COMPUTER SCIENCE (CS) IN THE COMPULSORY EDUCATION CURRICULUM: IMPLICATIONS FOR FUTURE RESEARCH

Don Passey
Professor of Technology Enhanced Learning
Centre for Technology Enhanced Learning, Department of Educational Research
Lancaster University, Lancaster, UK
Email: d.passey@lancaster.ac.uk; Telephone: 0044 1524 592314

ABSTRACT

The subject of computer science (CS) and computer science education (CSE) has relatively recently arisen as a subject for inclusion within the compulsory school curriculum. Up to this present time, a major focus of technologies in the school curriculum has in many countries been on applications of existing technologies into subject practice (both software such as office applications, and hardware such as robots and sensors). Through uses of these applications, information and communications technologies (ICT) have focused on activities to support subject and topic learning (across wide age and subject ranges). Very recently, discussions for including computers in the curriculum have shifted to a much greater focus on computing and CS, more concerned with uses of and development of programming, together with fundamental principles of problem-solving and creativity. This paper takes a policy analysis approach; it considers evidence of current implementation of CSE in school curricula, the six main arguments for wider-scale introduction of the subject, the implications for researchers, schools, teachers and learners, the state of current discussions in a range of countries, and evidence of outcomes of CSE in compulsory curricula. The paper concludes by raising key questions for the future from a policy analysis perspective.

KEYWORDS

School curricula, computer science, computer science education, national policy, economic drivers, future research.

1. INTRODUCTION

Computers, and technologies with computing facilities, already have a history in education, and their place in education is widely established. Although the subject of computer science (CS) and computer science education (CSE) has only relatively recently arisen within the post-compulsory curriculum, it has been quickly followed by a very wide range of in-depth study and creative application. In 1962, Purdue and Stanford Universities in the United States (US) established perhaps the first departments of computer science; the first Doctor of Philosophy (PhD) degree in computer science was awarded in 1965 by the University of Pennsylvania, also in the US, while a robotic hand was developed in the same year at the University of Belgrade, in Serbia. The introduction of computers into schools started after only a very short time interval; they can often be traced back to the 1980s or earlier (see Tatnall and Davey 2014), when single computing machines, initially running programs from tape cassettes, were introduced into schools in a number of countries across the world. Since that time, computing technologies have become increasingly diverse, both in terms of the facilities they offer (for example, being able to run programs from a hard disk, being able to access resources across the world via the Internet, being able to run and play video games, or being able to locate a geographical position and find directions to another location), and size and mobility (for example, using handheld and mobile devices such as mobile telephones, laptop machines, desktop machines, or large display facilities). It is now increasingly common for individual teachers or learners to possess more than one computing device of their own (perhaps a mobile telephone, a laptop, a Moving Picture Experts Group Layer-3 (MP3) player, and a games console, for example).

The original concerns of policy makers (largely at a national rather than regional or local level) when introducing computers into schools in the 1980s were not focused so much on how computing facilities could support subject or topic learning more widely, but were concerned much more with how teachers and learners could experience computers and computer facilities so that they might come to understand more about those that they would find in future employment situations (Passey 2014). Although this was the key reason for computers being introduced into schools in England at that time, government agencies, research institutions and educational advisors and practitioners quickly saw opportunities and ways for computers to support subject and topic learning that would go beyond the field and subject of computing, CS and programming. From the 1980s onwards, software programs were developed that were designed to enhance learning opportunities in classrooms, across subject areas, for example in mathematics, language and science. This form of development and trend, concerned with subject-supporting resources, has continued to this present time, to the extent that many rich resources are now accessible to teachers and learners, not only within their own local areas, but from worldwide resources.

This paper is not concerned with this shift in focus from early intentions to more recent intentions, but is concerned fundamentally with the current discussions about and focus for school curricula on computing, CS, CSE and programming. However, it is perhaps salient to highlight the fact that the contemporary concerns about a focus on computing and CS are not new. But it can be argued that the context in which this concern is now being discussed is different from that when it happened previously, in the 1980s. Considering the current context, this paper takes a policy analysis approach to the concerns for CS introduction into schools. It will highlight evidence of: current implementation of CSE in school curricula; the main arguments for wider-scale introduction of CSE; the implications for researchers, schools, teachers and learners; the state of current discussions in a range of countries; evidence of outcomes of CSE in compulsory curricula; and will conclude with key questions for future research from a policy analysis perspective. It is argued that these latter questions need to be researched if the outcomes of the current concepts for school CSE curricula are to be effective in meeting the needs of learners and their future employment prospects. In a previous paper (Passey 2015) I focused on the argument for achieving a balance of CS and ICT in curricula in compulsory education. In this paper I want to look in more detail at the reasons for including CS and CSE in compulsory education; in doing this, the policy analysis provides a partly positional paper, while also partly focusing on curriculum analysis and implications of an introduction of CS and CSE into compulsory education.

