Coding, Programming and the Changing Curriculum for Computing in Schools

Report of UNESCO/IFIP TC3 Meeting at OCCE – Wednesday 27th of June 2018, Linz, Austria
Date of report: 4 February 2019

Present:
Davide Storte, (UNESCO)
Mary Webb, UK (TC3 Curriculum Task Force chair)
TC3 Curriculum Task Force members:

- Rosa Maria Bottino, Italy
- Don Passey, UK
- Ivan Kalas, Slovakia
- Christine Bescherer, Germany
- Joyce Malyn Smith, USA
- Charoula Angeli, Cyprus
- Yaacov Katz, Israel
- Peter Micheuz, Austria
- Sindre Rosvik, Norway.
- Torsten Brinda, Germany
- Andrew Fluck, Australia
- Johannes Magenheim, Germany
- Bent B. Anderson, Denmark
- Gerald Fuschek, Austria

Representatives invited by UNESCO:

Marcel Jacques Simonette, Brazil
Edison Spina, Brazil
Khalid Berrada, Morocco

Others who did not attend but have contributed:

Maciej M. Sysło, Poland
Andrej Brodnik, Slovenia
Amina Charania, India
Tim Bell, New Zealand
Niki Davis, New Zealand
Gerard MacManus, New Zealand

Executive Summary

A meeting in Linz, Austria in June 2018 was convened by UNESCO together with IFIP TC3 in order to examine the needs and challenges/opportunities in relation to changing Computing curricula and the renewed focus on "coding" or programming and a stronger Computer Science content. The following key challenges previously identified (see task force/EDUSummIT report – Appendix 1) were all considered by delegates to apply in their own countries but with varying importance at different stages in their processes of curriculum change.

Key Challenges (see Task Force TC 3/EDUSummIT Report Appendix 1 of this report)

1. Lack of clear understanding of Computer science/Informatics as an academic discipline.
2. Need for Computer Science/Informatics as a distinct subject in school curricula is controversial and poorly understood.
3. Computational thinking, a core component of Computer Science/Informatics this considered to be important in 21st century skills, but due to its complexity, it is difficult to implement in schools.
4. The development of Computer Science/Informatics school curricula is impeded by insufficient empirical evidence of student learning in order to support content definition and sequencing.
5. Previous ICT curricula deliveries poorly prepared students for Computer Science/Informatics in further/higher education or professional employment.
6. Integrating Computer Science/Informatics across other subjects in school curricula has been ineffective.
7. Teacher professional development in a newly introduced Computer Science/Informatics subject is a challenge in quality and quantity.
8. Identifying and allocating the additional resources for teaching Computer Science/Informatics is a challenge.

Major Challenges

Challenge 1, the lack of understanding of Computer Science as an academic discipline, together with Challenge 7, the challenge of providing sufficient teacher professional development, were considered by all delegates to be major challenges. Addressing the challenges has not been straightforward. A number of specific tensions were identified which constrain approaches to solving the challenges. Typically, there are no ideal solutions and different approaches have their own advantages and limitations.

Major Tensions

The following specific tensions were identified:

1) The urgency of curriculum change versus the need for costly and time-consuming professional development of teachers.

Solution approaches identified:

a) Introducing a new curriculum even though professional development of teachers was known to be inadequately resourced. This solution, used in England, created massive pressure on teachers.
b) Delay change owing to the need for professional development of teachers.
c) Make more use of informal learning including online resources and courses.

2) Competition for resources – the need to improve learning across all subjects versus the need to develop computer science education.

Solution approaches identified:

a) Make increased use of "unplugged activities" (`https://csunplugged.org/`)to develop conceptual understanding in computer science.
b) Make more use of informal learning including online resources and courses.

3) Safety issues associated with increasing use of social media versus the potential for educational use of mobile phones.
Additional challenges:
Mobile phone bans affecting opportunities for students to use their own devices (BYOD).

Table of contents

Contents

Introduction

A meeting in Linz, Austria in June 2018 was convened by UNESCO together with IFIP TC3 in order to examine the needs and challenges/opportunities in relation to changing Computing curricula and the renewed focus on "coding" or programming and a stronger Computer Science content. A Task Force within IFIP TC3 had examined issues regarding Computing curricula and key challenges and opportunities (Fluck, Webb et al. 2016, Webb, Davis et al. 2017). Delegates to the Linz meeting discussed the following questions and challenges from the perspectives of their own country and context.

Questions

1) Key challenges, identified previously (see Appendix 1) are listed below. Which 2 or 3 of these have proved (are proving) particularly challenging for you and how have you addressed them?

2) Are there any new challenges (not identified below) that you have identified as being particularly important in the situation in your country?

3) If we were to produce some kind of report/guidelines/curriculum recommendations on behalf of TC3 and UNESCO what do you think might have been or might still be useful in your country for making decisions about the Computer Science/Informatics curriculum?

Key Challenges (see Task Force TC 3/EDUSummIT Report Appendix 1 of this report)

1. Lack of clear understanding of Computer science/Informatics as an academic discipline.
2. Need for Computer Science/Informatics as a distinct subject in school curricula is controversial and poorly understood.
3. Computational thinking, a core component of Computer Science/Informatics this considered to be important in 21st century skills, but due to its complexity, it is difficult to implement in schools.
4. The development of Computer Science/Informatics school curricula is impeded by insufficient empirical evidence of student learning in order to support content definition and sequencing.
5. Previous ICT curricula deliveries poorly prepared students for Computer Science/Informatics in further/higher education or professional employment.
6. Integrating Computer Science/Informatics across other subjects in school curricula has been ineffective.
7. Teacher professional development in a newly introduced Computer Science/Informatics subject is a challenge in quality and quantity.
8. Identifying and allocating the additional resources for teaching Computer Science/Informatics is a challenge.

Findings

The challenges identified previously were all considered by most delegates to apply in their own countries but with varying importance at different stages in their processes of curriculum change. There was also general agreement about which were the major challenges and the underlying reasons for these challenges as explained here.

Major Challenges

Challenge 1, the lack of understanding of Computer Science as an academic discipline, together with Challenge 7, providing sufficient teacher professional development, were considered by all delegates to be major challenges. The lack of understanding of the nature and importance of Computer Science stems partly from the low visibility of Computer Science compared with the very obvious and ubiquitous nature of user applications and changes in the digital world. Whereas the benefits and dangers of social media and business applications are becoming widely known, the underlying systems, the way they are changing and the knowledge needed for their development are not visible. Furthermore, there are obvious and important concerns regarding Internet safety and these may focus policymakers on attending to such concerns rather than the broader needs.

Associated with Challenge 1 is the need for careful use of terminology and making this available to policymakers. This issue is compounded by different uses of terms in different countries as well as varied and changing terminology throughout the history of development of Computer Science education.

This lack of knowledge of Computer Science as an academic discipline is particularly common among policymakers but is also often shared by teachers themselves, especially as many teachers have responsibility for teaching elements of the computing curriculum without having studied Computer Science themselves. Therefore, many countries have focused on the surface features and neglected the academic discipline that enables and underlies development of the technologies.

Challenge 2, the need for Computer Science as a distinct subject in school curricula relates to Challenge 1 in that some countries, e.g. Italy, France and Finland incorporate Computer Science inside their mathematics and technology curricula. These issues are compounded by the pressure on space in the curriculum resulting from new developments across many subject areas. Furthermore, although in this meeting we were endeavouring to compare and discuss the issues across different countries, it is important to recognise that in many countries there is no centralised curriculum and curriculum decisions are made regionally as in Germany, for example, or even at school level. For example, in Italy schools have a great deal of autonomy in how they design their curricula. In some diverse countries, such as Brazil, different regions of the country differ markedly in their infrastructure and resource availability. Challenge 3, regarding computational thinking, also affects the understanding of Computer Science. The term, computational thinking, is poorly understood and there are significantly different interpretations in various contexts. While some interpret computational thinking wrongly as synonymous with Computer Science, or even more narrowly, as being primarily about programming, others take a very broad view of computational thinking and ignore the need for a strong connection with Computer Science.

Teacher professional development is a major challenge for many countries because: 1) most have insufficient graduates in Computer Science coming through to train as teachers and 2) in some countries existing teachers of ICT have very limited qualifications and understanding in relation to new computing curricula. Teachers of ICT often face particular challenges in developing their capability to teach programming as many have had no or very limited education in programming themselves. The continuing professional development needed is time-consuming and costly.

Major Tensions identified while addressing the challenges

Addressing the challenge as discussed above has not been straightforward. In some cases, solutions have been found which often differ in different countries and different situations. The following specific tensions were identified. Typically, there are no ideal solutions and those adopted have depended on a number of factors including:

Tension 1: The urgency of curriculum change versus the need for costly and time-consuming professional development of teachers.

Solution approaches identified:

a) Introducing a new curriculum even though professional development of teachers was known to be inadequately resourced. This solution, used in England, created massive pressure on teachers. However, introducing the new curriculum as a matter of urgency, together with some monitoring and evaluation, did highlight the need for professional development, which was then addressed with the provision of substantial funding.

b) Delay change owing to the need for professional development of teachers. This approach can be more supportive for teachers but may delay much-needed change. This approach was adopted in Denmark.

c) Make more use of informal learning including online resources and courses rather than changing the formal curriculum. It was agreed that informal learning has much to offer in many contexts and opportunities for informal learning are increasing. In some contexts, informal learning can provide a bridge while curriculum change develops more slowly. However, most delegates agreed that solutions based on informal learning should not obscure the important policy decisions needed to recognise the importance of establishing computer science as part of the formal curriculum.

Tension 2: Competition for resources – the need to improve learning across all subjects versus the need to develop computer science education.

Solution approaches identified:

a) Make increased use of "unplugged activities", which are typically less resource-intensive, to develop conceptual understanding in computer science. However, the meeting recognised that "unplugged activities" are part of the solution and not the complete solution as typically they complement the use of computers.

b) Make more use of informal learning including online resources and courses that do not depend on school-based resources. Learning programming, in particular, is very well supported by online resources that can be used in informal context. However, as mentioned above, reliance on informal learning may detract from important policy decisions regarding curriculum content.

Tension 3: Safety issues associated with increasing use of social media versus the potential for educational use of mobile phones. Whereas, in the last few years there has been a move towards "Bring Your Own Devices" (BYOD) as a way of enabling students to develop their confidence in using technology to support their learning, more recently a backlash against the dangers of technology use have resulted in policies at school or country level heavily restricting or banning use of mobile phones in schools.

Tension 4: It has been recognised that it is important to start learning aspects of Computer Science early, including programming, but there are dangers in simply pushing down curricula, designed previously for older students, into primary education. These dangers are associated with both challenges for teachers in understanding key concepts themselves and also with ensuring that the material and pedagogical approaches are suited for students in earlier stages of development.

Solution approach identified:

a) Develop new curricula in collaboration with teachers. These curricula need to incorporate learning about safety issues and how to make appropriate use of technologies.

