Towards deeper understanding of roles of CS/ Informatics in the curriculum – position paper (draft 2)

Mary Webb, *mary.webb@kcl.ac.uk* King's College London, UK.
Niki Davis & Tim Bell, *Niki.Davis@canterbury.ac.nz tim.bell@canterbury.ac.nz* University of Canterbury, Christchurch, New Zealand.
Yaacov J Katz , yaacov.katz@biu.ac.il Bar-Ilan University, Israel
Nicholas Reynolds, nreyn@unimelb.edu.au Melbourne Graduate School of Education, The University of Melbourne, Australia.

Maciej M. Sysło, syslo@mat.umk.pl UMK Toruń, University of Wrocław, Poland
Dianne P. Chambers, Melbourne Graduate School of Education, The University of Melbourne.

INTRODUCTION

In the light of a recent resurgence of interest in Computer Science or Informatics as a key academic discipline that is important in the education of all citizens, TC3 as the Education Committee of the International Federation of Information Processing (IFIP) needs to continue to take a lead in this important area by reviewing recent developments, identifying key issues and dilemmas and proposing ways forward. This position paper has been drafted by members of the TC3 Task Force: *Curriculum- deeper understanding of roles of CS/ Informatics* and is intended to take forward this aim by building on previous discussions within TC3 working groups conferences. More specifically the position paper arises from a panel discussion at Vilnius 2015 in which perspectives from five different countries with different traditions of curricula were presented, analysed and their implications discussed with the audience.

Previous discussions at IFIP Conferences since 2012 have established that within the IFIP Community there is strong agreement that new technologies are enabling new ways of thinking in education, new cultures of learning and that curriculum and assessment needs to change to accommodate new opportunities. The need for development of "digital literacy" and the continued development of the use of technologies for learning is undeniable. The relative importance of CS/ Informatics as the underlying academic discipline and the need for all students to develop key understanding, skills and thinking approaches that emerge from Computer Science/Informatics is less clear. This uncertainty is mirrored in curriculum developments across the globe. In some countries a recent resurgence of focus on Computer Science is driven by economic imperative to create more computer scientists. At the same time many educators see the importance of people understanding the capabilities of computers so that they can exploit technologies for their own use as well as take their place as informed citizens. This new thinking and understanding is not the basic digital literacy whose importance is already well-established but a set of skills, understanding and thinking that can only be developed by engaging with: understanding Computer Science; understanding how computers work and designing and creating computer-based solutions, including through programming.

This position paper builds on focus group meetings that took place at the IFIP Conference in Manchester 2012, the World Conference on Computers in Education 2013 in Torun, Poland and KEYCIT 2014, Potsdam, Germany. These meetings highlighted that there are a range of views among professionals working in the area of Computer Science and ICT in education. A general agreement reached in these meetings was that in order to define a vision or framework, which may help to inform curriculum development, we need to define what is the range and scope of the subject and what are the key ideas and subject matter in the field(s) and at the same time explain why these are important for people to learn. In this way we can move towards a vision and rationale for the curriculum and perhaps a framework. Thus key questions emerging are:

1) What is the range of skills and understanding that should be developed?
2) Are such skills and understanding necessary for everyone?
3) At what age should such education commence and to what extent should it be/remain compulsory?
4) What pedagogical approaches are likely to be appropriate?

At the same time we need to acknowledge that curriculum design is complex and reconceptualisations in curriculum theory that started in the 1970s (Pacheco 2012) mean that there is no one theory of curriculum that is commonly accepted and will provide us with a basis for developing our vision. Instead there are a range of philosophical and epistemological considerations as well as factors related to internationalisation and politicisation that affect curriculum decisions (ibid.).

In this position paper we first explain the background to the changes and the discussions that have taken place in IFIP TC3 meetings. In addition, in the background section we outline the terminology to be used in the paper as the terminology itself has been the subject of much discussion and has the potential for confusion. We briefly review curriculum theory to identify important considerations and constraints. We then review developments in 5 different countries where recent curriculum developments have addressed some of the issues under discussion. We compare and contrast the responses of these countries and the reasons for differences.