2. EVIDENCE OF IMPLEMENTATION OF CS IN COMPULSORY CURRICULA

In this paper, the terms computing, computer science (CS), computer science education (CSE), computational thinking (CT), programming, and information and communication technologies (ICT) are used throughout (as defined below), but it should also be recognised that widely different uses of these and other terms arise in research and policy literature. This is particularly true when exploring literature from across a range of countries, and this can lead to difficulties when trying to ensure accuracy of communication of concepts, ideas and practice. This difficulty will become apparent when viewing, for example, the content of Table 1, where authors from different countries outline roles and practices of computing, ICT and informatics in their own countries. Terms have particular definitions in specific contexts, but in the context of this paper, it is the relationship of these terms to implementation within the compulsory education curriculum that is being explored. It is argued later that the nature of this relationship is in itself fundamentally important. Computing (in some countries referred to as informatics), defined often as use and operation of a computer so that data can be handled and processed, implies the need for this subject to be undertaken in the presence of computers. Traditionally perhaps, computing activities have been located in computer laboratories, but increasingly, with the advent of mobile computers, activities are able to be undertaken in a much wider range of locations, inside and outside classrooms. Computer science, a broader term, is concerned with the study of not just the development and use of computers, but also focuses on the principles behind them and their uses. Computer science, therefore, implies that the subject can be implemented in areas sometimes without as well as with computers. Indeed, some of that implementation might be located in areas beyond formal classrooms, in informal (such as home) or non-formal environments (such as in clubs or with interest groups). Computer science education is concerned with the pedagogy, or teaching and learning, of this computer science subject. Information and communication technologies (ICT) is similarly a very broad term, with focus not just on the breadth of technological devices that support information and communication approaches, but also the principles and practices concerned with applications within and across wide subject (such as languages, science or mathematics) and topic (such as vocabulary, nutrition or algebra) areas. Within subject areas, but vitally within the areas of computing and CS, computational thinking (CT) can be considered to be at the heart of the learning; this set of processes is concerned fundamentally with problem solving (indeed, some would go further and say complex problem solving) that can (but does not necessarily) involve programming. CS is concerned also with creativity, through ways individuals approach the application and transference of programming skills and competences to address problems. The relationship between these processes of problem solving, computing and programming is not at all clear within the literature, however. As Voogt et al. (2015) state, "there is a history in both research and [the] popular press of the use [of] programming as a way to develop thinking skills. CT focuses on developing these thinking skills while within subjects beyond computer science, CT does not necessarily require the use of programming nor are CT scholars making the claim that programming has to be the context in which these skills are developed" (p.716). This point is also considered by Lye and Koh (2014) in their review, from a programming, rather than a wider subject, perspective, stating that: "Programming is more than just coding, for, it exposes students to computational thinking which involves problem-solving using computer science concepts like abstraction and decomposition" (p.51).

Voogt et al. (2015), while considering the use of terms, and defining particularly the core skills and competences within the concept of computational thinking, discuss frameworks for how implementation of computational thinking and programming have been devised in the US and England, but indicate the paucity of evidence of how this implementation has been achieved, or what factors might influence it positively or negatively. Frameworks for implementation of computing and computer science certainly exist (as indicated in other parts of this paper in Australia or England, for example), but the number of studies that have explored this from an implementation point of view are scarce. Lye and Koh (2014) review 27 studies concerned with CT through programming across K-12 (the 5- to 18-year-old age range) and HE sectors, with only 9 in the K-12 sector. A separate report (Hubwieser 2012) looks at the implementation of a CS curriculum in gymnasia (selective high schools) in the State of Bavaria, Germany, over a period of some 9 years. A compulsory CS element of the curriculum is introduced for learners aged 11 to 12 years (grades 6 and 7), but CS is then optional as a subject elective for learners from 14 years of age (in grades 9 to 12). In terms of factors affecting implementation, Hubwieser highlights the importance of state-wide policies supporting the introduction, commitment of time within the entire subject curriculum, the training of teachers in the subject of computer science (and an agreement of standards that these teachers should reach), and definition of appropriate knowledge and understanding at each stage of the curriculum. This curriculum in Bavarian gymnasia appears to operate within a very formal environment. As will be argued in the next sections in this paper, evidence of how curricula might work beyond the formal - across formal, informal and non-formal environments - is clearly needed in our current context.

3. THE MAIN ARGUMENTS FOR CS CURRICULA IN SCHOOLS

Most schools and teachers using computing technologies have been concerned over the past 20 years or more with how these facilities can be integrated into subject and topic teaching, and how their deployment can support learning. Teachers have been concerned, for example, with how their learners might gain greater understanding through the teacher's uses of interactive whiteboards, or how the teacher can engage learners in reflective learning through appropriate feedback in electronic form. This focus has been concerned with applications of existing computing technology facilities (both software and hardware), rather than a focus on using the underlying computing facilities themselves, and how they might be developed and used through programming or networking to solve problems.