Key Needs to be addressed

Here we focused on ways in which a UNESCO/TC3 initiative might support countries in developing their curricula in relation to programming/coding. Four main areas were identified:

1) Access to resources and advice on how and in what circumstances particular resources might be relevant.
2) Advice concerning appropriate programming environments for learning programming

3) Advice on scheduling implementation of new curricula and cost estimates for key elements, particularly professional development.
4) Informal Learning

Individual country responses

There follow brief reports from each of the following countries, addressing the issues identified, Australia, Brazil, Cyprus, Denmark, England, Germany, Israel, New Zealand, Poland. These brief snapshots provide examples of how some of the challenges have played out, what specific tensions have arisen and what lessons have been learned.

AUSTRALIA

An Australian perspective on introducing a computer science/computing subject into schools

By Andrew Fluck, University of Tasmania,

Introduction

Australia introduced a national curriculum in 2009, which includes Information and Communication Technology (ICT) as a cross-curriculum capability, assessed in all subjects. This is a fairly office-applications-centric approach, but it does provide opportunity for students K-10 to master the basic use of a computer. In 2015, the 'Technologies' learning area of the Australian Curriculum was split in two, with the addition of a new discrete subject called 'Digital Technologies'. This focuses on creating digital solutions K-8 with optional material for Years 9 and 10.

Challenges

Our current challenges are Challenge 7 and Challenge 8 (above) – Teacher professional development for the new Digital Technologies subject, and building up infrastructure and equipment to the point where teachers can rely upon their availability and function.

Teacher training has NOT been planned or provided on a national scale. Since each state & territory has constitutional responsibility, this aspect of introducing the new Digital Technologies subject is diverse. A series of age-related MOOCs has been made available from the University of Adelaide with Google funding, but most teachers have to study these in their own time.
The Australian Government has created and maintains The Digital Technologies Hub (https://www.digitaltechnologieshub.edu.au). This provides teachers with a central curriculum reference point, scope and sequence documents, lesson plans and digital learning resources.
A new challenge is the rise of social media and associated negative impacts on children. These are leading to re-evaluation of BYOD policies and reviews are being conducted into the use of mobile phones in schools. The result is to exacerbate equipment shortages for studying Digital Technologies. While 'unplugged' materials are a useful teaching supplement, it is the view of this author that they cannot entirely replace the learning experience of using a computer.

Lessons learnt and ways forward

In hindsight, Australia could have planned better. For example, one state which intended to report student progress in Digital Technologies to parents by the end of 2018, decided to delay this to 2020 or later, because of the lack of reliable equipment and low teacher preparedness. Therefore, it would be useful to provide advice on scheduling and synchronising implementation, and indicate cost estimates for essential elements such as:

Teacher professional development
Policies for infrastructure and equipment
Curriculum development and implementation timelines
Provision of a central reference point

On a personal note, it may also be useful to provide advice on the number and variety of programming paradigms to which students should be exposed. Given its rising importance, it might also be suitable to include quantum computing experiences[1] in the later years of schooling.

---

[1] http://andrew.fluck.id.au/QCintroduction

BRAZIL

Key Challenges from the TC3 / UNESCO Meeting at OCCE 2018:

Perspective and Comments: CEST/ USP – Brazilian scenario

Edison Spina, University of Sao Paulo, Brazil spina@usp.br
Marcel Jacques Simonette, University of Sao Paulo, Brazil marceljs@usp.br

Introduction

We have presented to the members of our Study Center the eight challenges identified at EDUsummIT 2015. The first reaction of the group was discussion about: "Why computational thinking? Why not system thinking, complexity thinking, or critical thinking?" We realize that it is not a matter of choice between one thinking or another, all types of "thinking" are important, and "computational thinking" helps organize the logic behind the thinking issues.

Challenges

Analysing the challenges, we have identified three "big" challenges considering the Brazilian scenario:

2 - Need for Computer Science/Informatics as a distinct subject in school curricula is controversial and poorly understood.

> We still have problems with mathematics (including logic) and Portuguese (language) education, it is necessary to develop a clear understanding of the need for Computer Science. However, there are private schools in some Brazilians cities that already have Informatics in their curricula, at the same time, there are other cities in which the public schools do not have electric energy.

4 - The development of Computer Science/Informatics school curricula is impeded by insufficient empirical evidence of student learning to support content definition and sequencing.

> The evidence is indeed a problem in the development of computer science school curricula. High Education educators have a better understanding of this issue. However, elementary and junior high schools' educators really need substantial evidence. Nowadays, mathematics and Portuguese are the significant problems in Brazilian education.

7 - Teacher professional development in a newly introduced Computer Science/Informatics subject is a challenge in quality and quantity.

> In this topic, we have the problem: who will teach the teacher? It is necessary to graduate a new set of teachers that will have the knowledge required to train the teachers.

Lessons learnt and ways forward

Concluding, we believe that it is necessary to develop more reports about the need for Computer Science/Informatics as a distinct subject in school curricula. Also, it is essential to establish guidelines to teach teachers in Computational thinking, in formal and informal ways. Furthermore, it is essential to develop a dictionary/glossary of the terms, as there are different concepts being used, as: computational thinking, algorithmic thinking, coding skills, informatic skills, etc.

CYPRUS

The case of Cyprus

Charoula Angeli, University of Cyprus, Cyprus, cangeli@ucy.ac.cy

Background Information- centralized educational system

Primary Education (grades 1-6)

- Digital technology is used as a tool in teaching.
- Computer science (CS) is not part of the elementary school curriculum.

Secondary Education (grades 7-12)

- Computer Science as a subject-matter has been taught in grades 7-12 since 1988.
- It is mandatory for grades 7-10.
- Elective for grades 11-12.
- Taught by teachers who have a bachelor's degree in Computer Science.

In May 2018 the Minister gave the go-ahead for the development of a new elementary curriculum:

1. To design and develop the curriculum of a new thematic standalone unit on "New/digital Technologies" as part of the Design and Technology curriculum in elementary education.
2. But he mostly appointed CS teachers so we took that as giving us a hint about integrating CS in the elementary school curriculum.
3. The new unit should consist of 16 teaching periods (40 minutes each) in grade 5, and the same for grade 6

Key challenges

1. Define conceptually the term "New Technologies" and develop a framework for developing the curriculum.
    - Operational framework for NT that can be actually be taught by teachers.
    - What should this curriculum include? How a CS curriculum should make connections between computer science and society?
2. How many hours per area/topic?
3. Access to reliable computer labs.
4. Access to reliable wifi.
5. Access to resources/learning materials.
6. How to train the teachers.
    - Start with the teachers who teach design and technology and use those as multipliers to train the rest.

DENMARK

Danish perspectives on introductory informatics courses

By Bent B. Andresen, Aarhus University, bba@edu.au.dk

Introduction

Earlier this year, the Danish Government launched "Action Plan for Technology in Education". According to this plan, students in primary and lower secondary schools need to have a general idea about computing, i.e., what it is. The competence development includes coding and the creation of digital products. This trend is in line with the shift in focus in a number of other countries.

In Denmark, most students are digital literate. Recently, a comparative study provided evidence suggesting that the 4th grade students are among the best in the world navigating and reading web-content (Progress in International Reading Literacy Study, 2017). Additionally, another comparative study provided evidence that the 8th grade students are more computer and information literate than the students in most other countries (First International Comparative Study on Computer and Information Literacy, 2014). The Danish Government now plan to strengthen the so-called "Technology Comprehension".

Challenges

A challenge is a lack of clear understanding of this notion of Technology Comprehension. It includes but also goes beyond programming. It is considered essential for learning, working or talking responsible decisions in the digital age. So far, there is no precise description of the learning objectives in this area. In particular, it is unclear how the "Technology Comprehension" is related to self-directed use of digital technology to access, manage, integrate, evaluate and create information.

According to the Action Plan, the vision is that all children, adolescents and adults not just use the digital technology – they should also be able to relate critically to and shape this technology. Everyone does not need to become ICT specialists, but children, young people and adults should be able to commit themselves to digital communities and develop 21st century skills including "ICT-creativity" and "digital literacy".

The implementation period lasts four years. In this period, a one-year elective subject "Technology comprehension" will be introduced in primary and lower secondary school. The content includes programming, design of digital products and significance of digitization in society. Consequently, there is an increased focus on creating digital products and programming has returned to primary and lower secondary school.

In the four years plan period, other initiatives include:

1) A competition for students in 6th-8th grade for innovative problem solving using ICT (digital skills/CRAFT)
2) Development of digital literacy including coding skill for students in 4th grade (based on an idea developed by BBC/Micro:bit)

3) Increased focus in the final examinations of vocational training on evaluating students' digital skills).

For many years, teacher professional development regarding use of digital technology has been a challenge in Denmark. Due to the need of primary and lower secondary school teachers for digital literacy, The Danish Government want to increase professional development regarding the use ICT in 1:1 classrooms, which was implemented in all public K-12 schools and further education 3-5 years ago. In particular, it is planned to developing competencies regarding general "Technology Comprehension". These efforts include pre-service as well as in-service teacher education.

In Danish vocational education, the need for informatics education as a distinct subject in the curricula is poorly understood. This includes the need for education regarding computational thinking and coding. However, eight 'experts' and I are invited in August 2018 to advise the central authorities in Denmark regarding the need for digital competencies in vocational education.

Regarding adult education, the Danish Government wants to fill some of the gaps. More concretely, it wants all adults to develop "stronger basic IT skills" as a foundation for continuously acquiring new knowledge and upgrading in the digital field. In Denmark, all adults can attend the so-called "Preparatory Adult Education" ("FVU") to develop their basic digital skills. In order to increase basic skills of the workforce, this offer will now be enlarged with the subject "FVU digital".

There are already some experiences in this area. Nine years ago, I developed a curriculum for the Ministry of Education. This curriculum labelled "FVU IT" encompasses three levels. The length of the education at Level 1 and 2 is 40-60 lessons per level, and the education at Level 3 has a range of 60-80 lessons. The overall learning objective the first level is that the participants develop skills in assessing and processing digital content. The overall learning objective at the second level is that the participants automate already developed skills in searching, processing, arranging and collecting digital information and materials. At Level 3, the learning objective is that the participants improve or develop their knowledge about the use of ICT as a means of expression when performing a job, educating, or acting as active citizen.

This curriculum was implemented at adult education centres in Denmark. The experiment was thoroughly evaluated, and the evaluation results provided evidence suggesting that all the learning objectives were met. However, allocating resources was a challenge. After some years, the provision of education for adults in this area stopped. In the near future, it will be revitalized with the provision of "FVU digital".

Source:

Ministry of Education. (2018). Action plan for technology in Education. Copenhagen: Ministry of Education.

ENGLAND

A perspective from England on introducing Computer Science into schools

Mary Webb, King's College London, mary.webb@kcl.ac.uk

Introduction

This report applies to England although curriculum change in Wales and Scotland is largely similar to that of England and similar issues have been identified (The Royal Society 2017), although the curriculum specifications have been created separately. Following a report from the Royal Society (2012), a new curriculum, with a strong focus on Computer Science but also incorporating Information Technology and digital literacy, was introduced to England in 2014.

