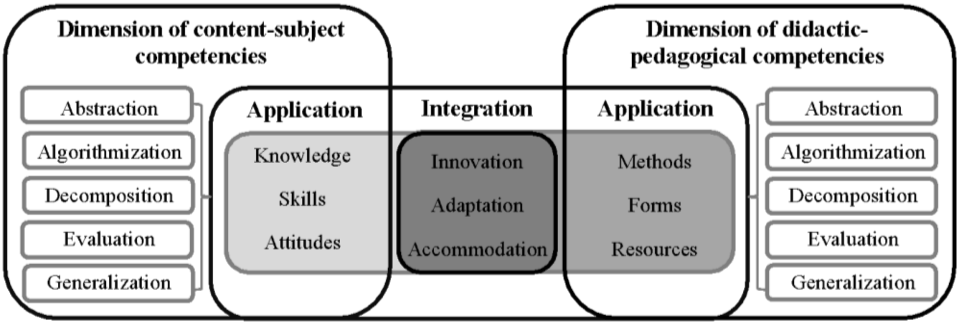

Figure 1. Proposed model of subject-didactic competences of teachers reflecting the development of students' computational thinking

Procedure for further verification of the theoretical model

The described model of subject-didactic competences of teachers reflecting the development of computer thinking of pupils and students will have to be specified in the first phase and filled with specific items of the CT domains, which will be developed on the basis of the discussion of the involved experts from the four countries, taking into account regional specificities and needs. The items will be defined both for the vocational-subject dimension, in each of the five domains, and for the didactic-pedagogical dimension, again in all five domains of this dimension.

These items and their arrangement will be verified in the first phase on the basis of qualitative-quantitative methods of pedagogical research, i.e. using Q-methodology. The Q-methodology, or also Q-sorting, works with a set of statements (in our case, the items of the dimensions of the model) that represent possible answers to the specific content of the subdomains (abstraction, algorithmization, etc.). Respondents, thus, will have the opportunity to express their level of agreement with the importance of each item within the CT domains, but also within the whole dimension. An analogous procedure will be followed within both dimensions and all five CT domains.

Respondents will compare the statements to each other and will assign levels of importance from highest to lowest, with each level of importance occupied by a completely defined number of statements. The sorting at the domain level will be adapted to this number so that the edges always contain fewer statements than the middle (the sorting follows a Gaussian curve). This procedure will force respondents to really think about priorities and identify those items that are most important for a given CT domain.

The model thus created and filled with content will be further validated using advanced non-parametric statistical methods of factor and cluster analysis. The aim of this phase of the research will be to validate the structure of each dimension of the module and its CT domains and to identify the preferred specific applications (Knowledge, Skills, Attitudes vs. Methods, Forms and Resources) of each dimension in the form of evaluation factors/criteria. For these purposes, a research instrument in the form of a questionnaire will be constructed and redistributed to respondents in all participating countries.

The initial research method used was factor analysis (McDonald, 1991, p. 230), which is a statistical method used to extract important combinations of factors with a high degree of correlation from a large data set. Thus, factor analysis allows finding latent (indirectly observed) causes of variability in the data. By finding the latent variables (factors), the number of variables can be reduced while retaining the maximum information, and it is also possible to find relationships between the observed variables and the derived factors. Factor analysis is one of the multivariate statistical methods (nowadays rather a group of methods) that originally originated in the evaluation of psychological test results. Later it was also applied in many other fields – technology, economics, anthropology, etc. It belongs, like principal component analysis, to the so- called variable reduction methods. In factor analysis, we assume that each input variable can be expressed as a linear function of a small number of common (hidden) factors and a single error factor.

In addition to nonparametric tests for dependent samples, which are designed for ordinal variables and require specifying the similarity of the variables to be surveyed, there are clustering methods. Since the dissimilarity of groups of variables is simultaneously detected, these tasks are referred to as segmentation in the current literature (especially in the context of the term "data mining") (Řezanková, 2010, p. 188).

Thus, by applying this procedure we verify not only the statistical validity of the individual dimensions of the theoretically proposed model, but also its specific items and their importance. On the basis of the cluster analysis it will also be possible to verify and prove statistically erasure factors and links integrating the dimensions when the necessary blending of the content-subject dimension and the didactic-pedagogical dimension occurs. On the basis of this fact it will also be possible to construct an evaluation tool with the help of which it will be possible to realize the diagnosis or self-diagnosis of the level of teachers' competences for the development of CT.

Conclusion

The current trend of implementing the development of computational thinking in national curricula is necessary to modernize the school systems of developed countries and to respond to accelerating technology and labor market developments. In many countries, this implementation builds on the long-heralded integration of programming into national curricula, or extends on the already established tradition of such teaching. These renovations are intended to ensure equity in basic computer science education, which in the past has been tied only to leisure activities or electives, and thus has not led to the formal development of computer literacy for the entire population.