Very recently, discussions have raised a fundamental issue: compulsory school curricula are not focusing adequately on computing, CS and learner uses of computing that will provide for adequate future needs. These discussions have led to some national curricula (such as those within Australia and England), now requiring a focus on computing and CS rather than on ICT (ACARA 2013; DFE 2013). This shift in focus, and a shift towards mandatory requirements for schools and teachers to focus on computing and CS, appears to be based on six main arguments.

There is an **economic argument**. That is, education should support learners in engaging through a curriculum that is most likely to support a future economy, where young people are able to meet the needs of current and future jobs and their skill requirements. Livingstone and Hope (2011), in a report on the future of the games and visual effects industries in the United Kingdom (UK), highlighted the dire need for more young people to become interested in and aware of prospects available to them in the video and special effects industries. They argued that if these industries are to continue to develop and be fulfilled in terms of employee numbers and skills in the future, young people need to gain computing skills to meet the needs of this form of employment. In the UK, there are parallel wider concerns with ICT skills shortages (see e-Skills 2012). As this report states, "Replacement demand will generate an additional 321,000 job openings in the sector which in addition to the 50,000 jobs created by growth means there is a total requirement of 371,000 between 2010 and 2020. ...Future skill needs in the sector can be grouped into five areas: security skills, business skills, technology specific skills, interpersonal skills and analytical skills" (p.v). In England alone it was estimated that an additional 745,000 ICT workers would be needed between 2013 and 2017 (according to the UK Digital Skills Taskforce 2014), but almost half of employers reported in 2011 that they had encountered difficulties in finding suitable applicants (e-Skills 2012). It is clear that the skills identified here are not simply programming skills, but include those regarded as necessary 21st century skills, such as those identified by Hanover Research (2011) from their review across six different future employment skill need sources - collaboration and teamwork; creativity, imagination; critical thinking; and problem solving.

While these future employment skill areas align closely with those concerned with CS, there is, nonetheless, an important **organisational argument** that is being made, for collaboration and teamwork skills. These skills are already requirements of some university-level CS courses. McDowell (2015), for example, states that "one objective of the computer games degree courses is to prepare learners to enter the games industry, and it is a stated requirement of the programme specification that students should work in small teams to develop a range of computer games" (p.8). Industries and institutions are increasingly engaging and employing learning technologists with CS skills to support their own individual local needs, to develop computing facilities that meet their specific requirements. In higher education (HE), for example, the University and Colleges Information Systems Association (UCISA) recent surveys (UCISA 2010, 2012, 2014) show that HE institutions in the UK are increasingly employing professionals in CS-related jobs – learning technologists, e-learning officers, e-learning advisers, and e-learning staff developers. Fifty-four out of 91 institutions participating in the UCISA 2010 survey reported having a form of learning technology support unit (with a team of people), while 56 had an educational development unit that provided support. Additionally, support was reported to be provided by information technology (IT) support units (rather than by individuals), which were most commonly (from 66 responses) reported in terms of central units providing IT or ICT support. The UCISA survey in 2010 reported that some 11 members of learning technology staff were employed on average by each of the institutions participating in the survey, and that most of these were located in support units, working with and alongside others. The UCISA 2014 survey indicated a rise in the number of learning technologists providing that support:

> *Despite the challenging economic climate and budgetary pressures, which have led just under half the number of responding institutions to restructure or change existing TEL support roles, 34 institutions reported that they had actually increased staffing levels for TEL since the last survey and 38 institutions foresee staff increases in the future. (p.13)*

Learning technologists in universities are not a completely homogenous group; individuals can be asked to undertake quite different tasks, which can range across the CS arena. Nevertheless, many learning technologists need to work with others, even if they are involved in CS or programming practices. Programming data base or web site applications, for example, requires an understanding of how such facilities will be used, and by whom. In some cases, learning technologists need to take strategic roles, but again, this requires working in groups and with others, rather than in isolation. Companies as well as universities and university departments are increasingly employing learning technologists to develop and handle learning management systems to enable online access and to engage in online access or learning managed and administered electronically (see the Bureau of Labor Statistics 2015 with regard to web developers, for example). There is every reason to believe that such a trend will continue rather than wane over the next 20 years.

There is a **community argument**. That is, computing facilities are increasingly being, and will increasingly be, used not only by companies and by individuals for social purposes but also by 'communities'. Social groups or communities, based perhaps on local government or local community groups, or indeed more widely scattered community groups, can have shared interests (for example, bird watching or music groups). Activities undertaken by these community groups will increase the need for some individuals to have and to use computing and CS skills to support not only themselves but also others across their community groups for specific shared purposes (easily evidenced by outcomes of an Internet search for web sites or the use of social media for specific interest groups in philately or genealogy, for example). Such community groups range across the age spectrum. Take, for example, the way that some 'older generation groups' are now becoming linked and engaging in uses of computing technologies to communicate with each other, and to take online courses that meet their own needs and interests (see, for example, the University of the Third Age Australia, with registration at local as well as national level) (U3A n.d.).