Key Issues and Challenges

All of the 8 challenges have applied during the process of changing the curriculum in England. Probably the most significant challenge has been teacher professional development. Professional development was a major challenge because the curriculum in England had been not fit for purpose (The Royal Society 2012). The "ICT" curriculum had gradually shifted over a number of years to a very limited offering of basic digital skills using office applications. Therefore, teachers, who were often drafted in from other subject areas, had no experience of teaching technical aspects of computing and programming. Very few of the existing ICT teachers had qualifications in Computer Science and even those who had suitable qualifications had been de-skilled as result of teaching this very limited curriculum. Funding for professional development was very limited, given the scale of the problem, and targets set for recruiting new teachers of computing were not met.

The British Computer Society and the grassroots teachers' organisation, Computing at School (CAS), strongly supported the curriculum change and set up a "Network of Excellence", a community of practice, based largely on voluntary contributions from higher education, in partnership with schools, to support the development of resources and teacher professional development. Although these initiatives involved many enthusiastic participants and had significant success, especially in interpreting curriculum requirements and developing resources, these initiatives were nowhere near sufficient to meet the challenge of professional development (The Royal Society 2017). Recently, in order to address the shortage of teachers of computing and the professional development challenge, the UK government has set up a "National Centre of Computing Education", with funding of £100 million and the aim to train 8000 teachers.

Challenge 1, the lack of clear understanding of Computer Science as an academic discipline, has been significant in developments in England. The problems identified by the Royal Society regarding the impoverished nature of the ICT curriculum (The Royal Society 2012), were created by this lack of understanding by policymakers and others. Following the curriculum reform in 2014, arguably problems with the development of new qualifications were predominantly due to lack of understanding by policymakers of the overall scope of curricula in relation to Computer Science and Information Technology. Thus, although the National Curriculum specification incorporated Computer Science, Information Technology and digital literacy, academic qualifications, at 14-19 years old, became exclusively focused on Computer Science with a neglect of Information Technology. Specifically, GCSE and A-level qualifications in ICT were discontinued and not replaced with appropriate qualifications to cover the Information Technology elements of the National Curriculum. These developments also coincided with a demand from politicians for greater academic rigour and harder courses so that at GCSE-level (14 to 16) and above Computer Science has come to be considered as a very difficult subject to be taken only by very able students.

Consequently, relatively few students are studying any aspects of computing beyond age 14 and the gender imbalance is continuing, with relatively few girls choosing to study Computer Science (Kemp, Berry et al. 2018).

Lessons Learnt

The introduction of a new curriculum without adequate resources for professional development was controversial and there are many who believe that a more managed introduction would have been preferable, not least many teachers who have felt overworked or overwhelmed by the rapid change. However, analysis of the examination data between 2014 and 2017 shows that an increasing percentage of schools are offering Computer Science qualifications at age 14-19 (Kemp, Berry et al. 2018) so this may be considered as a sign of success. Furthermore, the Royal Society's more recent review (The Royal Society 2017) reported that there was much to celebrate and advocated building on the work of the communities of practice and centres of excellence, with a specific focus on providing adequate resources for professional development.

More specifically, the provision of an appropriate range of courses to enable students at 14-19 to study appropriate material in computing remains challenging. Arguably and with hindsight, problems with the limited range of qualifications, might have been avoided by reviewing the whole landscape of Computing content and approaches at the same time, rather than focusing initially on developing new Computer Science specifications.

References

Kemp, P., M. G. Berry and B. Wong (2018). The Roehampton Annual Computing Education Report, London: University of Roehampton.

The Royal Society (2012). Shut down or restart? The way forward for computing in UK schools. London, The Royal Society.

The Royal Society (2017). After the reboot: computing education in UK schools. London, The Royal Society.

GERMANY

A German perspective

Torsten Brinda, Univ. of Duisburg-Essen, Germany, torsten.brinda@uni-due.de
Johannes Magenheim, Univ. of Paderborn, Germany, jsm@uni-paderborn.de
Christine Bescherer, Ludwigsburg University of Education, Germany, bescherer@ph-ludwigsburg.de

Challenges

1. Lack of clear understanding of Computer science education /Informatics as a school subject.

In Germany, computer science has been available as an optional or mandatory-optional school subject throughout the country since some decades already, but only in few federal countries it has been implemented as a mandatory subject for all students. In 2016, the board of ministers of education and cultural affairs published a strategy document on "education in the digital word"[2], a 60-pages long document, which mostly focuses on digital literacy and competent usage of digital tools but does not include any considerations concerning computer science education in a separate subject. The only explicit connections drawn state that the competences to be acquired are "so important that they cannot be addressed in a single subject like computer science but have to be integrated in all subjects". The document, which is now discussed in all federal countries and used for implementing digital education strategies, does not show a clear understanding that learning in the digital age is not only about learning with and using digital technology, but also about learning about digitalization and the underlying concepts.

In this context, the need to strengthen 'computational thinking' (Wing, 2006) is always mentioned. However, the term is controversially discussed even at the academic level (see, e.g., Denning, 2017) and is interpreted in a completely different manner in the context of educational policy. Everyone interprets the term as they see fit it for their educational interests. The spectrum of interpretation of computational thinking ranges from modeling, coding, and programming to the understanding application of informatics systems. Based on these different concepts, the term is then also used to justify CSE as an independent school subject or as an integrative learning area, integrated into all subjects.

A few years ago, the earlier German Minister of Economic Affairs (Gabriel) suggested the introduction of a new subject called "Programming language". Among political and other stakeholders within the education systems, computer science is very often equated with "programming" or "programming language" or "using digital technology", which indicates limited or even misconceptions about the discipline. However, these persons are very often the ones who make or influence the political decisions concerning this field.

In 2016, the German Informatics association organized a workshop on "education in the digital networked world" and brought together around 30 experts from computer science, computer

[2] https://www.kmk.org/fileadmin/Dateien/pdf/PresseUndAktuelles/2017/KMK-Strategie_Bildung_in_der_digitalen_Welt_Zusammenfassung_en.pdf

science education, media education, secondary schools, and politics. In this workshop, a model was developed (the *Dagstuhl triangle*, see Brinda/Diethelm 2017), which was well recognized in the German speaking countries. It integrates learning with and about digital technologies and also reflection about its impact on the individual and society. This model and a number of presentations and discussions with stakeholders were fruitful and contributed to the awareness that computer science can contribute to understanding of and participating in the digital world.

The 2018 German National report to IFIP TC3 also addresses and summarizes these problems (Source).

2. ***Need for Computer Science/Informatics as a distinct subject in school curricula is controversial and poorly understood.***

The advantages and disadvantages of computer science as a separate subject, integration of computer science aspects into other subjects, and teaching computer science through projects or extracurricular activities were discussed in (Brinda 2017). The same argumentation was used in a comment paper of the German Informatics association answering to the above-mentioned strategy paper[3].

3. ***Computational thinking, a core component of Computer Science/Informatics this considered to be important in 21st century skills, but due to its complexity, it is difficult to implement in schools.***

In Germany, this does not seem to be an issue, because there are a number of successful implementations of computer science education from primary school projects to advanced classed in upper secondary education. However, most of these classes are optional. With only optional classes, the goal of school to prepare everybody for life and work also in the digital world cannot be reached by everybody, because it depends on decisions made throughout the school career. The main problem for the implementation in school seems to be the following: the school curriculum is already quite full with other subjects, so a new mandatory course would have to be added to that or something else would have to be reduced or even removed. If the time to be spent at school would be extended, political stakeholders fear discussions with parents, students and teachers. If any other subject would be reduced at school, the political stakeholders fear discussions with the representatives of the respective subject. The education administration also often argues that there is a lack of qualified informatics teachers (chicken and egg problem). To avoid any discussion, they abstractly delegate computer science-related competences to all other subjects, which cannot work for the majority of teachers.

4. ***The development of Computer Science/Informatics school curricula is impeded by insufficient empirical evidence of student learning in order to support content definition and sequencing.***

In Germany, this is not a problem. A number of successful curricula exist in the federal countries. Empirical research concerning various fields is progressing and made available to teachers for

[3] https://fb-iad.gi.de/fileadmin/FB/IAD/Dokumente/gi-fbiad-stellungnahme-kmk-strategie-digitale-bildung.pdf

ongoing improvement of learning and teaching computer science. However, empirical research on computer science education in schools is still relatively young and not as multi-faceted as in the other STEM subjects. There are relatively few well-founded studies. Existing studies often have low case numbers, are sometimes methodologically critical, and the impact on teacher training in CSE and the current practice of computer science teaching is probably still rather low.

5. ***Previous ICT curricula deliveries poorly prepared students for Computer Science/Informatics in further/higher education or professional employment.***

The integration of ICT in various subjects has also been tried out in Germany in the last decades. The results of the ICILS study 2013 have shown that the development of computer and information literacy was below average in comparison to other participating countries. These results were a main motivator for the above-mentioned strategy paper of German educational politicians.

6. ***Integrating Computer Science/Informatics across other subjects in school curricula has been ineffective.***

see 5.

7. ***Teacher professional development in a newly introduced Computer Science/Informatics subject is a challenge in quality and quantity.***

Teacher qualification is a huge challenge. In Germany, teachers with a wide range of qualifications teach (about) computer science. In this context, a distinction should be made between three groups of teachers:

1. teachers who teach computer science (mostly at Gymnasium or in higher secondary schools).
2. teachers who teach computer science as an elective subject and who often do so without any specialist or didactic qualification. We simply do not know what is taught regarding content (possibly only user software training?)
3. teachers of all subjects who are expected to teach computer science education in their respective subjects in an integrated manner

There are teachers from other subjects, who follow material (such as provided by code.org) in projects or optional courses, others who have been qualified more or less in in-service certification courses (one to two years with few days per week), and those who studied a regular teaching degree program at a university (5-years program). The resulting qualification spectrum leads presumably to a wide spectrum of more or less elaborated computer science-related learning offers in class. Until now, there has not been any empirical research investigating this in Germany. Individual reports however indicate that sometimes “computer science” classes just focus on using application software and in other cases computer science is purely taught with blackboard and chalk. The introduction of computer science as a new subject in some of Germany’s federal countries was prepared with in-service training of teachers of other subjects combined with an expansion of the teacher education in computer science at the universities of the respective federal country.

Regarding group 3, a huge (quantitative) need for teacher training arises if something like a `media pass´ or integrative informatics education is to be implemented in all subjects, although such concepts failed grandiosely in Germany in the 1980s and 1990s. This would create a massive need for teacher training, which would have to relate to both the scientific fundamentals of computer science and its didactic teaching in the context of the respective subject. Unfortunately, it is becoming apparent that this need for teachers' qualifications is not taken into account by the educational administration and that it is expected that the experienced teachers will 'somehow' implement it.

