

# Computational Thinking Relationship with Digital Competence

Anita JUŠKEVIČIENĖ, Valentina DAGIENĖ

*Vilnius university Institute of Data Science and Digital Technologies  
Akademijos str. 4, LT-08663 Vilnius, Lithuania  
e-mail: anita.juskeviciene@mii.vu.lt, valentina.dagiene@mii.vu.lt*

Received: May 2018

**Abstract.** The European Commission Science Hub has been promoting Computational Thinking (CT) as an important 21st century skill or competence. However, „despite the high interest in developing computational thinking among schoolchildren and the large public and private investment in CT initiatives, there are a number of issues and challenges for the integration of CT in the school curricula“. On the other hand, the Digital Competence (DC) Framework 2.0 (DigCom) is promoted in the same European Commission Science Hub portal. It shows that both topics have many things in common. Thus, there is the need of research on the relationship between CT and digital competence.

The goal of this paper is to analyse and discuss the relationship between DC and CT, and to help educators as well as educational policy makers to make informed decisions about how CT and DC can be included in their local institutions. We begin by defining DC and CT and then discuss the current state of both phenomena in education in multiple countries in Europe. By analysing official documents, we try to find the underlying commonness in both DC and CT, and discover all possible connections between them. Possible interconnections between the component groups of approaches are presented in Fig. 1.

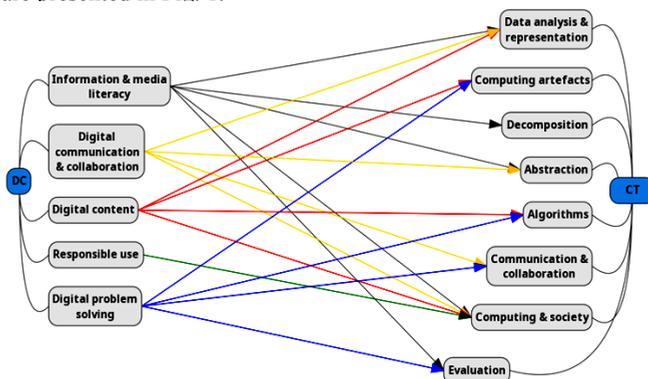


Fig. 1. The interconnections between DC and CT.

**Keywords:** computational thinking, digital competence, education.

## Background

Computational Thinking (CT) and Digital Competence (DC) are indicated by many education policy makers as important 21st century skills. The European Commission Science Hub has promoted CT and has launched the Digital Competence (DC) Framework 2.0 (DigCom) in its portal. Nowadays CT and DC are essential skills and the young generation should learn them for life.

During the last years, a lot of research work was devoted to both topics, and enormous amount of studies and practical experience have been performed and gained. Nevertheless, there is a huge need of research in these topics on many aspects. One of them is the relationship between computational thinking and digital competence.

This paper is an extension of a version presented in the Constructionism 2018 conference (Juškevičienė and Dagienė, 2018).

## Digital Competence

The invention of the computer and the internet has changed our lives and the education sector. It requires individuals to improve their competencies, especially to be digitally literate.

Digital competence is the most recent concept describing technology-related skills. Digital literacy is often seen as a synonym of digital competence. However, there are some stages in development of this concept, for example: computer skills → ICT skills → digital skills → digital competences (Ilomäki *et al.*, 2011; Laar *et al.*, 2017). Some researchers argue that the concept of digital skills, as a more holistic phenomenon, involves more features than digital competence (Pérez-Escoda, Rodríguez-Conde, 2015).

The amount of content (e. g. media, tools, technologies) in the internet is growing every day. Thus, it requires individuals to deal with more abilities. A JRC technical report (2012) analysed fifteen frameworks on DC and developed the following definition: “the set of knowledge, skills, attitudes (thus including abilities, strategies, values and awareness) that are required when using ICT and digital media to perform tasks; solve problems; communicate; manage information; collaborate; create and share content; and build knowledge effectively, efficiently, appropriately, critically, creatively, autonomously, flexibly, ethically, reflectively for work, leisure, participation, learning, socialising, consuming, and empowerment”. Being digitally competent implies particular abilities, such as, understanding media, searching for information and being critical about what is retrieved, and communication with others using a variety of digital tools. Seven areas of DC were identified:

- (1) information management,
- (2) collaboration,
- (3) communication and sharing,
- (4) content and knowledge creation,
- (5) ethics and responsibility,

- (6) evaluation and problem solving, and
- (7) technical operations.

The results of that report contributed to the DigCom project. Lately, contribution to a better understanding and development of DC in Europe was presented as a DC framework involving five competence areas (DIGCOMP, 2013):

- (1) information,
- (2) communication,
- (3) content creation,
- (4) safety, and
- (5) problem solving.

These areas consist of 21 competences. It was a meta-framework for existing frameworks, initiatives, curricula and certifications. DigComp 1.0 has become a reference for many DC initiatives at European and Member State levels. DigComp 2.0 keeps the same overall structure of 5 competence areas, however, slightly renamed (DigComp, 2016):

- (1) information and data literacy,
- (2) communication and collaboration,
- (3) digital content creation,
- (4) safety, and
- (5) problem solving.

One of the best known frameworks among educators and academics is the European Framework for the Digital Competence of Educators (DigCompEdu) which has six areas focusing on different aspects of educators' professional activities:

- (1) Professional Engagement – organisational communication, professional collaboration, reflective practice, digital continuous professional development.
- (2) Digital Resources – selecting, creating, modifying, managing, protecting and sharing digital resources.
- (3) Teaching and Learning – teaching, guidance, collaborative and self-regulated learning.
- (4) Assessment – strategies, evidence analysis, feedback and planning.
- (5) Empowering Learners – accessibility and inclusion, differentiation and personalisation, actively engaging learners.
- (6) Facilitating Learners' Digital Competence (DigCompEdu, 2017).

DC involves the confident, critical and responsible use of, and engagement with, digital technologies for learning, at work, and for participation in society. It includes information and data literacy, communication and collaboration, digital content creation (including programming), safety (including digital well-being and competences related to cybersecurity), and problem solving. The concept “digital technologies” is employed as an umbrella term for digital resources and devices, thus comprising any kind of digital input: software (including apps and games), hardware (e.g. classroom technologies or mobile devices) or digital content/data (i.e. any files, including images, audio and video).

## Computational thinking

Computational Thinking (CT) was presented by Jeannette Wing as a concept which involves solving problems, designing systems, and understanding human behaviour by drawing on the concepts fundamental to computer science (Wing, 2006). Before that, CT was mentioned by Seymour Papert in the context of an alternative, computationally-based mathematics education (Papert, 1996).

In the attempt to define CT, researchers often focus on the core components of CT. As argued in (Voogt *et al.*, 2015), it is more important to try to find similarities and relationships in the discussions about CT rather than to try to give an ultimate definition. Detailed analysis is made in the paper by (Kalelioglu *et al.*, 2016).