There is an **educational argument**. Elements of computing continue to develop, and it is not possible to see an end-point to these developments. With new technological developments and new areas of application being opened up (in robotics or design, for example), there is a clear argument that education should appropriately integrate CS to support awareness, understanding and future societal need. The provision of a CS curriculum offers this form of provision. With computing technologies becoming increasingly ubiquitous, it can be argued that younger as well as older users should have an increasing understanding of, and capabilities to use, the full range of computing facilities that exist, whether these facilities are accessed through programming, or through application. The educational argument is, therefore, concerned as much with lifelong education as with compulsory education. The European Union has identified, for example, digital skills that all citizens should have, if they are to engage fully and effectively with uses of digital technologies (Ferrari 2013). Part of these skills are concerned with computing and CS (for example, "apply settings, programme modification, programme applications, software, devices, to understand the principles of programming, to understand what is behind a programme").

Linked to this argument, it has become well recognised that CS and computing involve certain skills, and indeed that the disciplines are based upon certain fundamental skills and competencies. Skills such as problem solving, collaboration, creativity and logical thinking are often stated as outcomes for those engaging in computer science activities (Kay 1991; McCormack and d'Inverno 2012). Hence, there is a **learning argument**. Current and new facilities require users to have technical, operational and application skills and competencies if they are to use and apply such facilities to support themselves and others. A factor explored in one study - learner perceptions of computing careers - indicated that these were generally regarded to be poor (McEwan and McConnell 2013). Poor perceptions of the value of the subject has been discussed too by teachers, who have stated that they feel that the value of teaching ICT focusing on IT skills and digital literacy is low by comparison to the value associated with teaching computing. With ICT teaching described as being 'dull and unchallenging', teachers and educators concerned with these poor perceptions have argued that computing should be adopted more strongly, with its more highly-regarded associated creative and problem-solving approaches (Royal Society 2012). What concepts or theories of learning might best be applied to the teaching of CSE is, however, debatable. Perhaps the most extensive work in this field was published in the 1980s and 1990s by researchers investigating an experiential approach to learning termed 'constructionism' (see, for example, Papert 1980; Papert and Harel 1991). However, the research in this field was undertaken at a time when computing and resource facilities and approaches to teaching the curriculum were rather different from those existing today, and it should be recognised that these differences might now limit the ways in which such approaches are applicable (and indeed some concern is raised by results from a recent research study discussed in a later section of this paper).

There is, finally, a **learner argument**. It can be argued that learners should be enabled to engage not only in what are considered to be generic areas of future need (such as numeracy and literacy), but also in areas that interest them. Computing or CS is an area that is known to engage and interest some learners (Passey 2012), and it can be argued that for those individuals their engagement in this field should come at a time in their lives when they can potentially see ways in which that interest might shape their future as well as their immediate needs. Indeed, arguments for inclusion of computing and CS in school curricula to support learner interest from the age of 5 years is not uncommon (ACARA 2013; DFE 2013).

While CS provides opportunities for learning involving higher order levels of thinking and skills, as listed above (and discussed also by Grover and Pea 2013, and Kafai and Burke 2013), the subject has not necessarily experienced a wide uptake by learners when courses have been optional. Rather than seeing an increasing uptake of the subject, in the UK there has been a declining uptake of ICT-related subjects both at national General Certificate of Secondary Education (GCSE) examination level (at age 16 years) and at national Advanced- (A-) examination level (at 18 years). According to statistics produced by the Joint

Council of Qualifications (2014), the number of learners taking A-level qualifications in computing and ICT fell by 43% between 2003 and 2011. Similarly, girls taking IT-related courses fell from 47% in 2012 to 44% in 2013 for GCSEs and from 8% in 2012 to 6.5% in 2013 for A-levels (e-Skills and BCS 2014). In terms of such decreasing uptake, the additional role of social background as an influencing factor has been questioned; for example, both ICT use and ICT literacy levels have been found to be low on average in learners from socially disadvantaged backgrounds (The Prince's Trust 2014).

In terms of learner groups, the low numbers of women employed in the ICT sector is highlighted as an issue (and presumably, if addressed, providing a possible solution). The number of women in ICT employment is identified as being far less than the number of men; in 2013, women accounted for 16% of the ICT employment number in England (e-skills and BCS 2014). As the e-skills and BCS report states, "By 2013, of 1,129,000 people working as IT specialists in the UK, less than one in six (16%) were women", "Of the 753,000 people working in the IT sector at this time, just one in five (20%) were women", "In 2013, within the IT sector itself little more than one in ten (11%) IT specialists were women" (p.7). The report goes on to say that the proportion of women "in Higher Education in 2013, ...made up just 12% of applicants and 13% of acceptances" and in secondary education, "females accounted for just 6.5% of those taking Computing A-Level", while "The proportion of females who sat an IT related GCSE in 2013 was 44%" (p.7). These data suggest that there is a major decline in commitment to (and perhaps interest in) IT beyond GCSE (the national examinations at age 16 years) level.