BACKGROUND

A review of the ICT curriculum in the UK (The Royal Society, 2012) identified a need for major reform that recognises the value of Computer Science as an academic discipline. Similar calls have been made in the United States (Wilson, Sudol, Stephenson, & Stehlik, 2010) and throughout Europe (Joint Informatics Europe &

ACM Europe Working Group on Informatics Education, 2013). These initiatives emphasise refocusing Computing education to incorporate Computer Science as the underlying subject discipline. These calls have led to much debate about what should be included on Computing and/or ICT in the curriculum.

For IFIP TC3 it is important to consider both the specialised informatics curriculum and the wider importance of ICT/Informatics in other curriculum areas. For example IFIP TC3 includes as its aims:

To establish models for informatics curricula, training programmes, and teaching methodologies.

To consider the relationship of informatics in other curriculum areas (http://ifip-education.ning.com/page/aimsscope-of-tc3).

In order to ensure appropriate communication we first need to define the terms we are using as the variation in terminology has been a source of much confusion in relation to Computing/ICT. The Royal Society report (2012) provided some useful definitions based on the situation in the UK in 2012 (see Table 1) and these will form the basis for definitions in this paper with some further clarification as explained below.

Computing
The broad subject area; roughly equivalent to what is called ICT in schools and IT in industry, as the term is generally used.

ICT
The school subject defined in the current National Curriculum.

Computer Science
The rigorous academic discipline, encompassing programming languages, data structures, algorithms, etc.

Information Technology
The use of computers, in industry, commerce, the arts and elsewhere, including aspects of IT systems architecture, human factors, project management, etc. (Note that this is narrower than the use in industry, which generally encompasses Computer Science as well.)

Digital literacy
The general ability to use computers. This will be written in lower case to emphasize that it is a set of skills rather than a subject in its own right.

Table 1: Computing in schools terminology (The Royal Society, 2012 P.5)

Informatics, a term used widely across Europe, is broader than Computer Science, for example the Joint Informatics Europe & ACM Europe Working Group on Informatics Education use the term Informatics to "cover the entire set of scientific concepts that make information technology possible" (2013 P.9). Another term that arises from computer science and is important in many recent curriculum specifications is computational thinking. It would be possible to devote an entire paper to discussing how to define computational thinking but we will use the definition provided by the Royal Society (see Table 2) that that emphasises its relation to computer science but also indicates its broad relevance. In this paper the terminology shown in Table 2, which is largely based on the Royal Society Report, will be used.

Information Technology (IT) – The use of computers, in industry, commerce, the arts and elsewhere, including aspects of IT systems architecture, human factors, project management etc. (Note that this is adopted from the Royal Society Report and is the title of courses in the UK at GCSE and A-level)
Computer Science – The rigorous academic discipline, encompassing programming languages, data structures, algorithms, etc.
Computing – the broad subject area. This is now the title for the new curriculum in the UK
digital literacy – the general ability to use computers
Computational thinking – the process of recognising aspects of computation in the world that surrounds us, and applying tools and techniques from Computer Science to understand and reasoning about both natural and artificial systems and processes.

Table 2: Terminology used in this paper

Focus group meetings at IFIP conferences between 2012 and 2014 aimed to debate the issues with a view to moving towards a consensus about a vision for the curriculum and how to develop a framework for the design of a curriculum for Computing/digital literacy. The debate was enthusiastic and quite wide-ranging and suggested that reaching a clear consensus and way forward was likely to be difficult. Table 3 summarises key ideas that arose together with an estimate of the level of consensus amongst the participants (Webb, 2014).

Key idea/question about Computing curricula	Level of consensus
Computer Science and digital literacy are complementary – both are needed in the school curriculum	High
Need room for flexibility in interpretation	High
What is the importance of Computer Science for general education? – This is important	high consensus that this question is important
Problems of defining terms	Consensus that terminology is important and difficult
We need to develop aware citizens – not necessarily creators but more than consumers	Controversial
Teaching children to be aware, not necessarily how to create from scratch	Controversial

Current trend is a grass roots movement that appears to have joined forces and coordinated. At the heart of it is an understanding that Computing is essential for all children but also a need for opportunities for career paths and citizenship	Fairly high
A set of concepts based on Computer Science should be defined as a basis for the curriculum – some concepts have a long shelf-life whereas others are short-lived	Fairly high
Computer Science is for everyone	Controversial
What are the good practices that are working?	Controversy over whether this is an important question or not?
Towards a curriculum framework: When – from the beginning What – clear examples How – basic principles Who – concerns with teacher training	The key principles of what needs to be decided or agreed.

