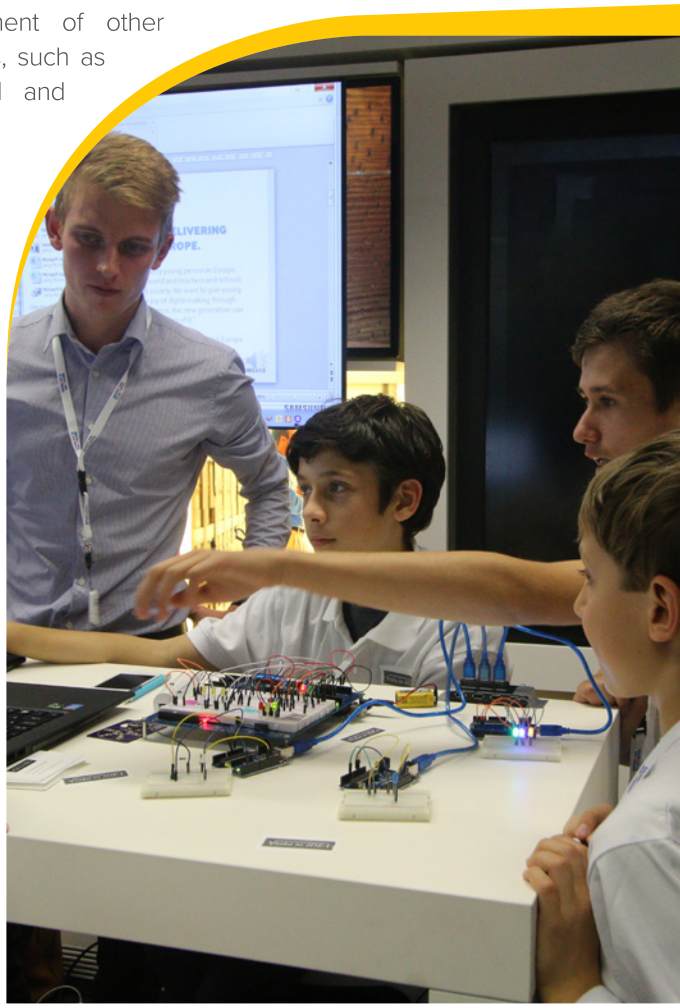
European Schoolnet Perspective
From research to policy action

ISSUE N°2 - APRIL 2017

The integration of Computational Thinking (CT) across school curricula in Europe

During the last 5 years, Computational Thinking (CT) has become one of the new 21st century skills needed for students to succeed in our digital society. Increasingly more countries are integrating coding, programming or CT in their curricula. According to our research, at least 20 countries have done so, namely: AT, BG, CZ, CH, DK, EE, ES, FI, FR, HU, HR, IT, IE, LT, MT, PL, PT, SK, UK (England), UK (Scotland). The main two reasons for this inclusion are:

- to support the development of other transversal 21st century skills, such as problem solving, analytical and logical thinking;
- to respond to the needs of the labour market which is more and more based on computer science skills across a variety of sectors.



The 2017 study "An analysis of educational approaches to developing Computational Thinking" aims to provide a comprehensive overview of recent research findings as well as grassroot and policy initiatives for developing CT as a 21st century skill among primary and secondary students. The study, funded by the Joint Research Centre of the European Commission, was carried out by European Schoolnet and the Institute for Educational Technology of the Italian National Research Council (ITD-CNR).

European Schoolnet started investigating these trends in 2014 in the report Computing our future, an updated version of which was published in the 2015 report.

This perspective paper provides some insights from the above-mentioned reports to inform advisable policy actions.

BACKGROUND INFORMATION

Nowadays, students need to know how to use technology, while at the same time understanding the underlying principles of digital devices and computing. Coding, programming, and CT respond to the need to understand the world around us by using computational methods to develop problem-solving strategies.

One of European Schoolnet's key aims is to **broker educational research findings and other evidence to our key stakeholders**: ministries of education, schools, teachers, and industry partners. The ***European Schoolnet Perspectives*** series is one way through which we achieve this.

Three issues are published annually. Each issue aims to:

- **Summarize research evidence** from key studies on innovation in education
- Translate this evidence into **concrete ideas for policy action**
- Conclude with the implications of the evidence for **using technology in teaching and learning**

All issues can be accessed online: **www.eun.org/observatory/perspectives**

Comments and suggestions on ***European Schoolnet Perspective*** are welcome: **info@eun.org**.

European Schoolnet is the network of 30 European Ministries of Education, based in Brussels. As a not-for-profit organisation, we aim to bring innovation in teaching and learning to our key stakeholders: Ministries of Education, schools, teachers, researchers, and industry partners.

www.europeanschoolnet.org
@eu_schoolnet
european.schoolnet

FINDINGS

What does Computational Thinking mean?