4. IMPLICATIONS ARISING FROM THOSE ARGUMENTS

Accepting the arguments above, it is clear that some shift towards CS within a school curriculum is desirable, and while curriculum practice is being developed in a number of countries, this is happening in rather different ways. A key question is: how to do this effectively, so that schools, teachers and learners are involved in practices that support current and future needs. Effective use and outcome is likely to require an understanding by those implementing this initiative of the arguments on which it is based, and how to consider and address the needs of each element of those arguments.

The evidence currently available about implementation of CS and CSE into school curricula, and the outcomes of CS and CSE in school curricula, is limited. If policy makers, educators, school managers and teachers are to be able to make informed decisions for the future, then there is a need for an evaluation and research strategy that will provide a range of evidence of value to those stakeholders. There is a clear need to:

- Consider how to develop an appropriate strategic evaluation (Passey 1999).
- Identify research studies that will provide short-term indicators and outcomes concerned with aspects of implementation and learning.
- Identify longer-term evaluation studies that will provide indicators of desired shifts over time, concerned with sustained interest, economic and employment concerns.
- Consider how the outcomes of these ranges of studies can inform at national levels, and how these can support a sharing of evidence for the benefits of all.

The six reasons for introducing CS and CSE into the compulsory curriculum should provide us with focal concerns for any strategic evaluation, and the focus of research studies to gain short-term indicators and outcomes (exploring learning, learner and organisational arguments), and those providing longer-term indicators and outcomes (exploring economic, community and educational arguments).

4.1 Research studies to provide shorter-term indicators and outcomes

In terms of the **learning argument**, which is likely to be the immediate priority concerns for teachers, they will require an understanding of the CS skills that should be taught and should be learned. It is easier to identify these in terms of programming; but it is also essential that these skills are considered from the point of view of their context, with appropriate associated soft and higher order skills. The new curriculum in Australia (ACARA 2013) states that it:

> *aims to develop the knowledge, understanding and skills to ensure that, individually and collaboratively, students: are creative, innovative and enterprising when using traditional, contemporary and emerging*

technologies, and understand how technologies have developed over time; effectively and responsibly select and manipulate appropriate technologies, resources, materials, data, systems, tools and equipment when designing and creating products, services, environments and digital solutions; critique and evaluate technologies processes to identify and create solutions to a range of problems or opportunities; investigate, design, plan, manage, create, produce and evaluate technologies solutions; and engage confidently with technologies and make informed, ethical and sustainable decisions about technologies for preferred futures including personal health and wellbeing, recreation, everyday life, the world of work and enterprise, and the environment. (p.2)

Although this description does not explicitly indicate the need for learners to consider associated soft skills, it is clear that 'creating products, services, environments and digital solutions' requires a clear focus on audience, which might well (or perhaps should) involve discussion and collaboration with users so that their needs and requirements are understood and fulfilled. By contrast, the aims of the new national curriculum in England (DFE 2013) can be interpreted at a much more individual learner level, meaning that associated soft skills might well be less likely to be considered:

to ensure that all pupils: can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation; can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems; can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems; and are responsible, competent, confident and creative users of information and communication technology. (n.p.)

Currently less research has looked at the value and outcomes of learning from a CS perspective than from an ICT perspective (for a review of the latter, see Higgins et al. 2012). In terms of looking at how programming and problem-solving are being achieved, with the ability of computing devices to now readily log how learners use them, successes can be more readily monitored, using records of completed tasks, for the individual. A comparison of successes versus non-successes can enable a proportion of success over time to be monitored, and whether this changes in any way over time. In terms of exploring how this might be happening across formal, informal and non-formal learning settings, monitoring time when use is occurring could enable an estimate of whether uses and success rates are related to more formal, non-formal, or informal activities. The days when use is occurring, whether these are in school time, at weekends, or during holidays, can support this form of analysis. From time and day indicators, likely locality can be suggested – school, club, or home. Patterns of use and successes can also be identified by gender, to see whether there are any substantial differences or shifts over time with boys or with girls. If the devices can tag elements of a postcode of the user, then even more specific locality (and potentially socio-economic background) factors can be explored in the analysis of grouping of patterns and successes.

The pedagogic approaches that teachers and others use in supporting learners are likely to be crucial factors in terms of outcomes. In this respect, a recent study (Johnson 2014), which explored how pairs of learners could develop CS skills with game authoring software, highlights well the need to consider very carefully how to conceptualise and manage the use of pedagogic approaches, and concludes by saying of the learners and the management of their learning that:

The wide range in outcomes further suggests that constructionist approaches are not suitable for all learners, especially those who need more guidance and structure. While most pupils in this study had an above average ability profile ..., they did not all display independent learning behaviours or make use of the sources of support made available to them, and this may account for the variation in the games produced. ...their success seemed to have as much to do with their willingness to learn independently as to do with their cognitive ability. This variability in pupils' readiness to learn independently may also reflect the extent to which they had or had not encountered similar project-based activities in other areas of the curriculum.