Table 3: Views that emerged from the WCCE panel discussion (Torun 2013) (based on (Webb, 2014)

The range of views and lack of consensus indicated in Table 3 as well as in other debates about the curriculum for Computing/digital literacy indicate the extent of the challenge of designing a curriculum framework.

What can we learn from general curriculum theory and curriculum studies?

An examination of general curriculum theory as it might relate to considerations for the design of the computing curriculum identified some key questions as: What is learners' entitlement? What is the nature of knowledge in relation to the curriculum? What is the relationship between theory and practice? How detailed should curriculum specifications be? (Webb, 2014).

According to Young (2013), in order to harness the emancipatory capacities of learners, the curriculum should take them beyond their own experience. Thus the goal of the curriculum becomes to define its content in a world in which the entitlement to knowledge is the goal. In this endeavour "powerful knowledge" is key, defined as specialised discipline-based knowledge which is different from the experience-based knowledge that pupils bring to school (Young, 2013). Clearly, as Young argues, this knowledge is not fixed nor is it equally easily identifiable across all subjects but in computer science as in other disciplines there are people committed to creating and evaluating the knowledge base, some of which is relatively stable and other aspects are changing quite rapidly.

According to Winch (2013) curriculum design is about the management of growth of expertise within a subject. Winch argues that gaining a coherent view of this "epistemic ascent" within a subject by identifying schemata that are at least conceptually and normatively sustainable even if they are not yet empirically ratified is a key element in curriculum design. Moreover Winch argues the need to explore the constraints that the conceptual structure of the subject might impose on pedagogically and cognitively coherent schemata of epistemic ascent and then explore the implications of such constraints within conceptualisations of the subject. The constraints identified by Winch include three interrelated issues. First, it is necessary early in a curriculum (e.g. at primary level) to introduce all three major types of knowledge. This is because knowledge of individual propositions implies some understanding of the concepts that such propositions express and this in turn implies a significant ability to understand and make inferences within the subject. This is Knowing How to do something. Second there is a need for a structured approach to progression in learning the basic facts and central concepts of the subject because knowledge is systematic in terms of 1) classification of its various conceptual elements; 2) the relationships between the elements and 3) the procedures required to gain and validate knowledge. Third the kind of knowledge required to expand and manage subject matter requires a profound understanding of the subject including all of these interacting knowledge types. This therefore is not accessible to school students but comes in more advanced studies beyond school. The fourth constraint follows from the third and requires that the relationship between the ways in which pupils learn by simulating procedures for the acquisition of knowledge in their learning and the actual processes of expansion of disciplinary knowledge should be clarified. For example, project work in Computing often involves the systems development life cycle. Winch argues that simulating such procedures may be pedagogically important in developing acquaintance with the knowledge set of the subject as well as building understanding of techniques used in knowledge management. However these simulations should not be seen as simplified versions of expert practice as that might propagate an illusion that high-level design and planning activities are generic and can be used free of the reality of the skills and materials that are needed to execute the plan. Instead it should be recognised that such expertise requires extensive knowledge and is therefore only possible in higher level courses that build upon previous structured development.

Vignettes of Curriculum Development Initiatives
A view from the UK

Review of the ICT curriculum in the UK (The Royal Society, 2012) identified a need for fundamental reform: a major concern was that the curriculum had become unbalanced with too much emphasis on basic digital skills at the expense of deeper understanding of concepts. The new curriculum in England for Computing introduced in 2014 recognises the value of Computer Science as the underlying academic discipline, forefronts computational thinking and expects pupils to "understand and apply the fundamental principles and concepts of computer science" as well as being able to "analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems". This does not mean that all of the "old ICT curriculum" has been dropped completely: students are still expected to be able to evaluate and apply information technology and to be "responsible, competent, confident and creative users of ICT".

Those of us who have been working in Computing Education for many years are delighted by these curriculum changes: in recent years many of the teachers, with in-depth understanding of computer science, have been deeply frustrated by constraints of the school curriculum. So there are now networks of teachers and computer scientists in the UK working very hard to make this new curriculum work and to support teachers in developing their knowledge and pedagogy. We recognise that there are high risks and major challenges to address, in particular: insufficient Computing teachers with the appropriate knowledge of the subject matter; persuading schools to find sufficient time for the new curriculum; ensuring assessment supports and enables exciting and challenging learning in Computer Science and identifying appropriate pedagogical approaches to achieve this.