Considering the design of the UNESCO Curriculum Guidelines, the question arises in this context as to what extent these various competency requirements of teachers should be referred to at least in a brief form. The qualification requirements should then also be described with regard to corresponding theoretical concepts of teacher competences at a general level (PCK, TPCK; e.g. Mishra/Köhler 2006) and specific competences for computer science teachers (e.g. Informatics PCK; Bender et al. 2015).

8. ***Identifying and allocating the additional resources for teaching Computer Science/Informatics is a challenge.***

see 3.

To cope with the problem of available time for a curriculum, a solution could be a combination of several actions:

- reduction of the lesson length by a few minutes, collection of this time to create space for the needed subject
- moderate extension of the time at school
- Germany provides hours for the curriculum, which schools can fill with individual content (for example for fostering mathematical and language competencies) → use part of such hours to extend computer science education

Additional Challenges

- Media education representatives are better networked with educational politics than representatives from the computer science education community
- Media education representatives have key positions in the ministries of education and thereby presumably prevent the extension of computer science education at school
- Lack of understanding, what Informatics/CS is among educational politicians
- Individual (mis-)conceptions about computer science among educational politicians
- Tendency to avoid conflicts among educational politicians -> competencies abstractly delegated into other subjects
- Many stakeholders: other stuff is more important (reading, writing, maths, …); CSE should only be part of vocational training; Students learn a lot about the use of informatics systems during their spare time with peers – why therefore waste lessons in schools

Lessons learnt and Recommendations

- Make sure, Ministers are reached with recommendations
- Provide some concrete recommendations, how to overcome the problem of limited resources concerning time etc.

- Decide on a qualification framework for teachers, who teach computer science. Maybe also for those, who have to integrate CS aspects in the ordinary teaching of their subjects: Fundamental CS competences for teachers (→ PCK, TPCK see 7. above)
- Write a paper from the educational politicians' perspective: what are the benefits for them, …, how to avoid conflict, …
- Make recommendations available in different languages
- The organizational and content-related linking of formal and informal educational offers (learning laboratories, museum pedagogy, extracurricular projects of schools, computer science competitions (in Germany BWINF, Biber,..) leisure pedagogy (boot camps) which offer CSE-related educational programs with and without schools is also of great importance for computer science education.

References

- Bender, E., Hubwieser, P., Schaper, N., Margaritis, M., Berges, M., Ohrndorf, L. et al. (2015). Towards a Competency Model for Teaching Computer Science. Peabody Journal of Education, 90 (4), 519–532. doi:10.1080/0161956X.2015.1068082
- Brinda, T.; Diethelm, I.: Education in the Digital Networked World. In: Tatnall, A.; Webb, M. (Hrsg.): Tomorrow's Learning: Involving Everyone. Learning with and about Technologies and Computing 11th IFIP TC 3 World Conference on Computers in Education, WCCE 2017, Dublin, Ireland, July 3-6, 2017, Revised Selected Papers. Springer, Cham, Switzerland 2018, S. 653-657.
- Brinda, T.: How to Implement Computing Education for All – Discussion of Alternative Organisational Models. In: Tatnall, A.; Webb, M. (Hrsg.): Tomorrow's Learning: Involving Everyone. Learning with and about Technologies and Computing 11th IFIP TC 3 World Conference on Computers in Education, WCCE 2017, Dublin, Ireland, July 3-6, 2017, Revised Selected Papers. Springer, Cham, Switzerland 2018, S. 648-652.
- Denning, Peter J. (2017), Remaining trouble spots with computational thinking, In Communications of the ACM, Volume 60 Issue 6, June 2017 Pages 33-39
- Mishra, P. & Koehler, M. J. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. Teachers College Record, 108(6), 1017–1054.
- Wing, Jeannette M. (2006) Computational thinking, In Communications of the ACM, Volume 49 Issue 3, March 2006 Pages 33-35

ISRAEL

Key Issues Emerging from the TC3 / UNESCO Meeting at OCCE 2018:

The Israeli Perspective

Yaacov J Katz, Bar-Ilan University and Michlalah - Jerusalem Academic College, Yaacov.Katz@biu.ac.il

1. When technology was introduced into Israeli elementary and junior high schools in the early 1970s and up to the end of the first decade in the 21st century, the mastery of computer literacy was declared to be the main goal of technology use.
2. Mastery of basic computer applications that facilitate word processing, presentation of assignments, familiarity with email and web browsers, and access to relevant databases was deemed to be of paramount importance.
3. Technology served solely as an important auxiliary learning accessory limited to mastery of technological techniques that facilitated students' efficient learning.
4. No attention was paid to the possibility that mandatory study of computer science could lead to a fundamental contribution to the improvement of cognitive processes and enhancement of learning.
5. Towards the end of the first decade of the 21st century it became clear that the use of technology as an educational accessory was not significantly contributing to fundamental cognitive processes.
6. Researchers began examining the potential contribution of technology to enhanced cognitive processes that would lead to a significant improvement of student learning.
7. Results of research studies have indicated that the mandatory study of computer science, which includes exposure to computational thinking skills including abstraction, algorithmic coding, analysis of data and problem-solving design, could potentially lead to the enhancement of students' cognitive processes.
8. Thus, in September 2016 the Israel Ministry of Education began the gradual implementation of a new computer science curriculum for elementary and junior high schools. The curriculum focuses on the use of computational thinking (including abstraction, algorithmic coding, analysis of data and problem-solving design) as a platform to enhance students' cognitive processes.

ITALY

Computational Thinking – The Italian case

Rosa Bottino & Augusto Chioccariello, bottino@itd.cnr.it

Institute of Educational Technology of the Italian National Research Council (ITD-CNR)

Introduction

Since 2014, In Italy there has been a discussion on the introduction of computational thinking in compulsory schools (age 6-16) led by the Ministry of Education.

In 2015, a general school reform was launched with the approval of the law n.107/2015. This law included computational thinking among key educational objectives.

The 107 law introduced the Italian National Plan for Digital Schools which is a three years plan which is composed by 35 different actions for school digital innovation.

In this plan there are different activities oriented to the introduction of computational thinking including a revision of related national indications. There is also a national project to introduce coding in primary school. This project includes 10 hours of coding in each primary school class. Till now, despite the fact that it is not compulsory for schools, the project has been very successful (e.g. Italy is the country scoring the highest number of participants in the European Code week since its introduction).

The scientific committee for the revision of the national guidelines (Indicazioni Nazionali)[4] has recently released a document proposing the inclusion of computational thinking in compulsory school curricula.

When the Ministry of Education published the document, clarified that the revision will not imply the introduction of a new subject neither in primary school nor in lower secondary school. This will probably imply that computational thinking will be introduced as a transversal subject in primary school and within the existing subjects of Mathematics and Technology in lower secondary school. This is a situation not uncommon in Europe – consider, for example, France, Finland, Sweden, Norway which have recently completed a curricula reforms following this path.

Key Challenges

- The necessity of a good definition of computational thinking teachers can refer to (in order to identify objectives, assessment criteria, and possible synergies and differences with other subject matter).
- Accommodating new topics in the curriculum poses a number of difficulties, not least that it implies taking hours away from other activities.
- The introduction of CT in the curriculum calls for major in-service teacher training initiatives to up-scale competences.

[4] The Italian MoE defines the national guidelines for the curriculum that serve as a guide for the formulation of curricula at the school level

- The necessity to involve school leaders so that they understand the importance of including CT and Programming and hence promote its implementation and support teachers' participation to in-service training initiatives.

The Institute of Educational Technology of the Italian National Research Council (ITD-CNR) has recently been involved by JRC (EU) in the publication of a Science for Policy report on Computational Thinking which provide a comprehensive overview and analysis of recent findings and grassroots and policy initiatives for developing CT as a competence for the 21st century among schoolchildren, and also to highlight the implications for policy and practice.

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K. (2016). *Developing computational thinking in compulsory education – Implications for policy and practice*; EUR 28295 EN; https://doi.org/10.2791/792158

Moreover, the Institute of Educational Technology of the Italian National Research Council (ITD-CNR) has recently been involved also in the publication of a report on computational thinking specifically focused on Nordic European Countries.

Bocconi, S., Chioccariello, A. and Earp, J. (2018). *The Nordic approach to introducing Computational Thinking and programming in compulsory education*. Report prepared for the Nordic@BETT2018 Steering Group. https://doi.org/10.17471/54007

NEW ZEALAND

An update on teaching Computational Thinking and Computer Programming in Aotearoa New Zealand

Tim Bell – University of Canterbury College of Engineering, Christchurch
(tim.bell@canterbury.ac.nz)
Niki Davis – University of Canterbury e-Learning Research Lab, Christchurch
(niki.davis@canterbury.ac.nz)
Gerard Macmanus– Digital Technologies Teachers Aotearoa/Hobsonville Point Secondary School, Auckland
(gerard.macmanus@hobsonvillepoint.school.nz)

Overview

Digital Technologies was added to the official New Zealand curricula (there are two national curricula) at the end of 2017, with the expectation that the new topics (which included programming in an area called Computational Thinking) will available in all schools by 2020. While the curriculum covers all years of primary and secondary school, the expected stage at which students might start writing simple computer programs on a device is around "level 3" on curriculum, which is a broad range from about 9 to 13 years old. Prior to that focus is on an "unplugged" approach (giving and following instructions in the physical world) and supporting ICT skills.

This curriculum is unusual in two ways:

1. NZ is a bicultural nation so it is important that the subject is taught from a perspective of te ao Maori (a Maori world view), so that teachers and students can relate to it. Therefore there is a dual curriculum (NZC and TMoA) and, even in English Medium, biculturalism is expected, see Gaddam et al (2018).
2. It has been added to an existing national curriculum by inserting it within the "learning area" of Technology.

Key challenges

In relation to the key challenges, identified previously (see Appendix 1) the most challenging is professional development, number 7. The least challenging is probably number 4, since the curriculum changes are now formally established.

1. Lack of clear understanding of Computer science/Informatics as an academic discipline.

Although CT has been adopted into curriculum, it is still confused by some with a focus on Digital Fluencies (ICT). This addition is very different to the wholescale revision of the NZ Curriculum in 2007 when schools as a whole thought about and implemented the changes. It appears that some schools are leaving it up to the teacher in charge to make this change because they underestimate the scale of the change.

2. Need for Computer Science/Informatics as a distinct subject in school curricula is controversial and poorly understood.

It is clearly established in the curriculum, but as part of technology curriculum. Because it isn't a distinct learning area there can be some confusion about its position. In the final three years of school it is optional (as is essentially the case for all subjects), but there is a minimum level expected prior to this. Navigating these changes and working in with other learning areas is unfamiliar territory for schools, and it may take some time for this to settle down.

3. Computational thinking, a core component of Computer Science/Informatics this considered to be important in 21st century skills, but due to its complexity, it is difficult to implement in schools.

Considerable age-appropriate resources are appearing, which are being curated so that teachers can access culturally appropriate material for the age group. Resources from overseas can be confusing, given the lack of coherence with New Zealand's curriculum, terminology and bicultural approach, but locally relevant resources have been developed.