CT definitions were analysed and presented in percentage form of the words used to describe the essence of CT: problem solving (22%), abstraction (13%), computer (13%), process (9%), science (7%), data (7%), effective (6%), algorithm (6%), concepts (5%), ability (5%), tools (4%) and analysing (4%). However, some researchers concluded that current limitations in the CT definition are that it is shaped by technology-aided problem solving (Haseski *et al.*, 2018). Thus, further dimensions need to be explored, and especially personal, environmental, social, affective, psychological and ethical factors need to be investigated.

The focus on CT skills rather than the definition of CT was presented in (Curzon *et al.*, 2014). The authors used a simplified set of skills: algorithmic thinking, evaluation, decomposition, abstraction, and generalisation, and implemented them in the developed workshops for teachers on CT themes in order to fill teachers' knowledge gaps about CT as a useful practical way to teach computing to school students. Similar set of CT concepts was used to develop the relationship between CT concepts, student activity, and curriculum subjects example (Catlin and Woollard, 2014).

More CT elements were discussed in order to find out how the definition of CT can be interpreted. The literature review made by (Grover and Pea, 2013) provided nine widely accepted core elements (Table 1). International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) define CT as a problem solving process with some particular characteristics; additionally, they elaborate the vocabulary of CT for the purpose of explaining the operational definition by listing nine CT concepts implicit in the operational definition (CSTA and ISTE, 2011). An overview of the main sources for CT definitions and components is provided in Table 1.

CT is in line with many aspects of 21st century competencies such as creativity, critical thinking, and problem-solving (Lye and Koh, 2014). Thus, it can be seen as one of the constructionist methods (like the problem-based learning) which allows students to learn about a subject by exposing them to multiple problems and asking them to construct their understanding or objects of the subject through these problems. The pioneer of the constructivist theory, S. Papert, has expanded the theory of constructivism by stating that learning is best when the learner actively develops objects of the real world (for example, a sand castle, an automatic watering system), and not just ideas or knowledge that deliberately engages in design. The learning process itself is improved by improving the conditions that learners can construct.

Additionally, constructionism can be seen as the theory of particular relevance when considering lifelong learning. Lifelong learning is defined as “all learning activity undertaken throughout life, with the aim of improving knowledge, skills and competences within a personal, civic, social and/or employment-related perspective” (CEC, 2001). Similarly, the CT is nowadays fundamental and children should learn it for life because it involves essential digital age competencies.

Table 1  
CT definitions and main components

Reference	Definition	Main components/skills/characteristics
Papert, 1996	Computers can help to solve problems in ways that “forge ideas” and that allow people to better analyse and explain problems, solutions, and the connections between them	<ul style="list-style-type: none"> <li>• Forging ideas</li> <li>• Analysing</li> <li>• Explaining</li> </ul>
Wing, 2006	CT is a universally applicable attitude and skill set everyone, not just computer scientist, would be eager to learn and use	<ul style="list-style-type: none"> <li>• Abstraction</li> <li>• Analysis</li> <li>• Automation</li> </ul>
Wing, 2011	CT is the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be carried out by an information-processing agent	<ul style="list-style-type: none"> <li>• Thought processes</li> <li>• Apply new computational methods to their problems</li> <li>• Reformulate problems to be amenable to computational strategies</li> <li>• Discover new science through analysis of data</li> <li>• Ask new questions that were not thought of, but which Are easily addressed computationally</li> <li>• Explain problems and solutions in computational terms</li> </ul>
CSTA & ISTE, 2011	CT is an approach to solving a problem and critical thinking that empowers the integration of digital technologies with human ideas	<ul style="list-style-type: none"> <li>• Formulating problems in a way that enables us to use a computer and other tools to help solve them</li> <li>• Logically organizing and analysing data</li> <li>• Representing data through abstractions, such as models and simulations</li> <li>• Automating solutions through algorithmic thinking</li> <li>• Identifying, analysing, and implementing efficient and effective solutions</li> <li>• Generalizing and transferring this problem-solving process to a wide variety of problems</li> </ul>
Grover & Pea, 2013	CT as the thought processes involved in formulating problems so their solutions can be represented as computational steps and algorithms	<ul style="list-style-type: none"> <li>• Abstractions and pattern generalizations</li> <li>• Systematic processing of information</li> <li>• Symbol systems and representations</li> <li>• Algorithmic notions of flow of control</li> <li>• Structured problem decomposition (modularizing)</li> <li>• Iterative, recursive, and parallel thinking</li> <li>• Conditional logic</li> <li>• Efficiency and performance constraints</li> <li>• Debugging and systematic error detection</li> </ul>

Continued on next page

Table 1 – continued from previous page

Reference	Definition	Main components/skills/characteristics
CAS, 2015	CT is the process of recognising aspects of computation in the world that surrounds us and applying tools and techniques from computing to understand and reason about systems and processes	<ul style="list-style-type: none"> <li>• Logical reasoning</li> <li>• Abstraction</li> <li>• Decomposition</li> <li>• Algorithmic thinking</li> <li>• Evaluation</li> <li>• Generalisation</li> </ul>
Google for education	CT is a problem solving process that includes a number of characteristics, such as logically ordering and analysing data and creating solutions using a series of ordered steps (or algorithms), and dispositions, such as the ability to confidently deal with complexity and open-ended problems	<ul style="list-style-type: none"> <li>• Abstraction</li> <li>• Algorithm design</li> <li>• Automation</li> <li>• Data analysis</li> <li>• Data collection</li> <li>• Data representation</li> <li>• Decomposition</li> <li>• Parallelization</li> <li>• Pattern generalization</li> <li>• Pattern recognition</li> <li>• Simulation</li> </ul>
ISTE, 2016	Students develop and employ strategies for understanding and solving problems in ways that leverage the power of technological methods to develop and test solutions	<ul style="list-style-type: none"> <li>• Formulate problem definitions suited for technology-assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions</li> <li>• Collect data or identify relevant data sets, use digital tools to analyse them, and represent data in various ways to facilitate problem-solving and decision-making</li> <li>• Break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving</li> <li>• Understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions</li> </ul>
Weintrop <i>et al.</i> , 2016	CT is a taxonomy of practices focusing on the application of computing to mathematics and science: data practices, modelling and simulation practices, computational problem solving practices, and systems thinking practices	<ul style="list-style-type: none"> <li>• Ability to deal with open-ended problems</li> <li>• Persistence in working through challenging problems</li> <li>• Confidence in dealing with complexity</li> <li>• Representing ideas in computationally meaningful ways</li> <li>• Breaking down large problems into smaller problems</li> <li>• Creating abstractions for aspects of problem at hand</li> <li>• Reframing problem into a recognizable problem</li> <li>• Assessing strengths/weaknesses of a representation of data/representational systems</li> <li>• Generating algorithmic solutions</li> <li>• Recognizing and addressing ambiguity in algorithms</li> </ul>
Kalelioğlu <i>et al.</i> , 2016	CT is complex higher-order thinking that require to use the power of cognitive ability and embrace the support of machines to think and solve problems	<ul style="list-style-type: none"> <li>• Abstraction and decomposition</li> <li>• Data collection and analysis, pattern recognition, conceptualising and data representation</li> <li>• Mathematical reasoning, parallelisation building and algorithms, and procedures</li> <li>• Automation, modelling and simulations</li> <li>• Testing, debugging and generalization</li> </ul>
Krauss & Prottsman, 2017	Using thinking patterns and processes to pose and solve problems or prepare programs for computation	<ul style="list-style-type: none"> <li>• Decomposition (data analysis)</li> <li>• Pattern matching (data visualization)</li> <li>• Abstraction (data modelling, pattern generalization)</li> <li>• Automation (algorithm design, parallelization, simulation)</li> </ul>