One of the important and more general issues arising from the experience in England is how to avoid such curriculum degeneration in future. There are several important factors that contributed to the demise of Computer Science in the curriculum including problems with our assessment and accountability system; weak specification of the curriculum leading to too much ambiguity; lack of understanding of most people of the importance of Computer Science. This last factor remains a significant problem: even people in the IT industry are prone to state that Computing is just a practical subject and there is no need for the underlying theory. There is also a current problem with the perception of programming as just coding. As Computer Science educators we need to not only examine the rationale and content of curriculum but also how to communicate the importance of Computer Science to all learners and the general public.

The approach we have taken in England is to specify a strong and challenging curriculum content of knowledge, understanding and skills underpinned by computer science and emphasising computational thinking up to the age of 14. From 14 to 16 students are still expected to develop their understanding of computing but have the opportunity to specialise in computer science or IT as well as continuing to develop their digital literacy. Developing appropriate pedagogical approaches for students of a range of ability remains challenging. On the Teaching London Computing Project (TLC.org.uk), a professional development programme for existing ICT teachers across London, we emphasise the importance of combinations of “unplugged “ activities as well as practical hands-on experience.

A view from New Zealand

Aotearoa New Zealand, especially the region of Canterbury which is home to a number of software companies with a global reach, provides an interesting perspective on Computer Science education. Canterbury is also home to Tim Bell leads a worldwide collaboration to produce strategicly important research and developmetn with online 'text books' that are used world wide, namely Computer Science Unplugged (Bell 2013) and the Computer Science Fieldguide for Teachers, which we also use and develop as part of our postgraduate programme for teachers of Computer Science education. Computer Science disappeared from the high school curriculum for a number of years and was introduced again in 2012 as an examination subject for seniors in high schools (NCEA Digital Technologies standards; NCEA is the summative final examination system in New Zealand) (Bell 2014). The reintroduction for high school only provides evidence that delaying the introduction of computing concepts to the last 2-3 years of school is not effective. The acquisition of programming knowledge and skills appears to be particularly challenging for students, and even more so for teachers. Duncan, Bell and other colleagues in the University of Canterbury are researching when and how Computer Science is best introduced into schools (e.g. Duncan, Bell & Tanimoto, 2014).

Bell and Computer Science colleagues in other New Zealand universities have recognised the importance of professional development for school teachers and their difficulties in acquiring enough knowledge and skills as mature adults (Bell et al., 2013). Therefore we have developed a course for practicing teachers that involves them actively developing curriculum resources and we are looking to develop more professional development opportunities, including most recently a course in programming and an industry placement. This is being developed in response to a national initiative informed by a report from the Institute of IT Professionals NZ (2014) that "proposes the creation of a new Digital Sciences Learning Area, incorporating Digital Literacy-related topics at primary and secondary school levels as well as digital practice, and Digital Technologies focused on Electronics and Infrastructure, Programming and Computer Science and Digital Media" (Institute of IT Professionals NZ, 2014: 5).

Digital Technologies Curriculum in Australia

An Australian Curriculum containing the compulsory subject, Digital Technologies, was about to be launched in 2015. A change of Federal Government and its review of the Australian Curriculum has resulted in a delay or even the possible cancelling of that curriculum component. The reviewers were of the opinion that the curriculum was too hard for teachers and not really necessary for young children. It was proposed that it either be offered only from Year 7 (the first year of secondary schooling) or at Year 9. The State of Victoria has decided independently that the subject Digital Technologies, should be a compulsory component of schooling from the first years of schooling (Foundation) to the last years of compulsory schooling

(Year 10). This curriculum is built on the notion that computing is its own specialised learning area (different from ICT) that needs its own curriculum and a way of assessing achievement through mandated Achievement Standards; something that had not occurred in previous curricula where ICT was seen as integrated and could often be neglected.

A pilot project to investigate the ways in which teachers went about implementing the new curriculum has been reported by the authors (Reynolds and Chambers, 2014; 2015). It indicated that teachers were capable of understanding the complexities of the new curriculum with adequate support. It also showed that teachers adopted three broad approaches to curriculum implementation. The project showed that teachers can build appropriate learning contexts for young children and for older students.