4. The development of Computer Science/Informatics school curricula is impeded by insufficient empirical evidence of student learning in order to support content definition and sequencing.

This was avoided in NZ by using *Progress Outcomes* to support sequencing. The Progress Outcomes set out in the curriculum are based on teacher judgements by a panel of teachers who reviewed a representative samples of student work. This is a form of empirical evidence that addresses having an age appropriate sequence. Because computer science was introduced to optional high school examinations in 2011, high school teachers had 6 years of experience with the concepts before they were introduced to the earlier parts of the curriculum. This has been valuable to have a teacher voice to position the new material, and also raise awareness and support colleagues teaching the junior levels. By the time full curriculum was designed and released in 2017 there had been a range of professional development, especially "by teachers for teachers" in which high school teachers supported primary colleagues, and a strong national subject association had formed (Digital Technologies Teachers Aotearoa, or DTTA).

5. Previous ICT curricula deliveries poorly prepared students for Computer Science/Informatics in further/higher education or professional employment.

There had been heavy reliance in the past on piecemeal assessment practices, particularly "Unit Standards" used for assessing fairly basic ICT skills. Focussing on Unit standards and reliance on these for credits led to a lack of deep learning (for example, this is discussed by Petrie, 2018).

6. Integrating Computer Science/Informatics across other subjects in school curricula has been ineffective.

Some high schools that have embraced integration have reported success through rich and deeper learning based on authentic contexts and themes. Integration can work well *if* a digital specialist works with teachers in other subjects (see Jellyman, 2015)

Also, integration is natural at primary school because the same teacher will be covering most subjects, and they can see connections with other areas, such as literacy and numeracy. Integration is more of a challenge at secondary schools: at this level integration is often associated with "innovative learning environments", which are viewed with suspicion by some teachers; also, some school structures can cause subjects to end up in silos.

7. Teacher professional development in a newly introduced Computer Science/Informatics subject is a challenge in quality and quantity.

These problems have been grouped into six parts:

1. There is a need for more teacher educators (PLD providers) to have a deep knowledge of Computational Thinking as well as adequate skill in teaching how to teach programming, along with relevant pedagogies and skills to facilitate professional learning and development. The challenge is for such people to upskill while there is an immediate demand for support. There is also a risk that the experts who review schools will lack depth in their knowledge and experience.
2. In 2017 NZ was short of professional development opportunities; one response was a rapid expansion of short courses, resources and services stimulated by funding from the Ministry of Education. This included a large "readiness" programme called "Kia Takatū ā Matihiko", which includes a range of online and face-to-face course for practicing teachers, including developing communities of practice to support each other.
3. The issue is different in preservice teacher education. It is often only one year for graduates and thus a packed curriculum, within which many aspiring teachers must also learn Māori language and pedagogy. Colleges of Education find it difficult to create space in their limited timetables for the extra material, and so they face similar issues to schools to try to implement the change. As for PLD providers, there is a need for increased depth of knowledge among preservice teacher educators and the teachers who work with student teachers when they are placed in schools.
4. As teachers upskill for the new curriculum there is a risk that they will take advantage of their new skills and move to industry. We are aware of the occasional case where this has happened, but the reverse has also happened - some industry professionals with a passion for teaching have seen the new curriculum, and have moved into teaching as a result. Others have helped with PLD (including companies providing some staff time as a social good donation to support schools).
5. Professional development and support is also needed for school leaders to support this change, including school principals and their governing boards. Without this, sometimes non-specialist teachers have be expected to be able to cover the new material without adequate support, and some schools haven't offered the time or understanding to support teacher professional development because of not understanding the scope of the change. Through various ways of reaching out to school leaders this is gradually changing, or though it may not be until schools are evaluated for their engagement with the curriculum that some leaders pay attention to it.
6. The Digital Technologies Teachers Aotearoa (DTTA; formalery NZACDITT) is national association with the goal of advocating for these subjects. It has a community of teachers who share resources, communicate and speak with one voice to get our subject area recognised and

supported. DTTA has had a significant role in supporting teachers in local groups, through informal PLD as well as more organised events.

8. Identifying and allocating the additional resources for teaching Computer Science/Informatics is a challenge.

The Ministry of Education has recognised this and funded the development of sample teaching material and exemplars to provide clear links with the progress outcomes and Digital Technologies topics.

School students who do not have access to trained teachers can study this subject online through several options, including NetNZ. The subject is also offered nationwide by Te Kura, which began as NZ's correspondence school in 1930s. There is also some curriculum support through a a set of online modules in OER provided by Auckland University's 'STEM Online'.

However, NZ is a bicultural nation so it is important that the subject is available from a perspective of te ao Maori (a Maori world view), so that teachers and students can relate to it. Therefore there is a dual curriculum (NZC and TMoA). It should also be noted that biculturalism is expected in English Medium too, see Gaddam et al (2018). Maori terminology is still being developed for computing and more resources are needed, including examples of good teaching and learning practice with exemplars for progress outcomes.

New challenges and suggested actions

2) Are there any new challenges (not identified below) that you have identified as being particularly important in the situation in your country?

As noted above, Aotearoa New Zealand is a bicultural nation and there is a need to respect and live with an indigenous world view. We see it as a strength as well as a challenge.

It is something that is particularly important to develop in this the international year of indigenous languages @IYIL2019.

3) If we were to produce some kind of report/guidelines/curriculum recommendations on behalf of TC 3 and UNESCO what do you think might have been or might still be useful in your country for making decisions about the Computer Science/Informatics curriculum?

Two suggestions:

1. At a global level it could be valuable to have a report on the ways in which this curriculum has been/could be integrated within initial teacher education, including strategies for upskilling the workforce of teacher educators and those who provide quality assurance for schools. This should also include UNESCO prioritised countries that have recently begun to work on policy development in this area, such as Ghana where the focus is currently on ICT (Kofi Ayebi-Arthur, Personal Communication).

2. Strategies and resources incorporating indigenous world views on this curriculum, teaching and related professional development etc. This could also be a way of raising awareness of strategies

that this country is aiming to develop in its future though the national curriculum (Ministry of Education (2007, page 10),

> Students will be encouraged to value:
>
> • excellence, by aiming high and by persevering in the face of difficulties;
>
> • innovation, inquiry, and curiosity, by thinking critically, creatively, and reflectively;
>
> • diversity, as found in our different cultures, languages, and heritages;
>
> • equity, through fairness and social justice;
>
> • community and participation for the common good;
>
> • ecological sustainability, which includes care for the environment;
>
> • integrity, which involves being honest, responsible, and accountable and acting ethically; and to respect themselves, others, and human rights

Useful sources

Bell T. (2018) Computer science in K-12 education: The big picture, *Olympiads in Informatics,* 12, 3-11. http://dx.doi.org/10.15388/ioi.2018.01.
Bell T. & Vahrenhold J. (2018) CS unplugged—How is it used, and does it work? *Lecture Notes in Computer Science* (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 497-521. http://dx.doi.org/10.1007/978-3-319-98355-4_29.
Bell T.C. & Duncan C. (2018) Teaching Computing in primary schools. In Sentance S., Barendsen, E. & Schulte, C. (Ed.), *Computer Science Education : Perspectives on Teaching and Learning in School.* London: Bloomsbury Academic.
Curzon P., Bell TC., Waite J. & Dorling M. (2018) Computational thinking. In Fincher S; Robins A (Ed.), *The Cambridge Handbook of Computing Education Research*. Cambridge: Cambridge University Press.

Gaddam, A., Lundqvist, K., & McRae, H. (2018). Culturally Relevant Approach to Encourage School Children Learn Computer Science Concepts. Using Game-based Education Strategy for Student Learning Effectiveness and Motivation. In *Proceedings of the Sixth International Conference on Learning and Teaching in Computing and Engineering* (LATICE), Auckland, New Zealand.

Jellyman, P. (2015). *Models of Curriculum Integration in New Zealand Secondary Schools Sabbatical report, Term 2, St Dominic's Catholic College.* Retrieved from

http://www.educationalleaders.govt.nz/content/download/73195/601556/file/Philip%20Jellyman%20Sabbatical%20Report%202015%20-%20Models%20of%20Curriculum%20Integration.pdf

Petrie, C. (2018). *Creative thinking and the new Digital Technologies curriculum*. NZARE Blog post. Retrieved from https://nzareblog.wordpress.com/…/…/07/creative-thinking-dt/

POLAND

A perspective from Poland on the introduction of Informatics into schools

by Maciej M. Sysło

UMK Toruń, University of Wrocław, WWSI Warszawa, Poland, syslo@ii.uni.wroc.pl

Introduction

The changes in the informatics education for all grades in K-12 have been the subject of nationwide proposals and discussions in 2014-2016 and finally the new informatics curriculum has been approved by the Ministry of National Education and introduced to K-8 in September 2017 and will be introduced to high schools, including vocational schools, in September 2019.

The curriculum is a separate document for each school level (1-3, 4-6, 7-8, 9-11+ext.) however **Unified aims**, which define five knowledge areas in the form of general requirements, are the same in all curricula. The most important are the first two aims and their order in the curricula: (I) **Understanding and analysis** of problems based on logical and abstract thinking, algorithmic thinking, and information representations; (II) **Programming and problem solving by using computers** and other digital devices – designing algorithms and programs, organizing, searching and sharing information, using computer applications. The content of each aim, defined adequately to the school level, consists of detailed **Attainment targets**. Thus, learning objectives are defined that identify the specific informatics concepts and skills students should learn and achieve in a spiral fashion through the four levels of their education. At each level the implementation of the curriculum varies across three elements – the first element is more important at lower levels and elements 2 and 3 become more important during progression: (1) problem situations, cooperative games, and puzzles that use concrete meaningful objects – discovering concepts, heuristics; (2) computational thinking about the objects and concepts – algorithms, solutions; (3) programming, moving from visual/block to text-based environment, including program testing and debugging. For benefits of such a spiral curriculum see: Chapter Structuring the Curriculum[5].

Key Challenges

1. Today in Poland, informatics is widely recognized, by the society, politicians, and decision makers as an academic discipline (in science and among technical disciplines); as a result we observe that informatics is the most popular study among high school graduates (twice as many graduates choose informatics than the next popular subject – management).
2. see 1.
3. Computational thinking is a new term/concept for almost all teachers, also for academic teachers, and teachers of teachers. Although our curricula, especially attainment targets, address and focus on abstraction, generalization, decomposition, algorithmic thinking, programming, and program testing and debugging, and problem solving and we want our teachers and students to think computationally, we avoid to use this term mainly in the curricula for initial school levels. Most teachers, especially in lower levels of education, are afraid of new ideas/terms in pedagogy, even if a new name is used to describe what they have been doing so far as happens with using computers. We hope teachers and students will develop the meaning and their understanding of computational thinking through properly chosen problems and environments, but without losing time on non-productive discussions about its meaning

[5] Webb, M., T. Bell, N. Davis, Y. J. Katz, N. Reynolds, D. P. Chambers, M. M. Sysło, A. Fluck, M. Cox, C. Angeli, J. Malyn-Smith, J. Voogt, J. Zagami, P. Micheuz, Y. Chtouki and N. Mori (2017). Computer Science in the School Curriculum: Issues and Challenges. Tomorrow's Learning: Involving Everyone. Learning with and about Technologies and Computing. WCCE 2017, IFIP Advances in Information and Communication Technology , Cham, Springer International Publishing.