It was the need for widespread formal development of the population in computer science and programming that helped popularise the concept of computational thinking and the associated renovation of the state curriculum. Informatics thinking, although still not precisely defined, is generally understood as a set of cognitive processes that lead to the solution of a problem in such a way that this solution is machine-processable and feasible. It is therefore a prerequisite for further education and development in computer science and programming, and a competence that is becoming indispensable as technology is gradually integrated into everyday life. The specific definition of computational thinking and its subcomponents for state education is regulated by the legislation of the state and is therefore completely individual for each education system.

From the point of view of the development of computer thinking in schools, its greatest contribution can be seen in the education of the general population in the field of programming and the principles of modern technology, facilitating adaptation to the new technologies coming and in supporting their creative use at work and in everyday life. Although the concept of computational thinking is not necessarily tied to programming, when its development is implemented in the classroom for practical application reasons, programming is appropriate. In this practical implementation, the use of specialized teaching aids and tools is suggested among which online tools are highlighted which will be discussed in detail in the following chapters, propaedeutic programming environments and educational robotics.

Despite the fact that the concept of computational thinking has an irrefutable interdisciplinary potential, its implementation in cross-curricular teaching is currently impossible in many systems due to the lack of teacher qualifications, lack of subject-specific materials and often economic support. However, in many countries, despite these difficulties, the interdisciplinary potential of developing computational thinking is supported by legislation. It is therefore highly likely that in the future there will be efforts to integrate the development of computational thinking and programming interdisciplinarily into public schools, and these tendencies are already evident, especially in the Nordic countries of Europe.

References

Angeli, Ch., Nicos, V. (2020). Developing young children’s computational thinking with educational robotics: An interaction effect between gender and scaffolding strategy. *Computers in Human Behavior*. doi:10.1016/j.chb.2019.03.018. ISSN 07475632.

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., Zagami, J. (2016). A K-6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge. *Educational Technology & Society, 19*(3), 47–57.

Balanskat A., Engelhardt K., Licht, A.H. (2018). *Strategies to include computational thinking in school curricula in Norway and Sweden – European Schoolnet's 2018 Study Visit*. Brussels: European Schoolnet.

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K. (2016). Developing computational thinking in compulsory education. *Implications for policy and practice*.

CSTA & ISTE (2011). *Operational Definition of Computational Thinking for K-12 Education*.

Kanemune, S., Shizuka, S., Tani, S. (2017). Informatics and Programming Education at Primary and Secondary Schools in Japan. *Olympiads In Informatics, 11*(1), 143–150. doi: 10.15388/ioi.2017.11.

Klement, M. (2018). Traditional topics for the framework educational programme focused on ICT area, and the perception of these topics by the primary school ninth grade pupils. *Journal of Technology and Information Education, 10*(1), 43–62.

Klement, M., Dragon, T., Bryndová, L. (2020). *Computational Thinking and How to Develop it in the Educational Process*, Olomouc: Publishing UP.

McDonald, R.P. (1991). *Factor analysis and related methods in psychology*. Prague: Academia.

Moller, F., Crick, T. (2018). A university-based model for supporting computer science curriculum reform. *Journal of Computers in Education, 5*(4), 415–434. doi:10.1007/s40692-018-0117-x.

Rambousek, V. et al. (2013). *Development of information technology competences in primary schools*. Prague: Czech Technical University.

Řezanková, H. (2010). *Analysis of data from questionnaire surveys*. Prague: Professional Publishing.

Román-González, M., Pérez-González, J.-C., Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Computers in Human Behavior, 72*, 678–691.

Rubio, M.A., Romero-Zaliz, R., Mañoso, C., de Madrid, A.P. (2015). Closing the gender gap in an introductory programming course. *Computers & Education, 82*, 409–420.

So, H., Jong, M.S., Liu, C. (2020). Computational Thinking Education in the Asia Pacific Region. *Asia-Pacific Edu Res, 29*, 1–8. doi:10.1007/s40299-019-00494-w.

Tang, X., Yue, Y., Lin, Q., Hadad, R., Zhai, X. (2020). Assessing computational thinking: A systematic review of empirical studies. *Computers & Education, 148*. doi:10.1016/j.compedu.2019.103798.

Tran, Y. (2017). Computational Thinking Equity in Elementary Classrooms: What Third-Grade Students Know and Can Do. *Journal of Educational Computing Research, 57*(1), 3–31.

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

Wing, J.M. (2014). computational thinking benefit society. *Social Issues in Computing blog*.