Constructionist approaches may also not be well-suited to some elements of game authoring. Some aspects of learning, such as the development of graphics software skills, or the learning of programming concepts need, at this level, to be formally taught if they are to be successfully used by all - for these *areas of learning, learning by doing and experimentation alone appear not to be sufficient. Pupils also need to be guided to complete tasks which are not immediately popular, such as planning the game program and object interactions. (p.252)*

Following rigidly a single conceptual approach, such as the form of constructionism described by Papert (1980), may not be sufficient to cover all the learning needs of any class or group of learners. CS activities should include those that consider how to integrate problem-solving approaches as well as those offered through a more didactic programming activity approach. Fulfilling these needs may not in itself be a simple task, and considerable research effort is likely to be needed to support teachers in pedagogical terms.

Considering the **learner argument**, if the curriculum is to enable learners to develop their own interests in CS, whether it is essential for all learners to have highly-developed CS skills is not at all clear, but, what is clear is that learners are enabled to gain what might be regarded as 'life skills' and to take forward their interests, so that CS is provided as an opportunity for all, but that those who have particular interests are enabled to take these interests as far as they are able. The Australian Curriculum, Assessment and Reporting Authority (2013) propose to do this by mandatory integration of CS and ICT up to Year 8 (age 13 to 14 years), with learners choosing optional choices in Years 9 and 10 (age 14 to 16 years).

An important point for any curriculum development to consider fundamentally is the fact that it is reported (not uncommonly) by lecturers that students do not find computing to be an easy subject; students often say that it is 'hard'. There may be different reasons for this, and perhaps the reasons are different for different individuals, and for some there might indeed be multiple reasons, while for others there might be more singular reasons. For example, the need for learned knowledge to be applied or transferred, or for the need to abstract knowledge and practice, or not understanding the underlying principles, may all be possible reasons for this reported difficulty. Indeed, the current curriculum for key stages 1 and 2 (5- to 11-year-old learners) in England (DFE 2013) if anything underlines these very issues, in stating that:

A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world. Computing has deep links with mathematics, science, and design and technology, and provides insights into both natural and artificial systems. The core of computing is computer science, in which pupils are taught the principles of information and computation, how digital systems work, and how to put this knowledge to use through programming. (n.p.)

How this is handled for all learners, rather than those who select a computing course, therefore, is itself a critical question. In the national curriculum in England, the programmes of study indicate what pupils should be taught, but they do not indicate how, or how to address what are known to be issues for learners. Schools will also need to consider how the curriculum might support CS involvement that is not biased towards certain groups, such as boys rather than girls. However, it should also be recognised that while the number of girls going into CS jobs are low by comparison to boys, girls' jobs tend to involve more 'soft' skills in communication while boys' jobs tend to involve more technical skills (Glover and Guerrier, 2010). In this context, Kirkup (2011) states that vocational education and training for ICT jobs might themselves have developed gender- and class-biased occupations, with girls being encouraged to explore areas of 'soft' ICT skills more, so that they end up with what are regarded as employment with lower-level skills. The situation in some other countries is also not dissimilar. In Germany, for example, Kirkup (2011) reports that the proportion of girls in electronics technician training is only 2.5% and in information technology specialist training is only 4.7%. The Tech Partnership (2014) in the UK is supporting schools with possible approaches intended to address these issues. Employers, including Hewlett Packard, British Telecom, Oracle and the National Grid, are supporting an initiative called TechFuture Girls, freely available, designed as after-school clubs to "provide industry-backed challenges for girls aged 10-14. These teach skills such as coding, cyber security, data management and video editing through activities based on girls' interests, including music, sport or fashion" (p.14). Other initiatives run by the Tech Partnership include TechFuture Classroom (curriculum resources based on real industrial projects), TechFuture Careers resources (reported to have been used by over 35,000 students), and 200 volunteers from industry have gone into schools to support a TechFuture Ambassadors programme. A further initiative, TechFuture Teachers, aims to "bring the power of industry collaboration to the benefit of teachers, with work shadowing, weekly webinars from industry, and other opportunities for professional development" (p.14).

Lessons from the past also need to be heeded. For example, an evaluation of a scheme in 2005 to provide after-school clubs for girls to engage them in computing (Fuller et al. 2013) suggested that this had reinforced existing gender stereotypes and expectations, concluding that the initiative was "unlikely to have a significant or sustained impact on what remains an occupational and subject area divided by gender" (p.499). Learning more about the factors that engage different learner groups in CS is clearly important. Schools are best placed to understand their learners; however, evidence of reinforcement of stereotypes and practices suggests that careful review and monitoring is needed in this area if schools are to be successful in shaping future interests in CS.