Informatics *versus* **programming**. In the new curriculum and especially in its implementation we avoid 'the equation': informatics = programming, which in effect may kill interest in informatics among school students. Informatics should be taught independently of specific application software and programming languages and environments. Programming is a tool in the process of developing concepts and computational thinking while approaching and solving problems from various areas and school subjects. We convince our teachers and students to: "First think computationally then program", and as a counterpart of "The purpose of computing is insight not numbers (R.W. Hamming, 1959) we declare that: "The purpose of programming is abstraction not programs" since every program is an effect of mainly abstract thinking on problem situation to be solved.

Recently, the "Hour of code" and code.org initiatives became very popular among children and young students. We are very happy that hundreds of thousands of students in Poland each month spend some time practising coding in the environments they know from games they used to play. It is a fantastic introduction to programming but many teachers do not go any further with their students and are satisfied with the codes students produce in these environments. Unfortunately, most of our teachers cannot read educational materials available in these environments. We are now working on providing them with versions in Polish, which additionally will put coding in the context of informatics education.

Programming is not only the way to communicate with computers. Programming can enhance students' problem solving skills in a constructivist way by constructing programs as 'real objects' and then using them in learning by doing (solving problems, making experiments with data, verifying hypothesis, proving statements).

4. –
5. The new curriculum has been proposed by a team of school and academic teachers and also discussed in academic circles to meet their expectations concerning preparation of graduates for higher education. Moreover, the curriculum team also supervised versions of the informatics curriculum for various specializations in vocational high schools.
6. Integrating ICT (informatics) across other subjects in school has been ineffective in practice. Our study shows that this is mainly due to: (1) the lack of integration of informatics with other subjects in the subjects' curricula and (2) the lack of basic informatics knowledge among teachers of other subjects.

   There are a number of steps we undertake to change and improve this situation: (1) students learn informatics by solving, with computers, problems coming from various disciplines (subjects)while , at the same time, developing their knowledge of those subjects and this also contributes to integration of informatics with other subjects; (2) in supporting this integration students may use their informatics knowledge and skills when learning other subjects; (3) we plan to provide teachers with guide books on integrating informatics with mathematics, physics, biology, etc. demonstrating contribution of informatics to other disciplines (school subjects in the curriculum) – such books could be used in project based learning across informatics and other subjects curricula.

7. We have published education standards for informatics teacher preparation which include modules on subject, pedagogy, and technology competences. Moreover, the standards focus

also on teachers' engagement in professional development – candidate teachers come from various pedagogical and subject areas and they need personalized professional development and training. We have also developed a certification procedure, which evaluates the classroom teacher's preparation for effective and successful managing of learning informatics by her/his students.

Special efforts are put to prepare teachers of initial education (in 1-3) who usually graduate in pedagogy and have only little background in ICT: (1) a new module on informatics will be proposed for future teachers in 1-3 and (2) the government devoted 124 ml. PLZL (30 ml. Euro) for in-service training of 5000-7000 teachers from small cities and rural areas.

8. Happily, the education system in Poland is in some sense "ready" for the new informatics curriculum. In particular, (1) informatics has been taught in schools in Poland (under various names) for more than 30 years and recently as a separate subject on each level of education in K-12; (2) the previous curriculum approved in 2008 included computer related subjects on each level, therefore schools employ teachers who teach such subjects – they only need now extra in-service training to meet the requirements of the new curriculum, especially on algorithmics and programming; (3) all schools have been equipped with basic hardware and software and are connected to the Internet; more and more software systems and packages are open and free for education, in particular, almost all programming environments can be freely accessed from school and home equipment; (4) the most important and encouraging is the enthusiasm and readiness of school students on all education levels to learn how to program and use programming skills in various subjects and environments, such as robotics, games, informatics and ICT competitions, and to enhance competencies in informatics.

References

Fluck, A., M. E. Webb, M. Cox, C. Angeli, J. Malyn-Smith, J. Voogt and J. Zagami (2016). "Arguing for Computer Science in the School Curriculum." Education Technology and Society **19**(3): 38-46.
Kemp, P., M. G. Berry and B. Wong (2018). The Roehampton Annual Computing Education Report, London: University of Roehampton.
The Royal Society (2012). Shut down or restart? The way forward for computing in UK schools. London, The Royal Society.
The Royal Society (2017). After the reboot: computing education in UK schools. London, The Royal Society.
Webb, M., N. Davis, T. Bell, Y. J. Katz, N. Reynolds, D. P. Chambers and M. M. Sysło (2017). "Computer science in K-12 school curricula of the 2lst century: Why, what and when?" Education and Information Technologies **22**(2): 445-468.

Appendix 1 EDUsummIT / IFIP TC3 Curriculum Report

EDUsummIT Thematic Working Group 9 and IFIP TC3 Curriculum Task Force

Curriculum - Advancing Understanding of the Roles of Computer Science/Informatics in the Curriculum

Summary Report and Action Agenda (updated July 2016)

Mary Webb, King's College London, UK

Andrew Fluck, University of Tasmania, Australia

Margaret Cox, King's College London, UK

Charoula Angeli, University of Cyprus, Cyprus

Joyce Malyn-Smith, Education Development Center, USA

Joke Voogt, University of Amsterdam, Netherlands

Jason Zagami, Griffith University, Australia

Introduction

This short report, written by EDUsummIT[6] Thematic Working Group 9, results from a series of discussions, over several years, led by the IFIP TC3 Task Force: "Curriculum - deeper understanding of roles of CS/Informatics", as well as from discussions before, during and after EDUsummIT 2015. The report was originally published in 2015 (Webb et al., 2015), and was updated in July 2016, incorporating elements from discussions in the IFIP TC3 Task Force and a symposium at the IFIP TC3 SaITE Conference (Webb et al., 2016).

At EDUsummIT 2015 it was argued that the major rationales for including Computer Science as a subject in the K-I2[7] curriculum are economic, social and cultural. The economic rationale rests not only on the need for a country to produce computer scientists to sustain a competitive edge in a world driven by technology but also on the requirement for Computer Science-enabled professionals in all industries to support innovation and development. The social rationale emphasises the value in society of active creators and producers rather than passive consumers of technology. Such capability provides people with power to lead, create and innovate within society and therefore is also an issue of entitlement to "powerful knowledge" (Young, 2013) giving individuals opportunities to choose their role in society. The cultural rationale rests on enabling people to be drivers of cultural change rather than having change imposed by technological developments. These arguments are presented in detail in Fluck at al. (2016).

In this brief report we explain firstly the background to recent curriculum changes and the global context in which they are taking place. Then we explain the issues and challenges for establishing and maintaining the roles for Computer Science in curricula for K-12. Next we present solutions and recommendations for policy

---

[6] EDUsummIT is a global community of researchers, policy-makers and educators committed to supporting the effective integration of Information Technology (IT) in education by promoting active dissemination and use of research.
[7] K-I2 refers to kindergarten through to Grade 12 (aged 4-19 years).

makers, educators, industrial partners and researchers and finally we outline our actions for taking forward these solutions.

Background and Context

Previous EDUsummITs examined broadly the skills needed in a world driven by technology. EDUsummIT 2011 identified the importance of new competencies for 21st-century learning, including digital literacy, in the curriculum (Voogt, Erstad, Dede, & Mishra, 2013). EDUsummIT 2013 examined computational thinking as a critical set of thinking skills of equal importance to literacy and numeracy in the education of all (Voogt, Fisser, Good, Mishra, & Yadav, 2015). This importance of computational thinking was highlighted by Wing (2006), especially in the context of Computer Science, although the concept is generally attributed to Papert (1980, 1996) who examined computational thinking in relation to learning mathematics and other subjects. Subsequently Barr and Stephenson's (2011) work led to an operational definition published by the Computer Science Teachers Association (CSTA) incorporating the following core characteristics:

- "Formulating problems in a way that enables us to use a computer and other tools to help solve them;
- Logically organizing and analyzing data;
- Representing data through abstractions such as models and simulations;
- Automating solutions through algorithmic thinking (a series of ordered steps);
- Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources;
- Generalizing and transferring this problem solving process to a wide variety of problems." (http://csta.acm.org/Curriculum/sub/CurrFiles/CompThinkingFlyer.pdf).

The recent curriculum developments that formed the impetus for the current report include a review of the information and communications technology (ICT) curriculum in the UK (The Royal Society, 2012) that identified a need for major reform and similar calls in the United States (Wilson, Sudol, Stephenson, & Stehlik, 2010; Cuny, 2012), Canada (http://cacsaic.org/HowAlbertaGotCS), throughout Europe (Joint Informatics Europe & ACM Europe Working Group on Informatics Education, 2013), Australia (http://www.australiancurriculum.edu.au/technologies/digital-technologies/rationale) and New Zealand (Bell, Andreae & Robins, 2012). These initiatives emphasise refocusing Computing/ICT education to incorporate Computer Science as the underlying subject discipline. In these countries where reform was recommended, curricula that had previously contained Computer Science had become weakened or diverted by other priorities. In other countries, such as Israel and Cyprus, Computer Science has been retained since its emergence in the 1980s. However, even in countries that have maintained Computer Science, rationales for its presence in curricula and curricula themselves vary (Hazzan, Gal-Ezer, & Blum, 2008; Webb, Davis, Reynolds, & Sysło, 2015) and are worthy of scrutiny.

The economic imperative is a strong driver behind the recent refocusing on Computer Science in the USA and Europe and focuses on the benefits of people being able to apply technological solutions and entrepreneurial talent to solve problems and promote sustainable development (Joint Informatics Europe & ACM Europe Working Group on Informatics Education, 2013; The Royal Society, 2012; Wilson et al., 2010). At the same time many educators see the importance of people developing understanding of the capabilities and potential of technologies and being able to engage in "computational thinking" and other forms of thinking promoted by studying Computer Science, such as systems thinking, in order to support learning and create informed and responsible citizens. This new thinking and understanding is not the digital literacy whose importance is already well-established but a set of skills, understanding and thinking that can be developed by engaging with and understanding Computer Science; understanding how computers work and designing and creating computer-based solutions, including through programming.

Key Challenges and Issues

Table 1 summarises the main challenges and their solutions identified at EDUsummIT 2015 and indicates recommendations for policymakers (P), educators (E), industrial partners (I) and researchers (R). Evidence of these challenges and justification for solutions is outlined below.