## CT frameworks and existing practical implementation solutions

A framework for studying and assessing the development of CT was developed by (Brennan and Resnick, 2012). Three key dimensions based on studying the activity in the Scratch online community and in Scratch workshops were identified: computational concepts (the concepts designers engage with as they program, such as iteration, parallelism, etc.), computational practices (the practices designers develop as they engage with the concepts, such as debugging projects or remixing others' work) and computational perspectives (the perspectives designers form about the world around them and about themselves). Based on these three dimensions, the suggestions for how to assess learning programming were developed.

The College Board (2017) developed the CT framework for a Computer Science Principles course for high schools in the USA. They identify six CT practices (Table 2). It is believed that these practises help students make sense of knowledge in order to accomplish the task, and learn collaboration and communication principles.

Table 2  
Six CT practices based on (College Board, 2017)

Practices	Students are expected to:
1 <i>Connecting computing</i> : relates to the influence of computing and its implications on individuals and society	<ul style="list-style-type: none"> <li>• Identify impacts of computing</li> <li>• Describe connections between people and computing</li> <li>• Explain connections between computing concepts</li> </ul>
2 <i>Creating computational artefacts</i> : given the creative nature of computing, this practice allows students to engage in computing by designing and developing computational artefacts	<ul style="list-style-type: none"> <li>• Create an artefact with a practical, personal, or societal intent</li> <li>• Select appropriate techniques to develop a computational artefact</li> <li>• Use appropriate algorithmic and information-management principles</li> </ul>
3 <i>Abstracting</i> : focuses on students' understanding and applying abstraction to develop models and simulations of natural and artificial phenomena	<ul style="list-style-type: none"> <li>• Explain data, information, or knowledge that are represented for computational use</li> <li>• Explain abstractions used in computation or modelling;</li> <li>• Identify abstractions</li> <li>• Describe modelling in a computational context</li> </ul>
4 <i>Analysing problems and artefacts</i> : involves developing solutions, models, and artefacts for problems and evaluating the appropriateness of the solutions	<ul style="list-style-type: none"> <li>• Evaluate a proposed solution to a problem</li> <li>• Locate and correct errors</li> <li>• Explain how an artefact functions</li> <li>• Justify appropriateness and correctness</li> </ul>
5 <i>Communicating</i> : allows students to describe the influence of technology and computation supported by data visualizations and computational analysis	<ul style="list-style-type: none"> <li>• Explain the meaning of a result in context</li> <li>• Describe computation with accurate and precise language, notation, or visualizations</li> <li>• Summarize the purpose of a computational artefact</li> </ul>
6 <i>Collaborating</i> : peers learn to work together effectively to solve ill-structured problems that use computation	<ul style="list-style-type: none"> <li>• Collaborate in solving a computational problem</li> <li>• Collaborate in producing an artefact</li> <li>• Share the workload by providing individual contributions</li> <li>• Foster a constructive collaborative climate by resolving conflicts and facilitating the contributions of the team</li> <li>• Exchange knowledge and feedback</li> <li>• Review and revise the work done</li> </ul>

Computing at School (CAS) presented the conceptual framework of CT by identifying key concepts (logic, algorithms, decomposition, patterns, abstraction, evaluation) and approaches (tinkering, creating, debugging, persevering, collaborating) involved in CT process as well as techniques (reflecting, coding, designing, analysing, applying) employed to demonstrate and assess CT (CAS, 2015). Each of the concepts of CT can be identified with approaches and techniques. This gives the opportunity to implement CT in the classroom.

The framework for CT as a Problem-Solving Process was developed by (Kalelioğlu *et al.*, 2016). It has five main categories of processes that consist of the actions extracted from literature analysis. These categories are as follows: identify the problem; gathering, representing and analysing data; generate, select and plan solutions; implement solutions; assessing solutions and continue for improvement. They argue that this framework could help to teach, learn and practice CT and informatics concepts within many courses.

In the book by Krauss and Prottsman (2017), a CT framework with four categories: decomposition, pattern matching, abstraction, and automation, is presented. Each of these categories has subcategories. This framework could help to plan computer science lessons. The detailed descriptions of lessons plan for each category are given as well.

Scratch, App Inventor, LegoMind Stroms, CS Unplugged activities, e.g. *Kid Krypto* (Rosamond, 2018) and various games are widely available tools and resources for CT development. CS Unplugged activities are adopted in different ways: videos, shows, outdoor activities, competitions. Another adoption is an internationally recognized challenge on informatics and CT called *Bebras* (Beaver), which has recently had over 60 participating countries. The *Bebras* challenge is an informatics education community-building model designed to promote informatics learning and CT at schools by solving short informatics-concept based tasks (Dagienė and Stupurienė, 2016; Izu *et al.*, 2017). The initial goal of the *Bebras* project was to motivate students to be more interested in informatics topics with a strong intention to deepen algorithmic and operational thinking and extend to CT (Dagienė *et al.*, 2017).

## Interconnection between Digital Competence and Computational Thinking

The characteristics of CT and DC approaches based on results of literature review were analysed. The analysis showed that CT skills are overlapping and relatively broad in context. Eight CT component groups were identified. Additionally, the list of abilities are presented in Table 3 (in the last column).

The first CT concepts group – **Data analysis & representation** involves processes of data collection, analysis and representation. The concept of generalization is also included in this group. Generalization is a way of gaining extra information by finding similarities between items (Krauss, Prottsman, 2017). Thus, the concept of pattern recognition also belongs to this group. Generalization can be defined as an activity that identifies patterns among individual sub-problems and simplifies them (Beecher, 2017). This group is mostly related to (CSTA and ISTE, 2011) the vocabulary and framework for CT proposed by (Atmatzidou and Demetriadis, 2016).

The second group – **Computing Artefacts** involves the creative aspects of computing and is developed based on six CT practise frameworks. It consists of the process of designing and developing computational artefacts as well as applying computing techniques to creatively solve problems.

**Decomposition** (Decom) is the third group. This concept is identified in most of the literature on CT and simplified means the process of breaking down the task into smaller manageable parts.