Considering the **organisational argument**, schools will require an understanding of how CS has been used and integrated into practices in a range of organisations that go beyond 'the programmer in their bedroom or garage'. The fact that CS skills are now increasingly used not alone, but within teams and groups, as in the UCISA examples above, or in the case of video games development (Passey 2012), means that schools, teachers and learners should consider whether and how to develop CS within team work or group situations rather than skills being developed in isolation, individually. In the study mentioned (Passey 2012), the organisation of activities for young people in schools was based on advice from a leading developer in the video game production field. Teams of learners were asked to develop video game levels. The video game creator advised the schools to set up their teams using the range of individuals involved in video game creation: a story lead (to create dialogue, text and scripting); an art lead (to create models and textures); a sound lead (to create speech and special effects); a creative director; and a lead programmer. When the schools in this project set up teams to develop video game levels, individuals worked collaboratively and co-operatively, their different skills and strengths being deployed and shared across the team. Those focusing on CS skills did not do this in isolation; they were integrally involved with the team. In these activities the entirety of soft skills deployed and developed were measured, through self-reported levels of soft skills before, during and after completion of the project. Individual skills involved, and those that developed further across the period of the project, were (according to the teachers and learners themselves): thinking skills; problem-solving skills; researching skills; generating ideas; identifying solutions; making skills; evaluating skills; communicating skills; scripting skills; story boarding skills; sequencing skills; logical thinking skills; artistic skills; team working; planning skills; and leadership skills. In this project, CS skills were being used in an integrated way, and having these skills on their own, developed in isolation, would not only have provided a false view of how the industry organises team working to include those individuals who contribute CS skills, but would also not allow the skills to be easily or efficiently integrated into the entirety of the design and production of the outcome. Future research studies need to identify how shared and team approaches as much as individual skills are both organised and developed. From a curriculum perspective, it will be important to know in which ways those with CS skills can work with others in groups, in pairs, individually, on projects as well as on specific skills.

4.2 Evaluation studies to provide longer-term indicators and outcomes

Considering the **economic argument**, if learners are to understand how their CS curriculum will meet future employment and skill needs, teachers will require an understanding of how CS is affecting employment and economies, and how jobs are increasingly using CS. Whether this understanding can be developed from local, regional, national and international perspectives is a question that should clearly be debated. There is some evidence that is accessible about job changes over time (such as that from the Bureau of Labor Statistics 2012, 2015), but how such evidence is made accessible to schools, teachers and learners is an aspect that is likely to need a much greater level of discussion and development. US data (Bureau of Labor Statistics 2012) show employment areas that have the most likely growth up to the year 2022: industrial-organisational psychologists; personal care aides; home health aides; insulation workers; interpreters and translators; diagnostic medical sonographers; bricklayers and tillers; occupational therapy assistants; genetic counsellors; physical therapist assistants; physical therapist aides; skincare specialists, etc. But, whether this form of evidence can usefully be applied, or even be accessible within a local area, or at a national level, and how this relates to computing and CS, is not at all clear from those data available.

Skill or job shortages tend to be the data that are reported rather than predictions of job growth areas. Future research in this area is clearly needed. At one level it will be important to identify how CS enables a shift in terms of interest and uptake at HE and employment levels, but at another level it will be important for teachers to have details about future employment shifts in CS-related jobs that can be applied within their individual contexts. Monitoring and understanding shifts in local labour markets will also be an important element for schools to locate. There are initiatives in the UK that seek to support in this way. The Tech Partnership (2014) and CAS (2015) both provide links and support for schools, the former focusing more on present and future labour and employment, while the latter (with 10 centres of excellence across England) focuses more on CS projects and practices. It will be important that schools can create links with these groups, in order to both be aware of local opportunities, but also to gain from development opportunities.

Considering the **community argument**, teachers will require an understanding of how CS skills can be deployed within community-based situations. There are examples of initiatives where schools in the Netherlands, for example, enable engagement of their learners with external research issues that are identified by industry and community groups (reported in Passey 2013). This form of practice enables the learners to deploy problem-solving approaches, some involving levels of CS integration.

Within the compulsory education sector, other approaches have been piloted, exploring ways that different age groups might work with others in the wider community. In the UK, a recent example involved the setting up of a project within a primary school, where fathers were encouraged to come into school to work with their children on developing a Lego Mindstorms robot, and then to undertake a series of activities involving programming the robot. The project leaders found that fathers who came into the school had not previously had strong contact with the school, yet engaged fully with their children in these activities (during afternoon sessions, one session a week for 4 weeks, each lasting about 2 hours). The fathers reported their reasons for participating were concerned with their interests in 'making' (often arising from their earlier experiences); many of them had made models when they were young, perhaps using Meccano or Lego, and while programming was new to many of them, their interest in the technical making aspects was sufficient for them to feel that they could cope with taking on some additional skills that they were not familiar to them, especially as they could do this within a supportive environment and with their own children.