Evidence from the analysis

Computational Thinking refers to the key ideas and concepts of Computer Science. Jeanette Wing defines it as follows: “Computational thinking is the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can effectively be carried out by an information-processing agent.” Jeanette Wing’s landmark definition provides two valuable perspectives: (1) CT is a thought process, thus independent of technology; (2) CT is a specific type of problem solving that entails the reformulation and decomposition of a problem.

Despite the wide adoption of Wing’s definition, CT is understood or translated in many different ways. For instance, in Finland the term in use is ‘algorithmic thinking’. Very often the term is closely related to coding, while some scholars emphasise the close link between CT and logical thinking.

A set of core concepts and skills that characterise CT emerges from the literature. These include: abstraction, algorithmic thinking, automation, decomposition, debugging, and generalization. Coding/programming is a constituent of CT. However, there is consensus that CT actually entails much more than coding/programming, including for example processes of problem analysis and problem decomposition.

BACKGROUND INFORMATION

The pioneering decision of England to mandate computing in primary and secondary schools from September 2014 onwards created a momentum leading to curricula reforms in different parts of Europe, with the consequent recognition of the relevance of CT and/or programming. The Computing programme of study from England is an inspiring document that triggered changes in practice, as witnessed by the upsurge of teacher training provision in England. The programme of study claims: “A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world.” The main CT concepts that the curriculum promotes are abstraction, algorithms, logical reasoning, programming languages, data representation, the analysis of problems in computational terms, and debugging.

Ideas for policy action

- Define at national level, and in collaboration with curriculum designers and educators, which kind of knowledge, skills and attitudes the integration of CT in the curriculum should foster, and how they should be taught and assessed.
- Compile a shared terminology across countries to facilitate the process of curriculum integration, while at the same time respecting teachers’ freedom to introduce CT in a way that is suited to their specific context.
- Exchange regularly on approaches to integrating CT in the curriculum in different countries and the lessons learnt.

Transfer of skills to other domains

Evidence from the analysis

The transfer of cognitive skills is key to the claims for introducing CT in compulsory curricula. There is still a limited number of research works that focus on how CT concepts and constructs are transferred to other domains. The available evidence from a few single studies looking into CT related concepts and the possible transfer of skills suggests that:

- Learning programming does not automatically imply the development of problem solving skills. Programming languages are just tools and do not assure that curricula and teachers will focus explicitly on generalizable problem solving skills.
- Students who learn coding at early stages can improve their understanding of mathematical processes.
- There is no sufficient evidence yet concerning the long-term impact of studying coding/programming or CT in schools, or showing a direct relationship between having followed these activities in or outside schools and the take-up of related STEM studies.

BACKGROUND INFORMATION

A study testing first year students of informatics clearly shows that students start their studies in informatics with underdeveloped algorithmic skills, only a very few of them reaching the level of extended abstract thinking. It was found that students almost exclusively consider traditional programming environments appropriate for developing CT skills. Furthermore, they do not apply concept and algorithmic based methods in non-traditional computer related activities.

Ideas for policy action

- Consider commissioning further qualitative research to investigate the type of skills students acquire when exposed to a coding, programing or CT curriculum and which pedagogical approaches foster the acquisition of these skills.
- Gain a more comprehensive picture of CT skills based on additional assessments, such as a test of students' knowledge transfer.
- Support cooperation projects between universities and schools to bridge the current gap between skills acquired at school and demands by universities related to STEM studies. Such projects could also include parents, as they often guide their children in their career choice. Many parents are not familiar with new career opportunities related to STEM subjects.

Assessment

Evidence from the analysis

The importance of assessment for full and effective integration of CT in education is clearly highlighted in the literature. In most cases, the strategy adopted for CT assessment is to analyse the artefacts (e.g. games or models) that students develop as indications of their CT abilities. Other strategies include multiple-choice tests, attendant rubrics to assess CT skills, or getting students to modify the code of an existing program so as to accomplish specific objectives. Design-based approaches, such as programming interactive media, are also emerging as key elements of assessment systems. However, few signs are emerging of new, comprehensive approaches that encompass the complexity of the cognitive processes involved in CT.

Ideas for policy action

- Continue to investigate, develop and pilot new assessment methods.
- Consider case studies focusing on concrete implementation (pedagogy, tools, assessment).
- Define new tools and criteria to help teachers access CT skills, in particular as part of a cross-curricular approach.

Teacher training

Evidence from the analysis

The introduction of CT in compulsory education creates a demand for large-scale in-service continuous professional development, as many teachers did not learn about CT in their initial training. Moreover, teaching CT may require new pedagogical approaches that place students at the centre of the learning process.