The fourth group – **Abstraction** can be defined as the solution for a more general problem by ignoring certain details (Krauss, Prottzman, 2017). It also involves developing and representing models of the real world (Yadav and Hong and Stephenson, 2016).

Table 3  
CT component groups

CT components	Concepts involved	Abilities
<b>DatAnaRep:</b> Data analysis & representation	Data collection Data analysis Data representation Generalisation Patterns finding Drawing conclusions	<b>GRAD1</b> To gather appropriate information and making sense of data <b>GRAD2</b> To make sense of data, find patterns, and draw conclusions <b>GRAD3</b> To depict and organize data in appropriate graphs, charts, words, or images <b>GRAD4</b> To expand an existing solution for a given problem to cover more cases <b>GRAD5</b> To identify the patterns and commonality between artefacts, processes or systems <b>GRAD 6</b> To find similarities between items as a way of gaining extra information  <i>Based on:</i> (CSTA&ISTE, 2011; Atmatzidou et al, 2016; Catlin & Woollard, 2014; Krauss, Prottzman, 2017)
<b>ComA:</b> Computing Artefacts	Artefact development Artefact designing	<b>CCA1</b> Create an artefact (content) with a practical, personal, or societal intent <b>CCA2</b> Select appropriate techniques to develop a computational artefact (content) <b>CCA3</b> Use appropriate algorithmic and information-management principles  <i>Based on:</i> (College Board, 2017)
<b>Decom:</b> Decomposition	Breaking into parts	<b>D1</b> To break down a problem (data, processes) into parts that are easier to manage or into constituent parts to make them easier to work with  <i>Based on:</i> (Atmatzidou & Demetriadis, 2016; CAS, 2015; Google 201)
<b>Abst:</b> Abstraction	Details suppression Modelling Information filtering	<b>A1</b> Explain data, information, or knowledge represented for computational use <b>A2</b> Explain abstractions used in computation or modelling; <b>A3</b> Identify abstractions <b>A4</b> Describe modelling in a computational context <b>A5</b> Filtering information when developing solutions  <i>Based on:</i> (College Board, 2017; CAS, 2015)

Continued on next page

Table 3 – continued from previous page

CT components	Concepts involved	Abilities
<b>Algo:</b> Algorithms	Sequence of steps Procedures Set of instructions Automation	<b>AT1</b> Identify the processes and sequence of events <b>AT2</b> Plug pieces into an algorithm to help with a result <b>AT3</b> Control a process by automatic means, reducing human intervention to a minimum  <i>Based on:</i> (Catlin, Woollard, 2014; Krauss & Prottzman, 2017)
<b>ComCol:</b> Communication & collaboration	Communication Collaboration Computational analysis	<b>CC1</b> Explain the meaning of a result in context <b>CC2</b> Describe computation with accurate and precise language, notations, or visualizations <b>CC3</b> Summarize the purpose of a computational artefact <b>CC4</b> Collaborate with another student in solving a computational problem, producing an artefact (content) <b>CC5</b> Share the workload by providing individual contributions to overall collaborative effort <b>CC6</b> Foster a constructive collaborative climate by resolving conflicts and facilitating the contributions <b>CC7</b> Exchange knowledge and feedback <b>CC8</b> Review and revise their work as needed to create a high-quality artefact  <i>Based on:</i> (College Board, 2017; CAS, 2015)
<b>ComSoc:</b> Computing & Society	Computing influence Computing implication Computing concepts	<b>CnC1</b> Identify impacts of computing, describe connections between people and computing <b>CnC2</b> Explain connections between computing concepts  <i>Based on:</i> (College Board, 2017; CAS, 2015)
<b>Eval:</b> Evaluation	Evaluation Correction	<b>E1</b> Evaluate a proposed solution to a problem <b>E2</b> Locate and correct errors <b>E3</b> Explain how an artefact (content) functions <b>E4</b> Justify appropriateness and correctness  <i>Based on:</i> (College Board, 2017; CAS, 2015)

The core concept of CT is **Algorithms**, and this concept is mainly devoted to the process of algorithm design. It also involves the automation concept as it can be defined as the process of plugging pieces into an algorithm to help obtaining a result (Krauss, Prottzman, 2017). Similarly, these two concepts were grouped together in (Duncan, Bell and Atlas, 2017).

The sixth group is **Communication & collaboration**, the ability to communicate and work with others to achieve a common goal or solution, which are essential dimensions of CT. In order to successfully design, build, and improve computational artefacts, the application of teamwork and collaboration, based on effective team practices, is important (College Board, 2017).

The practice that relates to the influence of computing and its implications on individuals and society is called **Computing & Society**. It can be seen also as responsible use of technologies by understanding the impact of computing and the connection between society and computing. It involves concepts such as cybersecurity concerns, privacy, self-protection in the Internet, and potential beneficial and harmful effects of computing innovation.

The **Evaluation** group involves the process of ensuring that a solution, whether an algorithm, system, or process, fit for the purpose (CAS, 2015). Simplified: evaluating the

appropriateness of the proposed solutions and artefacts (College Board, 2017). It can be done systematically (through criteria and heuristics) to make substantiated value judgements (Catlin and Woollard, 2014).

Digital competencies selected from the European Framework for the DC of Educators (DigCompEdu, 2017) were analysed. Facilitating Learners' Digital Competence area is presented in detail in Table 4.

Table 4  
DC components and abilities (based on: (DigCompEdu, 2017))

Components	Concepts involved	Abilities
<b>InfMedLit:</b> Information & media literacy	Information search Information analyses Critical evaluation Information procession	<b>IML1</b> To articulate information needs, to search for data, information and content, to access and navigate <b>IML2</b> To create and update personal search strategies <b>IML3</b> To adapt search strategies based on the quality of information found <b>IML4</b> To analyse, compare and critically evaluate the credibility and reliability of sources of data, information and digital content <b>IML5</b> To organise, store and retrieve data, information and content in digital environments <b>IML6</b> To process information in a structured environment
<b>DigComCol:</b> Digital communication & collaboration	Digital interaction Digital communication strategies Data exchange Self-empowerment Digital collaboration Behavioural norms Digital protection Civic participation	<b>DCC1</b> To interact through a variety of digital technologies <b>DCC2</b> To understand appropriate digital communication means for a given context <b>DCC3</b> To share data, information and digital content with others through appropriate digital technologies <b>DCC4</b> To know about referencing and attribution practices <b>DCC5</b> To participate in society through the use of public and private digital services <b>DCC6</b> To seek opportunities for self-empowerment and for participatory citizenship through digital technologies <b>DCC7</b> To use digital technologies for collaboration, and for co-construction of resources and knowledge <b>DCC8</b> To be aware of behavioural norms and know-how while interacting in digital environments <b>DCC9</b> To adapt communication strategies to the specific audience and to be aware of cultural diversity in digital environments <b>DCC10</b> To create and manage digital identities <b>DCC11</b> To protect one's own reputation <b>DCC12</b> To deal with the data that one produces through several digital technologies, environments, and services
<b>DigCon:</b> Digital content	Digital content development Copyright and licenses Reference sources Digital identity	<b>DCr1</b> To create and edit digital content in different formats <b>DCr2</b> To express themselves through digital means <b>DCr3</b> To modify, refine, improve and integrate information and content into an existing body of knowledge <b>DCr4</b> To create relevant content and knowledge <b>DCr5</b> To understand how copyright and licenses apply to data, information, and digital content <b>DCr6</b> To plan and develop a sequence of understandable instructions for a computing system to solve a given problem or perform a specific task