Table 1 Challenges and Solutions for Advancing Understanding of the Roles of Computer Science/Informatics in the Curriculum

Challenge	Solution/Recommendation to P, E, I, R
1. Lack of clear understanding (outside the field of Computer Science) of Computer Science/Informatics as an academic discipline.	(a) Adopt a globally agreed statement of Computer Science/Informatics as a discipline in its own right (P, I, R, E). (b) Articulate the nature, importance and relevance of Computer Science/Informatics to society and education (P, I, R & E).
2. A need for Computer Science/Informatics as a distinct subject in school curricula is controversial and poorly understood.	Disseminate and communicate a clear rationale to different stakeholders about the need to have Computer Science/Informatics as a distinct subject in school curricula (P, I, R & E).
3. Computational thinking, a core component of Computer Science/Informatics, is considered to be important in 21st century skills, but due to its complexity, it is difficult to implement in schools.	Promote computational thinking through the means of a Computer Science/Informatics curriculum, which aims at making computational thinking commonplace (P, R & E).
4. The development of Computer Science/Informatics school curricula is impeded by insufficient empirical evidence of student learning in order to support content definition and sequencing.	Design Computer Science/Informatics curricula based on a content analysis, and then continue to research students' learning difficulties as well as the effects of different pedagogical approaches. (E & R).
5. Previous ICT curricula deliveries poorly prepared students for Computer Science/Informatics in further/higher education or professional employment.	Facilitate better smart partnerships between education systems and industry/professional associations. (E & I)
6. Integrating Computer Science/Informatics across other subjects in school curricula has been ineffective.	Identify clear learning outcomes, assessments and standards for Computer Science/Informatics. (E, I, P & R)

Challenge	**Solution/Recommendation to P, E, I, R**
7. Teacher professional development in a newly introduced Computer Science/Informatics subject is a challenge in quality and quantity for many countries.	**a) Encourage more Computer Science/Informatics graduates to become teachers. (P, I & E)** **b) Add a Computer Science/Informatics specialisation to pre-service training for primary school teachers. (P & I)** **c) Make Computer Science/Informatics professional learning a requirement for periodic teacher re-accreditation/licensing. (P)** **d) Schools need resource allocations to free teachers to undertake the professional learning and preparation for a new Computer Science/Informatics subject. (P)**
8) Identifying and allocating the additional resources for teaching Computer Science/Informatics is a challenge.	**(a) Some of Computer Science/Informatics can be taught without computers. But computers especially mobile devices can enhance the learning experience. (P, E, I & R)** **(b) Teacher training needs to provide skills in using the available resources in the most efficient way. (E)** **(c) Identify, and if not available, commission teaching support materials in mother-tongue language especially for younger students (P,E,I).**
Key: Policy maker (P), Educator (E), Industry partners (I), Researcher (R)	

1. Lack of understanding of Computer Science/Informatics as an academic discipline

The nature of Computer Science as an academic discipline is relatively invariant across cultures, and a common definition[8] is widely accepted. However, developments in Computer Science and consequent technological innovations are rapid, making it difficult for people to understand their importance and relevance. This rapid change also mitigates against clear understanding of the importance of Computer Science in the K-12 curriculum.

2. A need for Computer Science/Informatics as a distinct subject

The major rationales for including Computer Science in the K-12 curriculum as a distinct subject were outlined earlier in this paper as were the characteristics of computational thinking. Our recommendation for

---

[8] This is a definition of Computer Science, which is widely accepted: "the scientific and practical approach to computation and its applications. It is the systematic study of the feasibility, structure, expression, and mechanization of the methodical procedures (or algorithms) that underlie the acquisition, representation, processing, storage, communication of, and access to information."[Wikipedia] It seeks to answer the following questions: What is information? What is computation? How does computation expand what we know? How does computation limit what we can know? (Denning, 2007).

Informatics, which is a term used widely in education in Europe, incorporates Computer Science but is broader and encompasses the entire set of scientific concepts that make information technology possible (Joint Informatics Europe & ACM Europe Working Group on Informatics Education, 2013).

promoting computational thinking through learning Computer Science is based on the understanding that computational thinking is integral to problem-solving approaches in Computer Science (Wing, 2006). Furthermore, working with concepts from Computer Science and programming, which is not usually taught in other curriculum subjects, provide very practical ways to engage with computational thinking even for young children, for example, programming a robotic toy (such as the "Beebot") to follow forward/left/right movements to reach a destination. Computational thinking and other skills developed in Computer Science, including programming, can then be applied to other curriculum areas (see, for example, Barr & Stephenson, 2011).

3. *Computational thinking is difficult to implement in schools*

As argued above, computational thinking involves developing ways of solving problems, designing systems, and understanding human behaviour that draws on concepts fundamental to computer science. However, large-scale studies of computer practices in schools (e.g. Voogt & Knezek, 2008; Plomp, Anderson, Law & Quale, 2009) and earlier reviews (Cox & Abbott, 2004) have shown that within the current school curriculum teachers neither have sufficient time nor expertise to implement computational thinking. The addition of Computer Science as a distinct subject would lead to expert teachers in the field and the curriculum time for computational thinking to become an integral part of every child's education. In Angeli et al. (2016), we elucidate an approach to developing computational thinking skills for K-6 that can motivate students and develop usable skills and confidence through tackling real-world problems.

4. *The development of Computer Science/Informatics school curriculum is impeded by insufficient empirical evidence*

Defining a Computer Science appropriate curriculum structure and sequencing is challenging because there is less evidence of how students develop understanding of Computer Science compared with other subjects. However, there are epistemological considerations and constraints which can guide curriculum design (Winch, 2013; Young, 2013). Furthermore, lessons from curriculum theory and from experiences of curriculum design in other subjects suggest that we need to live with uncertainty and to accept the need for a dynamic and continually renegotiated curriculum (Webb, 2014) and at the same time continue to research how students develop their understanding. In conjunction with specification of curriculum structure and sequencing there is a need for clear learning outcomes, assessments and standards for Computer Science/Informatics throughout K-12. Such clear specification should help to alleviate the problems of confusion that have contributed to the failure of integrated solutions for incorporating Computer Science/Informatics into the curriculum (Wilson, Sudol, Stephenson, & Stehlik, 2010). Therefore, given the additional challenges of teacher professional development, we recommend that in primary education, while it may well be integrated across the curriculum in its delivery, Computer Science is identified as a subject area with specialist support while in secondary schools Computer Science is taught as a separate subject by specialist teachers.

5. *Previous ICT curricula deliveries poorly prepared students for Computer Science/Informatics in further and higher education or professional employment*

Most countries have had national programmes and policies for many years regarding the incorporation and teaching of ICT in schools (Plomp et al., 2009). However, for many years, national curricula in many countries have mainly focused on teaching basic computer skills such as word-processing, using email, drawing a graphics program, communication using Email and Chat and searching for information using the Internet (ibid.) and not the teaching of computational thinking which is a very important 21st century skill. A Computer Science curriculum would have this subject as its core and ensure all school pupils developed competence in it.

6. Integrating Computer Science/Informatics across other subjects in the school curriculum has been ineffective

In spite of enormous growth in the number of computers in schools and widespread access to the Internet, an international study of policies and practices of using ICT across the curriculum in 37 countries (Plomp et al., 2009) has concluded that the integration of ICT use into other school subjects such as science and mathematics, let alone the teaching of Computer Science, has been spasmodic and in many schools non-existent. For example, reporting on the United States in 2009, Andersen & Dexter stated that, “Many of America’s 54 million elementary and secondary school students remain largely unaffected by existing technological infrastructure. Large numbers of teachers and students rarely touch a computer” (Andersen & Dexter, 2009, p. 707).

7. Teachers’ professional development in a newly introduced Computer Science/Informatics subject is a challenge in quality and quantity for many countries

With the shift from ICT across the curriculum incorporated by teachers of a range of different curriculum subjects to a requirement for schools to deliver a CS/Informatics curriculum there has been an urgent need to train existing and new teachers in this subject. To date, in most developed and developing countries, there are insufficient specialist computing teachers to teach this curriculum in all schools (Webb, 2014). In Angeli et al. (2016), we examine the new technological pedagogical content knowledge that teachers need to teach computational thinking, as one element of Computer Science, as well as the challenges for professional development and how they can be addressed.

8. Identifying and allocating the additional resources for teaching Computer Science/Informatics is a challenge

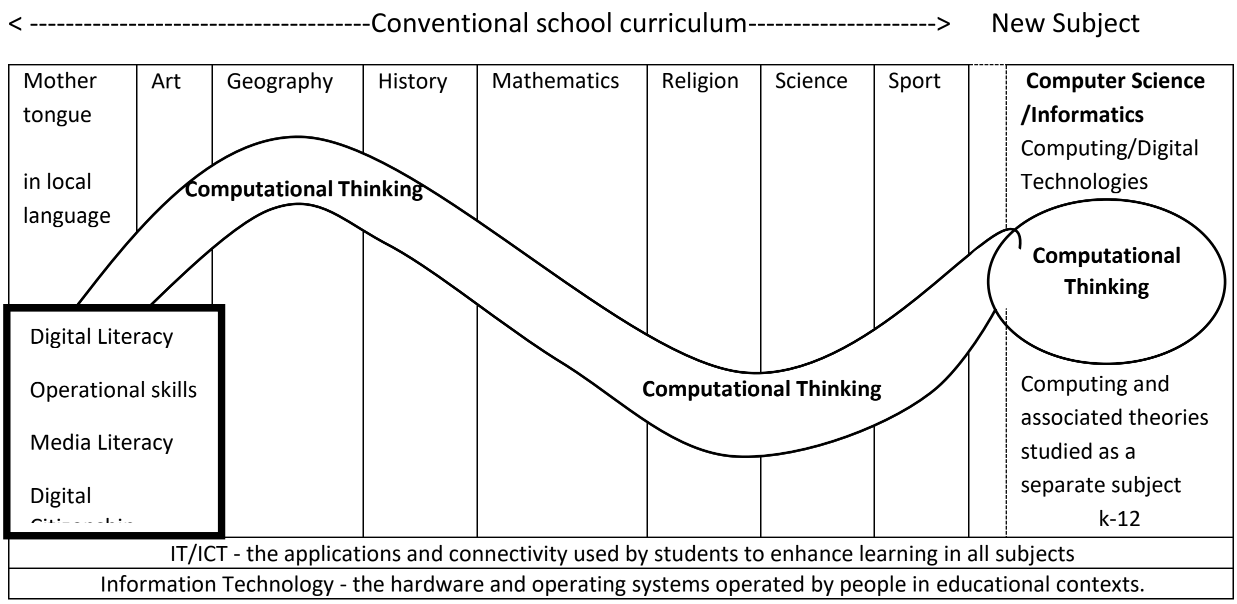
The challenge of providing appropriate resources for teaching Computer Science obviously varies between countries (Plomp et al., 2009) but there are ways of teaching many elements of Computer Science without using computers, e.g. the "unplugged" approach (Bell & Newton, 2013) which can be engaging for students. However, there are possibilities for new software resources for teaching Computer Science enabling improved or accelerated learning of difficult concepts. This is particularly evident in learning programming, which is generally regarded as difficult, but developments of tools for visualising programming, for example, can improve learning provided teachers understand their pedagogical significance (see Ben-Ari, 2013, for a review). Thus it was agreed that resource provision should not be a barrier to introducing Computer Science into the curriculum but making available a good range of resources and ensuring teachers are able to use them efficiently should be an ultimate goal.