This Victorian Curriculum (ACARA) is built on five Key Concepts of Abstraction, Data collection (representing and interpreting), Specification, Digital systems, and Interactions and impacts (p.26). The use of these five Key Concepts as the basis for the curriculum moves this curriculum from a purely Computer Science curriculum (although it does contain aspects of Computer Science and coding) to one that attempts to address Digital Technologies as a whole discipline, a discipline where the actions and interactions of humans and computers is of as much importance as the specific knowledge and skills required to think computationally. This curriculum assumes that teachers will be competent, or can be supported to competence, to teach this learning area and that students, even as young as five, have the capacity and the right to develop the skills and knowledge required to operate effectively and ethically in a digital world.

A view from Israel

"Computer Science" is a major subject offered in a small but significant number of Israeli high schools that are located at the upper end of the high school technology track. These are usually elite institutions where very talented students study computer science, physics, chemistry and biology (biotechnology) (following Barak, Waks and Doppelt, 2000). However, the mainstream of high school students in public high schools do not study computer science per se as a stand-alone subject but are instructed in technology literacy as a major medium and methodology that contributes to their learning in all the subjects that they study (following Cheema and Zhang, 2013). The Ministry of Education in Israel has a well organised and detailed strategy that directs high schools in both the enhancement of "Computer Science" as a stand-alone high school subject as well as the promotion of "Computer Literacy" as a must for any successful high school student in all subject areas under study ((Naim, 2010).

Computer use in education is one of the most potent and significant means and platforms for instruction and learning in the 21st century (Alsafran and Brown, 2012). It is imperative that teachers develop knowledge and computer skills that allow them to indulge in teaching based on the transferral of knowledge within their sphere of expertise while utilising available digital databases. Thus, teachers need to be less dependent on their own knowledge of subject matter acquired during their pre- and in-service training, and more dependent on their ability to master the technological alternatives that refer them to digital sites where relevant, updated subject matter can be accessed. In addition teachers need to be technologically literate facilitators who enhance their students' ability to become autonomous learners who possess the computer skills necessary for accessing data relevant to the subject matter they are studying. Teachers need to develop psycho-pedagogical strategies that can motivate their students to acquire the necessary technological capabilities that will allow them to accurately and efficiently become autonomous learners with the ability to access as well as master relevant subject matter in all areas of the school curriculum (following Katz, 2014).

A New Informatics Curriculum for All Students in Poland

The new informatics curriculum benefits from our experience in teaching informatics in schools in Poland for almost 30 years – the first curriculum of informatics as an elective subject in high schools was approved by the ministry of education in 1985 and this subject has never been removed from the high school curricula. It happened 20 years after the first regular classes on informatics were held in two high schools in Wrocław and in Warsaw in 1965. Today, informatics is a compulsory subject in middle (7-9 grades) and in high (10-12 grades) schools and it will replace computer lessons (mainly on ICT) in elementary schools (1-6 grades). The new informatics curriculum is also addressed to vocational high schools.

The new unified informatics curriculum is addressed to ALL students in K-12 and its main goal is to motivate students to use computational thinking and to engage in solving problems in various school subjects. Moreover its aim is also to encourage and prepare students from early school years to consider computing and related fields as disciplines of their future study and professional career. To this end, the curriculum allows teachers and schools to personalize learning and teaching according to students' interests, abilities, and needs.

The new curriculum is built by describing concepts, activities and personal goals in five areas: (1) Understanding and analysis of problems based on logical and abstract thinking, algorithmic thinking, and information representations; (2) Programming and problem solving by using computers and other digital devices – designing algorithms and programs; organizing, searching and sharing information; using computer applications; (3) Using computers, digital devices, and computer networks – principles of functioning of computers, digital devices, and computer networks; performing calculations and executing programs; (4) Developing social competences – communication and cooperation, in particular in virtual environments; project based learning; taking various roles in group projects. (5) Observing law and security principles and regulations – respecting privacy of personal information, intellectual property, data security, netiquette, and social norms; positive and negative impact of technology on culture, social live and security. In each of these areas, learning objectives are defined that identify the specific informatics concepts and skills students should achieve in a spiral fashion through the four levels (primary 1-3 and 4-6, middle 7-9, and high 10-12).

Preparation standards for informatics teachers at each school level, teacher evaluation criteria and certificates, teaching and learning materials for students and for teachers accompany the new curriculum. Moreover, all topics in the curricula for other school subject, appropriate for including and using informatics concepts and skills, have been annotated with comments how to apply computational thinking to enhance knowledge and skills in the other subjects.