Continued on next page

Table 4 – continued from previous page

Components	Concepts involved	Abilities
<b>ResUse:</b> Responsible use	Physical wellbeing	<b>RU1</b> To protect devices and digital content, and to understand risks and threats in digital environments
	Psychological wellbeing	<b>RU2</b> To understand safety and security measures
	Social wellbeing	<b>RU3</b> To protect personal data and privacy in digital environments
	Digital safety	<b>RU4</b> To understand how to use and share personal information while being able to protect oneself and others from damages
	Digital threats	<b>RU5</b> To understand that digital services use a “Privacy policy” on how personal data is used
	Digital responsibility	<b>RU6</b> To avoid health risks and threats to physical and psychological well-being while using digital technologies
		<b>RU7</b> To protect oneself and others from possible dangers in digital environments, e.g. cyberbullying
		<b>RU8</b> To be aware of digital technologies for social wellbeing and social inclusion
		<b>RU9</b> To be aware of the environmental impact of digital technologies and their use
		<b>RU10</b> To monitor student behaviour in digital environments in order to safeguard their wellbeing
		<b>RU11</b> To react immediately and effectively when learners’ wellbeing is threatened in digital environments
<b>DigProSol:</b> Digital problem solving	Problem solving	<b>DPS1</b> To identify technical problems when operating devices and using digital environments, and to solve them
	Digital competence development	<b>DPS2</b> To adjust digital environments to personal needs
	Evaluation	<b>DPS3</b> To identify, evaluate, select, and use digital technologies to solve a given task or problem
	Self-development	<b>DPS4</b> To use digital technologies in innovative ways to create knowledge
	Personal needs	<b>DPS5</b> To understand where their digital competence needs to be improved or updated
		<b>DPS6</b> To support others in their digital competence development
		<b>DPS7</b> To seek opportunities for self-development and to keep up-to-date with the digital evolution

Visualization of the interconnection between CT and DC component groups is presented in Fig. 1. These interconnections were developed from a matching of relationships between the details of components and features of CT and DC gathered from their respective literature reviews. First, component groups were identified and definitions were collected for CT and DC. Then, the related abilities of each component group were listed (see Tables 3 and 4). Based on these definitions and abilities, main concepts involved in each group were identified (shown in the second column of each table). The main concepts and abilities in each one were used to identify a match, as shown in Fig. 2.

The interconnections presented in Fig. 2 are a visual portrayal of matches identified from analysis of the interconnections between DC and CT components. It was quite clear that the Information and media literacy group that involves abilities to manage information has interconnections with CT groups that involve similar abilities: Data

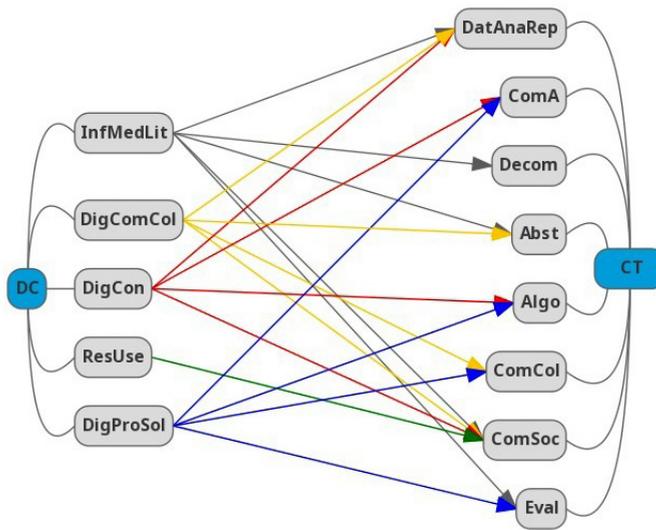


Fig. 2. The interconnection between DC and CT components.

analysis and representation, Decomposition, Abstraction, and Evaluation. It is interesting to note that it has interconnection with the Computing and Society group that relates to the influence of computing and its implications on individuals and society (College Board, 2017). As expected, the second group of Digital communication and collaboration which requires learners to effectively and responsibly use digital technologies for communication, collaboration and civic participation has interconnection with CT groups related to information exchange: Data analysis and representation, Communication and collaboration and, Computing and Society. Additionally, it connects to Abstraction group that is a way of expressing an idea in a specific context while at the same time suppressing details irrelevant in that context (Beecher, 2017), and it looks like it has nothing in common to communication or collaboration at a first sight. The Digital content group involves abilities to deal with content development strategies and licenses. It is not a surprise that it is interconnected with four CT groups related to content procession, development strategies and social wellbeing: Data analysis and representation, Computing artefacts, Algorithms, and Computing and Society. The fourth DC group, Responsible use, involves abilities to empower learners to manage risks and use digital technologies safely and responsibly, and it has relation to a very similar CT group - Computing and Society. Finally, the last Digital problem solving group is interconnected with four CT groups of Computing artefacts, Algorithm, Communication and collaboration, and Evaluation involve abilities that are essential for problem solving. Thus, it is quite clear that the last, Digital problem solving group, is interconnected with these four CT groups.

The presented interconnections of DC and CT groups shown in Fig. 3 were identified from similarities between abilities in each group. Patterns were spotted by looking for concrete descriptions, nouns and verbs that appeared in both cases. This enabled a sim-

plification of them in order to identify interconnections. The findings from the literature review, generalisation and analyses are described below.