While the order of challenges shown in Table 1 represents a logical progression for considering curriculum rationale and design, the order of priority and difficulty will vary across contexts. For example, currently teacher professional development is a major challenge for those countries that are introducing or re-introducing Computer Science.

Critical for global communications about these challenges is the issue of varying terminology used in different countries. Figure 1 offers a mapping of such diversity and the position of the elements of computing-related terms in relation to the conventional school curriculum.

Further issues that were addressed by other EDUsummIT working groups and are relevant to understanding of the roles of Computer Science/Informatics in the curriculum include: 1) the interrelationships between curricula, pedagogy and assessment; 2) the relationship between formal and informal learning; 3) the need to take account of informal learning in assessment. Another issue, at a finer grained level of detail, that concerns policy makers and educators, is the choice of programming languages for learning. We agreed that it is the problem-solving processes and design of algorithms that should take precedence over the learning of specific programming languages. However, it was also agreed that program implementation, testing and debugging is essential and the choice of language affects learning opportunities and pedagogy. Furthermore the predominance of English as the basis for programming languages mirrors the wider tension between schooling in a child’s own language and the importance of accessing international material.

Figure 1 Computing-related terms in relation to the conventional school curriculum



Action Plan

The EDUsummIT working group has completed the following elements of the action agenda from EDUsummIT 2015.

1. Elaborated 2 detailed research papers to address challenges and issues raised in this paper on a)“Arguing for Computer Science in the School Curriculum” (Fluck et al., 2016) and b) “Defining Pedagogical Content Knowledge needed for primary and secondary teachers to teach Computer Science” (Angeli et al., 2016)
2. Disseminated outcomes at various conferences including WERA at AERA 2016 and SaITE 2016

The working group and IFIP TC3 Curriculum Task Force will:

3. Continue to disseminate outcomes at various conferences including WCCE 2017 and the next EDUsummIT 2017
4. Develop a UNESCO Policy Paper on Advancing Understanding of the Role of Computer Science in the Curriculum
5. Create a further research paper to address challenges and issues raised in this paper on the “Challenges for specifying structure and sequence in the Computer Science curriculum: the interrelations between resource issues and pedagogical approaches”
6. Inform national governments of the findings and recommendations outlined in this report

Future work of the IFIP TC3 Curriculum Task Force will include:

7. Addressing challenges and issues raised in this report by a) examining how Computer Science has been incorporated into the curriculum in various contexts and b) examining frameworks that support the development of Computer Science curricula in various contexts
8. Examining how computational thinking can be developed and consolidated within Computer Science curricula and wider curricula contexts
9. Examining the relationship between computational thinking, digital and other literacies and Computer Science.

References

Anderson, R.E. & Dexter, S. National Policies and Practices on ICT in Education: United States. In Plomp, T., R.E. Anderson, Law, N. & Quale, A. (2009). Eds. *Cross-National Information and Communication Technology: Policies and Practices in Education*. Information Age Publishing. Charlotte. USA.

Angeli, C., Voogt, Malyn-Smith J., Webb M. E., Fluck A., Cox M. and Zagami J. (2016). "The Design of the Computer Science Curriculum and the Knowledge that Teachers Need to Teach Computer Science in K-12 " *Education Technology and Society 19(3): 47-57*.

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the Computer Science education community? . *ACM Inroads, 2*(1), 48-54.

Bell, T., Andreae, P., & Robins, A. (2012) Computer Science in NZ High Schools: The First Year of the New Standards. In L. A. S. King, D. R. Musicant, T. Camp, & P. Tymann (Eds.), *Proceedings of the 43rd ACM technical symposium on Computer Science Education, Raleigh, NC, USA (pp. 343–348).* New York: ACM.

Bell, T., & Newton, H. (2013). Unplugging Computer Science. In D. M. Kadijevich, C. Angeli & C. Schulte (Eds.), *Improving computer science education* (pp. 66-81). New York, London: Routledge.

Ben-Ari, M. (2013). Visualisation of programming. In D. M. Kadijevich, C. Angeli & C. Schulte (Eds.), *Improving Computer Science education* (pp. 52-65). New York, London: Routledge.

Cox, M. J. and Abbott, C. (Eds.), (2004) *ICT and attainment – a review of the research literature. Full Report*. Becta Coventry/DfES London January 2004.

Cuny, J. (2012). Transforming High School computing: A Call to Action. *ACM Inroads*, 3(2), 32-36.

Denning, P.J. (2007) Computing is a Natural Science. *ACM Communications 50(7)13-18*.

Fluck, A., M. E. Webb, M. Cox, C. Angeli, J. Malyn-Smith, J. Voogt and J. Zagami (2016). "Arguing for Computer Science in the School Curriculum." *Education Technology and Society 19(3): 38-46*.

Hazzan, O., Gal-Ezer, J., & Blum, L. (2008). A model for high school computer science education: the four key elements that make it! *SIGCSE Bull., 40*(1), 281-285. doi: 10.1145/1352322.1352233.

Joint Informatics Europe & ACM Europe Working Group on Informatics Education. (2013). *Informatics education: Europe cannot afford to miss the boat: Report of the joint Informatics Europe & ACM Europe Working Group on Informatics Education.*

Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas*. Basic Books, Inc.

Papert, S. (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning*, 1(1), 95-123.

Plomp, T., R.E. Anderson, Law, N. & Quale, A. (2009). Eds. *Cross-National Information and Communication Technology: Polices and Practices in Education*. Information Age Publishing. Charlotte. USA.

The Royal Society. (2012). *Shut down or restart? The way forward for computing in UK schools.* London: The Royal Society.

Voogt, J. & Knezek, G. (Eds.) *International Handbook of Information Technology in Primary and Secondary Education*. Berlin, Heidelberg, New York. Springer.

Voogt, J., Erstad, O., Dede, C., & Mishra, P. (2013). Challenges to learning and schooling in the digital networked world of the 21st century. *Journal of Computer Assisted Learning*, 29(5), 403-413.

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*, 1-14.

Webb, M. E., A. Fluck, M. Cox, C. Angeli-Valanides, J. Malyn-Smith, J. Voogt and J. Zagami (2015). Thematic Working Group 9: Curriculum - Advancing Understanding of the Roles of Computer Science/Informatics in the Curriculum. EDUsummIT 2015 Summary Report: Technology Advance Quality Learning for All. K.-W. Lai. Bangkok, Thailand: 60-69, available at: http://www.curtin.edu.au/edusummit/local/docs/edusummit2015-ebook.pdf.

Webb, M. E., C. Angeli, Y. Chtouki, M. Syslo, M. Lokar, P. Micheuz, A. Fluck, M. Cox, J. Malyn-Smith, J. Voogt, J. Zagami, N. Mori and A. Brodnik (2016). Symposium: Roles of CS/ Informatics in the curriculum – why, what, how*? Stakeholders and Information Technology in Education ('SAITE') Conference*. Universidade do Minho, Guimarães, Portugal.

Webb, M. E. (2014, July 1st – 4th, 2014). Considerations for the design of Computing curricula. Paper presented at the K*EYCIT 2014 – Key Competencies in Informatics and ICT*, University of Potsdam, Germany.

Webb, M., Davis, N., Katz, Y. J., Reynolds, N., & Sysło, M. M. (2015 July 1 - 3, 2015). *Towards deeper understanding of the roles of CS/ Informatics in the curriculum*, Vilnius University, Lithuania.

Wilson, C., Sudol, L. A., Stephenson, C., & Stehlik, M. (2010). *Running on Empty: The Failure to Teach K-12 Computer Science in the Digital Age*. Association for Computing Machinery (ACM), Computer Science Teachers Association (CSTA).

Winch, C. (2013). Curriculum Design and Epistemic Ascent. *Journal of Philosophy of Education*, 47(1), 128-146.

Wing, J. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-36.

Young, M. (2013). Overcoming the crisis in curriculum theory: a knowledge-based approach. *Journal of Curriculum Studies, 45(2), 101-118*.

Acknowledgements

Thanks to:

1. Contributors to the EDUsummIT 2015 discussion paper on which this paper drew: Yousra Chtouki, Al Akhawayn University in Ifrane, Morocco; Niki Davis & Tim Bell University of Canterbury, Christchurch, New Zealand; Yaacov J Katz, il Michlala - Jerusalem Academic College and Bar-Ilan University, Israel; Nicholas Reynolds & Dianne P. Chambers, The University of Melbourne, Australia; and Maciej M. Sysło, University of Wrocław and UMK, Toruń, Poland.
2. Other members of the IFIP TC3 Curriculum Task Force for comments and suggestions at various stages

and especially those whose work informed the pre-EDUsummIT discussions.

3. Those who contributed papers to the symposium at SaITE 2016 (Webb et al., 2016): Matija Lokar, University of Ljubljana, Faculty of Mathematics and Physics, Slovenia, Peter Michaud, Alpen-Adria-University of Klagenfurt Institut für Informatik und deren Didaktik Universitätsstraße, Andrej Brodnik and Nataša Mori, University of Ljubljana, Faculty of Computer and Information Science, Slovenia.
4. All attendees of EDUsummIT 2015 for review and suggestions.
5. Attendees at the symposium at SaITE 2016 for comments and suggestions during and/or after the symposium.
6. Members of TC3 Committee for comments and suggestions during and after the TC3 meeting.

Current Membership of TC3 Task Force: "Curriculum - deeper understanding of roles of CS/Informatics"

Mary Webb (convener), King's College London, UK

Pieter Hogenbirk, Helen Parkhurst School, Netherlands

Nick Reynolds, University of Melbourne, Australia

Yaacov Katz, Michlala - Jerusalem Academic College and Bar-Ilan University, Israel

Andrej Brodnik, University of Ljubljana, Slovenia

Ivan Kalas, Comenius University in Bratislava, Slovakia

Maciej M. Sysło, UMK Toruń, University of Wrocław, WWSI Warszawa, Poland

Andrew Fluck, University of Tasmania, Australia

Torsten Brinda, University of Duisburg-Essen, Germany

Sindre Røsvik, Chair IFIP TC3, Norway

Niki Davis, University of Canterbury, Christchurch, New Zealand

Peter Micheuz, Alpen-Adria-University of Klagenfurt, Austria

Monique Grandbastien, University of Lorraine, Nancy, France

Johannes Magenheim, University of *Paderborn*, Germany

Sigrid Schubert, University of Siegen, Germany

Hans Frederik, VU University Amsterdam, Netherlands

Johan van Niekerk, Nelson Mandela Metropolitan University, South Africa.