Emerging Themes

Several important themes emerged from these brief vignettes. First is the question of entitlement and who is the curriculum for? Both in the UK and Poland new curriculum developments have emphasised a computing curriculum for all students starting from elementary school. Australia nearly made a similar change but this has been delayed or cancelled owing to political change. Israel has maintained Computer Science as a curriculum subject for many years but only for very talented students. New Zealand's current curriculum contains Computer Science as a high school subject for seniors.

A second but related question is when do the key knowledge elements need to be introduced? The reintroduction of Computer Science as a high school subject for seniors has highlighted problems associated with this late introduction. Decisions about the early introduction of Computer Science in the UK and Poland were partly based on a view from school teachers and computer scientists in higher education that aspects of computer science, especially programming, require gradual acquisition and development over many years. This is also supported by New Zealand researchers.

A third theme is the content and in particular the balance across the content of computer science, IT, digital literacy and computational thinking. The new Polish curriculum has identified five areas which appear to represent a fairly even balance across these areas of content. The UK curriculum also aims to achieve such a balance although because it is a reaction to previously unbalanced curriculum with insufficient computer science and programming, there appears to be much more emphasis on the latter. Discussions on the Australian curriculum have centred on the issue of how to balance these different areas.

A fourth theme concerns the importance and possible constraints of having appropriately trained teachers. All of the vignettes emphasise the importance of appropriately trained teachers and the challenge that this provides in countries currently engaged in major curriculum change in this area e.g. UK, Australia where there are not enough teachers with appropriate knowledge and expertise. In Australia this has influenced the decision to put curriculum change on hold. In the UK the discussions emphasised the risk of failure owing to inadequate teaching knowledge and expertise but chose to take the risk and attempt to mitigate it rather than delaying curriculum change.

A fifth theme concerns identifying and working to mitigate and avoid systemic issues created by previous curriculum change. These are evident from the UK experience and that in New Zealand. Other emerging themes and issues that are less clear at present include: reasons and drivers for curriculum change and the importance of pedagogical considerations in determining curriculum change.

The themes we have identified from these vignettes also echo some of the key ideas that emerged in earlier discussions within IFIP (See Table 3) as well as theoretical considerations outlined earlier. The themes and related issues are discussed in subsequent sections in relation to these considerations beginning with the issue of entitlement.

Entitlement

Answers to the question of entitlement has varied across the countries considered in the vignettes above. While the UK and Poland have taken the firm view that the computing curriculum is for all, Israel has opted for a segregated model based on students' capabilities. Australia and New Zealand are still considering this question and New Zealand, in particular, is researching pedagogical approaches to examine whether the curriculum could indeed be made accessible to all.

Young's argument, outlined above, is that the curriculum question: what knowledge? is primarily an epistemological one about what should constitute students' entitlement, together with identification of the epistemological constraints on structuring knowledge from the discipline into sequences suitable for different developmental stages (Young, 2013). Learners' entitlement implies entitlement for all and therefore we need to consider: do *all* students need to understand the powerful knowledge in Computing that we have begun to identify?

There are three particularly compelling arguments for the Computing curriculum in compulsory education. First if learners are never introduced to Computing as a disciplinary area and to the knowledgebase and approaches that Computing academics and professionals use, then they are unlikely to be able to determine whether this is for them. This therefore is an entitlement issue. Second, as many in the profession have argued, programming is difficult and it takes many years to learn to program. While programming is only one element of Computer Science, it is an essential element and it is inconceivable that an introductory course in Computer Science would not contain programming. Furthermore, while Computing professionals do not necessarily do the programming themselves, they need to understand essentials of programming in order to undertake a career in Computing. There is a view among Computer Science educators and emphasised in the New Zealand experience, that coming to programming late in students' development is disadvantageous and that if they were to learn some of the techniques, approaches and thinking involved in programming at an earlier stage more of them would be successful. This therefore is both an entitlement issue for individuals looking towards a fulfilling, creative and potentially lucrative career as well as of concern to countries in terms of their economic performance and prosperity. The third argument is based on the ubiquitous nature of Computing: since so much of our lives is dependent upon Computing we need to develop the understanding and skills of Computing necessary to participate in society. Both the Royal Society Report (2012) and the Joint Informatics Europe & ACM Europe Working Group (2013) emphasise individual entitlement, effects on economic prosperity and social aspects in their arguments for redeveloping Computer Science education.