Information & media literacy group has 6 abilities listed as IML1-IML6 (Fig. 3.). The first Information & media literacy ability is information searching. Thus, the ability to gather information is very similar as Abstraction ability to information filtering in solution development, and in order to articulate information needs, the Computing and society ability of understanding connections between computing is needed. The second Information & media literacy ability is the development of search strategies, and it has relation with the ability to gather appropriate information and making sense of data. Additionally, in order to update the search strategy, the expansion of an existing solution to cover more cases could be used. Information filtering is needed for updating personal search strategies as well as in the ability to explain connections between computing concepts. The third Information & media literacy ability is to adapt search strategies, and it has the same interconnections with DC abilities as the second Information & media literacy ability. The fourth Information & media literacy ability is the evaluation of content. Thus, the analysis and comparison of content is related to pattern finding and conclusion making, as well as to the evolution of a proposed solution and its appropriateness, including justification of its correctness. To organize content in the digital environment

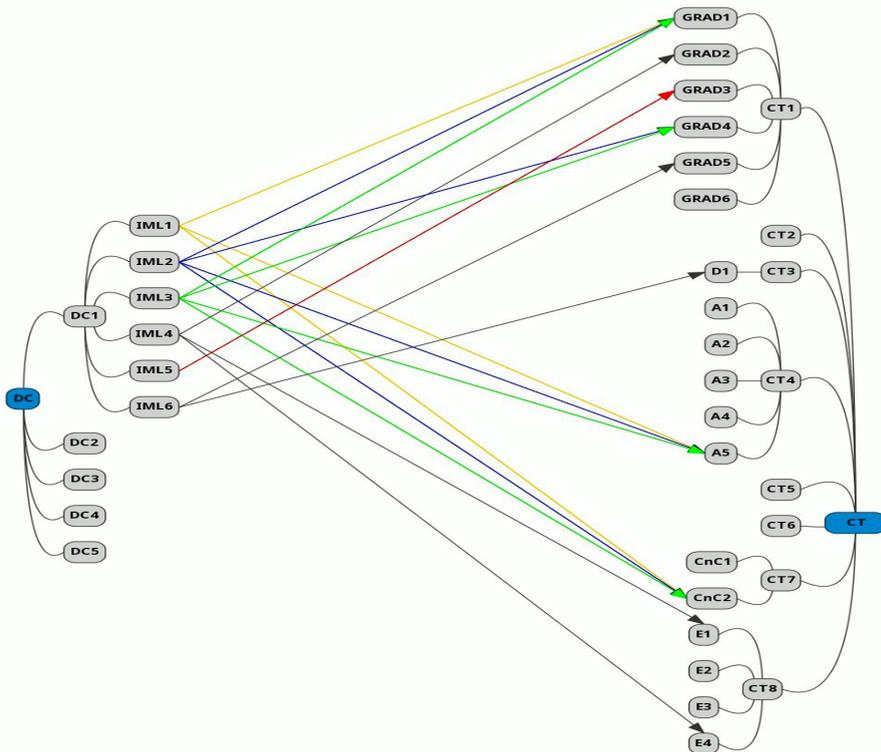


Fig. 3. The interconnection between Information & media literacy – DC1 and CT.

(sixth Information & media literacy ability) is directly related to the ability of depicting and organizing data in graphs, charts, images. The last listed Information & media literacy ability – information processing and organizing in structured environment – has interconnection with two CT abilities: to identify the patterns and commonality between content could be one of the steps for information organizing as well as breaking down the content into smaller parts to easier manage them.

The second DC group, Digital communication & collaboration, has twelve abilities: DCC1–DCC12, see Fig. 4(a). Eleven of them are related to two CT groups, mostly to Communication & collaboration. Of course, the ability to interact and the ability to share content has connections to the ability to exchange knowledge. The ability to understand appropriate digital communication means for a given context could be related to the ability to explain the meaning of a result in context. It is clear that the ability to know about referencing and attribution practices, the ability to participate in society through the use of digital services, and the ability to create and manage one or multiple digital identities, the ability to protect one’s own reputation) are connected to Computing & Society group ability to identify impacts of computing and describe connections between people and computing. This ability deals with privacy and security concerns, self-protection in Internet, the way people connect, computing innovation (social, economic, cultural) effect,

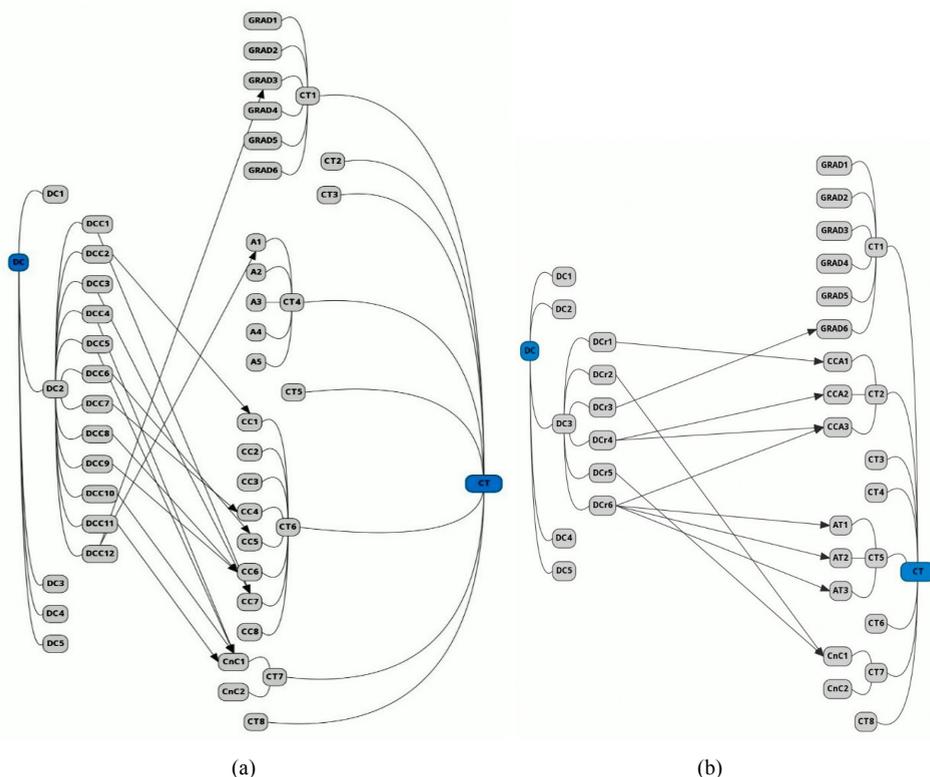


Fig. 4. The interconnection between (a) Digital communication & collaboration – DC2 and CT abilities, and (b) Digital content – DC3 and CT abilities.

beneficial and harmful effects of computing (College Board, 2017). The sixth Digital communication & collaboration practice to seek opportunities for self-empowerment and participatory citizenship could be related to practice of sharing the workload to collaborative effort by providing individual contributions. The seventh Digital communication & collaboration ability to collaboratively create content is interconnected with the ability to collaborate in producing content and problem solving.

The ability to foster a constructive collaborative climate by resolving conflicts and facilitating the contributions is related to the eighth and the ninth Digital communication & collaboration abilities, because in order to foster the constructive collaborative climate, one must be aware of behavioural norms and cultural diversity. The ability to deal with the data that one produces through several digital technologies, environments, and services has connection to CT Data analysis & representation group's ability to deal with data in appropriate formats and Abstraction group's ability to explain how the content is represented for computation.