Ideas for policy action

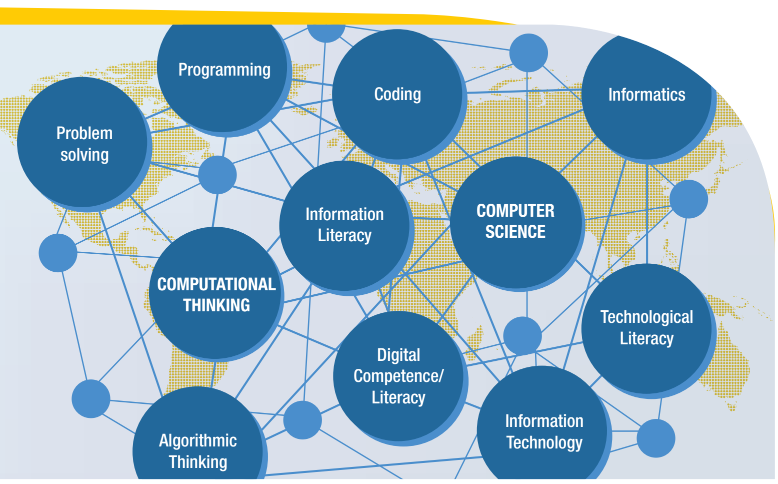
- Consider providing training opportunities that have a strong focus on pedagogy and that involve hands-on learning that is easy to transfer to the classroom.
- Devote attention to fostering peer exchange and community building that enables the sharing of good practice among teachers.
- Provide additional support to teachers by offering workshops, MOOCs or other online courses. Consider in particular blended approaches combining online and face-to-face elements.

IMPLICATIONS FOR TECHNOLOGY USE IN TEACHING AND LEARNING

CT emerged as an extremely interesting new concept to help prepare children for future challenges in an increasingly digital society. However, its real impact on children's learning and skills is still not clear. If CT is to earn a place in the curriculum over the coming decades, results in the coming years will need to demonstrate that teaching CT has an actual impact on children's learning and skills. Rigorous research on specific aspects such as assessment methods and transfer of knowledge will be key to the successful implementation of CT in formal education. Finally, as more tangible results on implementation and pedagogical choices become available in many countries, the exchange of experience and lessons learned at both European and international levels will become crucial.

More generally research, examples of curriculum integration and expert opinions show:

- Choosing the right programming language adapted to the age of children, level of difficulty and learning outcomes to be achieved is crucial. Teachers need to be supported in choosing the right tools. Research also suggests that students should be exposed to more than one programming language.
- It is not always necessary to use technology in order to teach what algorithms are. Unplugged activities - teaching computing without technology - involve problem solving to achieve a goal and dealing with fundamental concepts of computer science in the process.
- Experts recommend multiple pathways for the development of CT throughout compulsory education to give learners the opportunity to design, create and experiment in areas they care about and relate CT activities to real life situations.



Source: Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K. (2016). Developing computational thinking in compulsory education – Implications for policy and practice; p. 22

What the research shows

More generally, research since the early days of technology in schools shows that ICT can:

- Support innovation and new ways of organising learning in time and space
- Support effective pedagogies, notably active learning, collaboration, project-based learning, independent learning and personalisation
- Motivate and engage students and help them understand complex concepts, providing them with richer and more compelling learning environments, and improving productivity
- Support access and inclusion, in particular of students with disabilities, those with learning difficulties, and those from a different language background.
- Help students develop digital age competences, including higher order thinking skills, creativity and digital competence
- Enable new forms of feedback and assessment, including learning analytics and adaptive learning, games and simulations
- Make possible activities that would otherwise not be possible for example showing dangerous experiments, enabling collaboration over distance, and involving outside experts
- Prepare students for life and work after school and to play their part in a society which has transformed the way young people communicate, seek help, access information and learn.

3 key factors for the successful use of ICT in education

1. The school needs to have a positive culture of innovation, reflection and improvement
2. Technology has to be fit for purpose, accessible and perform reliably
3. Teachers need appropriate competences and support.

European Schoolnet

Publisher:
European Schoolnet
(EUN Partnership AIBSL)
Rue de Trèves, 61
1040 Brussels, Belgium
t +32 (0)2 790 75 75 | f +32 (0)2 790 75 85

Authors: Anja Balanskat, Katja Engelhardt, Anusca Ferrari
Editor: Caroline Kearney
Design: Jessica Massini
Picture credit: European Schoolnet, Shutterstock
Web version: www.eun.org/observatory/perspectives
More information: info@eun.org