Curriculum content and balance

As outlined previously, the issue of balance in the curriculum can be viewed across computer science, IT, digital literacies and computational thinking. The UK and Poland have incorporated elements of all these in their curricula although the balance is only likely to be clear from more detail and analysis. Australia's deliberations have centred around this idea of balance as well as considering other factors such as students' and teachers' capability. Israel's approach has been to emphasise the importance of "Computer Literacy" for all students and Computer Science for very talented students.

The curricula in the UK and Poland are consistent with the Royal Society's description of the discipline of Computer Science encompassing foundational principles, widely applicable ideas and concepts as well as techniques and methods for solving problems and advancing knowledge as well as a distinct way of thinking and working (The Royal Society, 2012). The Polish curriculum incorporates a whole area on developing social competences including project based learning and taking various roles in group projects. These issues of cooperation, collaboration and communication were discussed during the development of the UK curriculum but were not included in the specified content owing to concerns about assessment.

Key concepts identified by the Royal Society Report were programs, algorithms, data structures, architecture and communication (The Royal Society, 2012). The Joint Informatics Europe & ACM Europe Working Group identified similar concepts but their version, as they explained, were just examples from a much longer list. Therefore there remains a task, perhaps for IFIP TC3, to consider a complete high-level list of concepts for the curriculum. The techniques and methods that the Royal Society Report identified were modelling, decomposition, generalising with algorithms or data, designing, writing, testing, explaining and debugging programs (The Royal Society, 2012). Again the Joint Informatics Europe & ACM Europe Working Group identified similar techniques and methods but they also identified the importance of various intellectual practices such as tolerance for ambiguity (Joint Informatics Europe & ACM Europe Working Group on Informatics Education, 2013). Thus we are seeing consensus emerging from these working groups about the key concepts and techniques of the discipline although perhaps not yet agreement about the importance of more general intellectual practices and social competences.

Curriculum change: risks and drivers

As explained earlier various constraints have been considered in relation to decisions about curriculum change. In particular the availability of appropriately qualified and trained teachers presents a risk that the curriculum innovation might fail. Such failures might inhibit future change. Thus decisions about desired curriculum content need to be followed by pragmatic considerations of what is possible and how much it might cost. Related to these decisions are considerations of the drivers for curriculum change. In the UK concerns over lack of appropriately trained teachers were high but strong drives from industry prompted political action to drive forward change on the basis of economic advantages.

A further set of constraints, which has yet to be researched in depth, is about how the conceptual structure of the subject might impose on pedagogically and cognitively coherent schemata of epistemic ascent (Winch 2013). We have identified that in different countries there are different views about how well students might cope with concepts and approaches in computing.

CONCLUSION AND RECOMMENDATIONS

The brief review of curriculum developments discussed in this article has highlighted issues contributing to complexity as well as tensions and constraints in relation to designing a Computing curriculum. The themes emerging from the five vignettes and their relationship to ideas from curriculum theory provide some pointers for others who are considering curriculum change. A key consideration should be learners' entitlement (Young, 2013) and identifying knowledge which is sufficiently stable to be classified as powerful knowledge. Our analysis suggests that a consensus is emerging with respect to powerful knowledge which includes the key concepts of the discipline of computer science and of the techniques and methods. There is not yet clear agreement about the importance of various intellectual practices such as tolerance for ambiguity, which are broader than computing. There is also no clear agreement about the importance of social competences in relation to computing and whether they are assessable.

Constraints which are affecting decisions about the introduction of new computing curricula include: concerns about students' capabilities and availability of teachers who have appropriate knowledge and skills. Whether these constraints can be overcome depends partly on pedagogical issues including the value of "unplugged" activities and our developing understanding of how students' learn to think computationally and solve problems by developing algorithms.

REFERENCES

ACARA, (2014) http://www.australiancurriculum.edu.au/technologies/digital-technologies/Curriculum/F-10?layout=1

Alsafran, E. and Brown, D.S. (2012). The relationship between classroom computer technology and students' academic achievement. *Research in Higher Education Journal,15 (Mar), 1-19*.

Barak, M, Waks, S. and Doppelt, Y. (2000). Majoring in technology studies at high school and fostering learning. *Learning Environments Research, 3(2), 135-158*.