Fig. 4(b) presents the third group of Digital content abilities: DCr1–DCr6. The first ability is to manage digital content in different formats. It is related to the ability to create content with a personal intent. The ability to express themselves through digital means and to be aware of copyright and license application is connected to the ability CnC1 to describe the connection between people and computing. The third ability is partly related to Data analysis & representation group's ability, because in order to modify, refine, improve and integrate information into an existing body of knowledge, one could find similarities between items as a way of gaining extra information. On purpose to create original and relevant content; and knowledge (fourth ability), the abilities to select appropriate techniques for information-management principles are useful. The sixth ability to perform a task or to solve a problem by developing a sequence of instructions for computing system has four interconnections with abilities to use appropriate algorithmic principles, to identify sequences of events, to plug pieces into an algorithm and to control a process by automatic means.

Fig. 5(a) shows that all the abilities of the Responsible use group, which deal with safety and security understanding, protection and privacy, and wellbeing in digital environment are interconnected with the ability to identify the impacts of computing, and to describe connections between people and computing. Part (b) of Fig. 5 shows how the abilities of the seventh Digital problem solving group are interconnected with nine CT abilities. The first ability of identifying and solving problems when using devices connects to the ability of identifying the processes of events. For the purpose to identify, select, use, and evaluate possible technological responses to solve a task, all abilities of the Evolution group are needed: evaluate proposed solution, correct errors, explain solution functions and justify appropriateness and correctness. The fourth ability to create knowledge by using technologies in innovation can be seen as a use of technologies with practical, personal or societal intent and appropriate management principles. Additionally, in order to create a high quality content, the review and revision of one's own work is needed. Also, it is related to the competences of self-improvement needs recognition and self-development regulation. The sixth ability to support others in competence development has connection to the ability to facilitate the contribution of a team member in a constructive way, thus, enabling their improvement.

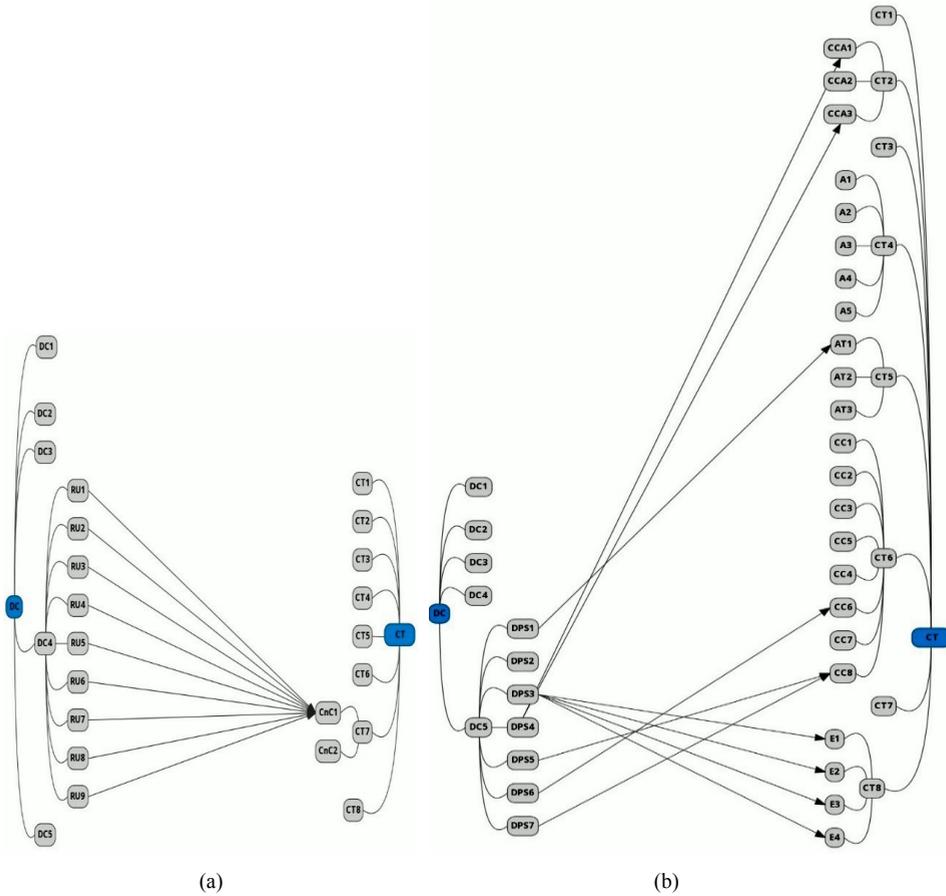


Fig. 5. The interconnection between (a) Responsible use – DC4 and CT abilities, and (b) Digital problem solving – DC5 and CT abilities.

### Conclusion and Discussion

The analysis of possible interconnections shows that both Digital Competencies (DC) and Computational Thinking (CT) have a lot of in common. Many abilities and competencies are overlapping. Only one of the listed abilities of DC (Digital problem solving – to adjust and customise digital environments to personal needs) has no direct connections to CT abilities. This ability originates from the area of personalization research, and, thus, it could be concluded that it is currently out of the focus area of CT. Digital problem solving is a very important and large area, and more detailed investigations of its connection to CT are needed.

Five CT abilities were left without connections: ability to explain how abstractions are used in computation or modelling; ability to identify abstractions; ability to describe modelling in a computational context; ability to describe computation with accurate and precise language, notations, or visualizations; and the ability to summarize the purpose

of a computational artefact. These abilities are related to abstraction and computation processes, and, actually, they are included in the DC framework indirectly.

The discussed framework of CT component groups and the interconnections between DC and CT could be further developed, and especially quality-research based on interviewing experts is needed. An evaluation study of experts' opinions of different countries for the proposed framework is in our future plans.

A limitation of this work is that there was no focus on age groups, educational course subject or gender. As argued in the work by (Shailaja and Sridaran, 2015), CT skills could be grouped by age, e.g. visualization, pattern recognition and generalization can be learnt in K-2, abstraction and critical thinking in grade 6 to 8. Research made by (Rijke *et al.*, 2018) showed that after the age of 9.5 years, female students begin to outperform their male peers on the abstraction tasks. However, were not found any differences in performance on the decomposition task between age groups, either between males and females. Additionally, deeper efforts should be invested to integrate CT in different subjects. CT can benefit students studying in any area, and skills for academic and work lives can be successfully taught in any subjects (Lockwood and Mooney, 2017). Similar questions can be applied to DC.

## Acknowledgment

This research is/was funded by the European Social Fund under the No 09.3.3-LMT-K-712 “Development of Competences of Scientists, other Researchers and Students through Practical Research Activities” measure. The authors also gratefully acknowledge the support of the Nordic Research Council through the NordPLUS programme of transverse actions. In particular, the funding of the two-year project “Culturally Diverse Approaches to Learning Mathematics and Computational Thinking”, with project code NPHZ-2018/10063, is acknowledged.