Bell, T. (2014) Ubiquity Symposium: The science in computer science: unplugging computer science to find the science. *Ubiquity*, March, 1-6. http://dx.doi.org/10.1145/2590528.2590531.

Bell, T. (2014). Establishing a nationwide CS curriculum in New Zealand high schools. *Communications of the ACM*, **57**(2), 28-30.

Bell, T. (2013). Unplugging computer science. In D.M. Kadijevich (ed) *Improving Computer Science Education.* (66-81) New York/Abingdon: Routledge.

Cheema, J.R. and Zhang, B. (2013). Quantity and quality of computer use and academic achievement: Evidence from a large-scale international test program. *International Journal of Education and Development using Information and Communication Technology, 9(2), 95-106.*

Duncan, C., Bell, T. and Tanimoto, S. (2014) *Should your 8-year-old learn coding?* Berlin, Germany: Proceedings of the 9th Workshop in Primary and Secondary Computing Education - WiPSCE '14, 5-7 Nov 2014. 60-69

Institute of IT Professionals NZ. (2014). *Digital Sciences Learning Area A possible model for the future of Digital Technologies education in New Zealand schools.* Wellington: IIPNZ. Retrieved from http://iitp.nz/files/IITP%20Digital%20Sciences.pdf

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.

Katz, Y.J. (2014) The historical relationship between affective variables and ICT based learning and instruction and achievement in the Israeli school system. In A. Tatnall & W. Davey (Eds.) *Reflections on the history of computers in education: early use of computers and teaching about computing in schools*. Heidelberg: Springer-Verlag, 324-338.Pacheco, J. A. (2012). Curriculum Studies: What is The Field Today? *Journal of the American Association for the Advancement of Curriculum Studies, 8*.

Naim, Y. (2010). *Attitudes of elementary school pupils and teachers towards a technology-based learning environment*. Ramat-Gan, Israel: Unpublished M.A. Dissertation, School of Education, Bar-Ilan University.

Reynolds, N. & Chambers, D. P. (2014) Digital Technologies: Teachers interpreting a new curriculum (in French) Revue Internationale de l'Education de Sèvres – International Journal of Education of Sevres *Pédagogie et révolution numérique*. No 67.

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

Thompson, D., Bell, T., Andreae, P. & Robins, A. (2013). *The role of teachers in implementing curriculum changes.* Denver, CO, USA: 44th Association for Computing Machinery Technical Symposium on Computer Science Education

(SIGCSE '13), 6-9 Mar 2013. In Proceedings 245-250. http://dx.doi.org/10.1145/2445196.2445272.
Webb, M. E. (2014) *Considerations for the design of Computing curricula*. Paper presented at the KEYCIT 2014 – Key Competencies in Informatics and ICT, , July 1st – 4th, 2014, University of Potsdam, Germany.
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. doi: 10.1111/1467-9752.12006
Young, M. (2013). Overcoming the crisis in curriculum theory: a knowledge-based approach. Journal of Curriculum Studies, 45(2), 101-118.

Acknowledgements
key presenters at WCCE 2013 are listed below.
Monique Grandbastien, France
Alain Senteni, UAE and Quebec
Mary Webb, England
Nicholas Reynolds, Australia
Andrej Brodnik, Slovenia
Ivan Kalas, Slovak Republic
Bruria Haberman, Israel
Harriet Taylor, USA

Task Force Curriculum - deeper understanding of roles of CS/ Informatics

Task Force membership:

Mary Webb, *mary.webb@kcl.ac.uk* King's College London, UK. (Convenor)
Torsten Brinda, torsten.brinda@uni-due.de
Andrej Brodnik, andrej.brodnik@fri.uni-lj.si
Niki Davis, *Niki.Davis@canterbury.ac.nz* University of Canterbury, Christchurch, New Zealand.
Andrew Fluck, Andrew.Fluck@utas.edu.au University of Tasmania, Australia
Pieter Hogenbirk , P.Hogenbirk@helenpark.nl
Ivan Kalas, Ivan.Kalas@fmph.uniba.sk
Yaacov J Katz , yaacov.katz@biu.ac.il Bar-Ilan University, Israel
Nicholas Reynolds, nreyn@unimelb.edu.au Melbourne Graduate School of Education, The University of Melbourne, Australia.
Maciej M. Sysło, syslo@mat.umk.pl UMK Toruń, University of Wrocław, Poland
Sindre Røsvik