## References

- Aho, A.V. (2012). Computation and computational thinking. *Computer Journal*, 55, 832–835.
- Atmatzidou, S., Demetriadis, S. (2016) Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, 75, 661–670.
- Beecher, K. (2017). *Computational Thinking: A Beginner's Guide to Problem-Solving and Programming*.
- 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, 1–25.
- Catlin, D., Woollard, J. (2014). Educational robots and computational thinking. In: *Proceedings of 4th International Workshop Teaching Robotics, Teaching with Robotics & 5th International Conference Robotics in Education*, 144–151.
- College Board (2017). *AP Computer Science Principles. Course and Exam Description*. College Board, NY.
- Commission of the European Communities (CEC) (2001). Making a European Area of Lifelong Learning a Reality. EUR-Lex.
- Curzon, P., McOwan, P.W. (2017). *The Power of Computational Thinking: Games, Magic and Puzzles to Help You Become a Computational Thinker*. World Scientific.

- Curzon, P., McOwan, P.W., Plant, N., Meagher, L.R. (2014). Introducing teachers to computational thinking using unplugged storytelling. In: *ACM Proceedings of the 9th workshop in primary and secondary computing education*, 89–92.
- Dagienė, V., Stupuriene, G. (2016). Bebras – a sustainable community building model for the concept based learning of informatics and computational thinking. *Informatics in Education*, 5(1), 25–44.
- Dagienė, V., Sentance, S., Stupuriene, G. (2017). Developing a Two-Dimensional Categorization System for Educational Tasks in Informatics. *Informatica*, 28(1), 23–44.
- Duncan, C., Bell, T., Atlas, J. (2017). What do the teachers think?: Introducing computational thinking in the primary school curriculum. In: *ACM Proceedings of the Nineteenth Australasian Computing Education Conference*, 65–74.
- Ferrari, A. (2013) DIGCOMP: *A Framework for Developing and Understanding Digital Competence in Europe*.
- Ferrari, A. (2012) *Digital Competence in Practice: An Analysis of Frameworks*. JRC technical report. Google for Education. *Exploring computational thinking*. Retrieved from <https://edu.google.com/resources/programs/exploring-computational-thinking/#!home>
- Grover, S., Pea, R. (2013) Computational thinking in K–12: A review of the state of the field. *Educational Researcher*, 42(1), 38–43.
- Haseski, H.I., Ilic, U., Tugtekin, U. (2018) Defining a New 21st Century Skill-Computational Thinking: Concepts and Trends. *International Education Studies*, 11(4), 29.
- Ilomäki, L., Kantosalo, A., Lakkala, M. (2011). *What is Digital Competence?* In portal: Brussels: European Schoolnet.
- ISTE. (2016). *ISTE (International Society for Technology in Education) Standards for Students*. Eugene, OR.
- ISTE, CSTA. (2011). *Computational Thinking in K–12 Education Leadership Toolkit*.
- Izu, C., Miroló, C., Settle, A., Manilla, L., Stupuriene, G. (2017). Exploring Bebras Tasks Content and Performance: A Multinational Study. *Informatics in Education*, 16(1), 39–59.
- Juškevičienė, A., Dagienė, V. (2018). Interconnection between computational thinking and digital competence. In: Dagienė, V., Jasutė, E. (Eds.), *Constructionism 2018*. Vilnius, Lithuania, 305–314.
- Kalelioglu, F., Gülbahar, Y., Kukul, V. (2016). A framework for computational thinking based on a systematic research review. *Baltic Journal of Modern Computing*, 4(3), 583.
- Krauss, J., Prottsman, K. (2017). *Computational Thinking and Coding for Every Student*. The Teacher's Getting-Started Guide. Corwin Press Inc.
- Lockwood, J., Mooney, A. (2017). *Computational Thinking in Education: Where does it fit?* A systematic literary review. arXiv preprint arXiv:1703.07659.
- Lye, S.Y., Koh, J.H.L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12?. *Computers in Human Behavior*, 41, 51–61.
- Papert, S., (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning*, 1(1), 95–123.
- Pérez-Escoda, A., Rodríguez-Conde, M.J. (2015). Digital literacy and digital competences in the educational evaluation: USA and IEA contexts. In: *ACM Proceedings of the 3rd International Conference on Technological Ecosystems for Enhancing Multiculturality*, 355–360.
- Redecker, C. (2017). *European Framework for the Digital Competence of Educators: DigCompEdu* (No. JRC107466). Joint Research Centre (Seville site).
- Rijke, W.J., Bollen, L., Eysink, T.H., Tolboom, J.L. (2018). Computational Thinking in Primary School: An Examination of Abstraction and Decomposition in Different Age Groups. *Informatics in Education*, 17(1).
- Rosamond, F. (2018). Computational Thinking Enrichment: Public-Key Cryptography. *Informatics in Education*, 17(1).
- Royal Society. (2012). *Shut down or restart: The way forward for computing in UK schools*. Retrieved from <http://royalsociety.org/education/policy/computing-in-schools/report/>
- Shailaja, J., Sridaran, R. (2015). Computational Thinking the Intellectual Thinking for the 21st century. *International Journal of Advanced Networking & Applications*, May 2015 Special Issue, 39–46.
- Van Laar, E., van Deursen, A.J., van Dijk, J.A., de Haan, J. (2017). The relation between 21st-century skills and digital skills: A systematic literature review. *Computers in human behavior*, 72, 577–588.
- Voogt, J., Fisser, P., Good, J., Mishra, P., Yadav, A. (2015). Computational thinking in compulsory education: Towards an agenda for research and practice. *Education and Information Technologies*, 20(4), 715–728.
- Vuorikari, R., Punie, Y., Gomez, S.C., Van Den Brande, G. (2016). DigComp 2.0: The Digital Competence Framework for Citizens. Update Phase 1: The Conceptual Reference Model (No. JRC101254). Joint Research Centre (Seville site).
- Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., Wilensky, U. (2016). Defining compu-

- tational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, 25(1), 127–147.
- Wing, J.M. (2006). Computational Thinking. *Communications of the ACM*, 49(3), 33–35.
- Wing, J. (2011). *Research Notebook: Computational Thinking – What and Why?* The Link Magazine, Spring. Carnegie Mellon University, Pittsburgh.
- Yadav, A., Hong, H., Stephenson, C. (2016). Computational thinking for all: pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends*, 60(6), 565–568.

**A. Juškevičienė** is a doctor of technological sciences (informatics engineering). She is the researcher at the Vilnius University Institute of Data Science and Digital Technologies. The areas of her scientific interest focus on technology enhanced learning, intelligent and adaptive systems, recommender systems, semantics and ontology, evaluation of quality of learning software and learning process. She has been working very active on several national projects, helps to organize seminars and conferences. She has published a number of scientific papers and publications in popular magazines, participated in a number of large scale EU-funded R&D projects.

**V. Dagienė** (Professor, Dr.) is head of a department at Vilnius University Institute of Data Science and Digital Technologies. She has published over 200 scientific papers, and more than 50 textbooks in the field of informatics for schools. She is actively involved in various national and international committees as well as work groups on informatics education research. V. Dagienė is editor of two international journals: “Informatics in Education” and “Olympiads in Informatics”. In 2004 she founded International Challenge on Informatics and Computational Thinking *Bebras* which runs annually and involves more than 60 countries.