This form of project is not only concerned with development of CS skills, it is concerned also with a potential relationship between formal (classroom), non-formal (interest group) and informal (home) learning, and relationships of an intergenerational nature. While it has been suggested that there can be a 'digital disconnect' between creative uses of ICT and media that occur at home and the more traditional approaches that might be adopted at school (Furlong and Davies 2012), it is clear also that the ways that teachers can handle and manage activities within classrooms with a group of 30 learners is a factor that needs to be considered in this respect from a management perspective. The project above involved just 4 parents and 4 children, working in pairs. But with a class of 30 children, this is a model that might not be easy for a teacher to emulate (even having the space available to accommodate 30 pairs of children and fathers working together). However, if there are ways that the teacher can take advantage of projects that support classroom practices, then this is clearly of potential benefit. A current technology used by learners in schools in the UK, for programming and control, is the Raspberry Pi. These units are used by learners outside as well as inside schools, and some parents are taking more interest in how they might work with their children in using these within their own family home settings. A recent conference session for teachers and parents highlighted how a parent had worked with their child, programming a Raspberry Pi to capture video footage of a nesting bird and the hatching of its chicks.

The British Broadcasting Corporation (BBC) is now developing, with a consortium including ARM, Microsoft and Samsung, a new device called Micro Bits, which aims to put "control in the hands of the children" (CAS 2015). The intention is to equip all year 7 learners in the UK (about 750,000 learners, 11 years of age) with this device. As the Computing at School (CAS) article says, "Putting the kit in the hands of the children will help engage parents too", since the devices will be "owned by children". Teachers, parents and peers will all be able to benefit from access to and use of these devices, but clearly benefits that would bring together the opportunities when learners engage in formal, non-formal and informal activities are likely to be ultimately important, since separating them might not allow the same learning paths for their development. An important aspect of this Micro Bits initiative is concerned with an intention to support the addressing of what has become known as the 'digital divide', which has been used to describe and identify the different experiences that learners might have in different social contexts. In this respect, unequal access to new technology, where home access to a computer has been positively associated with higher levels of educational attainment (Chowdry, Crawford and Goodman 2010), is an aspect that this initiative seeks to accommodate at least, if not address.

Further research into this area of community involvement will require an exploration of approaches that can enable formal, non-formal and informal learning activities to be developed appropriately, and considered as an entirety (described in one context and approach by Kisiel 2014, as 'boundary activities').

Considering the **educational argument**, teachers will require an understanding of how CS can be integrated into curricula at school and subject levels. Many curricula are developed in ways that lead to formal classroom level practices; but as argued above, CS requires consideration of how curricula can be accommodated that can lead to non-formal and informal practices as well as formal ones. The new Australian curriculum states that learners should develop “knowledge, understanding and skills ...individually and collaboratively” (ACARA 2013). Individual learning can certainly be organised in formal ways, where learners have access to an individual desktop machine, perhaps. But for collaborative endeavour, it should be possible also for learners to be able to work in non-formal (groups like clubs or societies, to focus on specific interests) or informal situations (where they might use more mobile or flexible access). At the same time, schools will need to be aware of local possibilities if they are to be able to develop practices through links with non-formal and informal partners (businesses, agencies, community groups or parents, for example).

Further research is needed in this field, to identify ways that teachers can locally take approaches that match contexts and needs of their learners. Overall, the need for teacher development in terms of CS is clear; the fact that teaching practices have been developed through an ICT perspective in the past does not mean that teachers will be able to naturally or easily focus on CS needs and approaches as well. There is a need for some immediate, but also longer-term evaluation in this area, as the integration of CS in schools shifts over time.

5. COUNTRIES CONSIDERING CHANGES TO THE CURRICULUM

Recent national reports from members of the International Federation of Information Processing (IFIP) community suggest that the UK and Australia are indeed not alone in considering shifts with the balance of CS within the curriculum. Whether the six arguments above are a part of that concern in any specific country is not clear in the descriptions that follow, but, during May and June 2015, thirteen national reports on education and technology separately identified how these countries (spread across four continents) are currently considering CS within the compulsory education curriculum. While there is no single approach being identified in these reports across these thirteen countries (and indeed, as will be discussed later, this provides both an opportunity and challenge), it is clear that policy and curriculum concerns regarding the inclusion of CS is being raised internationally. Table 1 offers a view of how the thirteen countries are currently considering and approaching CS in terms of provision within the compulsory education sector. The sources of the individual national reports (from national Information Federation for Information Processing (IFIP) representatives to the technical committee on education), are shown in the table.

Country	**CS**	**Source**
Austria	Mandatory ICT/Informatics Education is only in grade 9 (age 14 years)	Futschek 2015
Australia	At all school levels the teaching of computer science or informatics is not mandated in any jurisdiction. The State of Victoria is committed to delivering a Digital Technologies Curriculum in 2016 to all levels of schooling. It is recommended that a Digital Technologies curriculum will be introduced as an elective subject in year 9. Each State has a final years’ of schooling subject offering in IT/ICT. In 2015, and moving towards 2016, the State of Victoria has implemented a suite of final years of schooling (Years 11 and 12) in computing, algorithmics and informatics	Reynolds 2015
Finland	In secondary schools, there are ICT driving license courses and quite a number of specialised courses like programming or numerical mathematics, and robotics has become very popular	Koivisto 2015
France	Computer science has been an elective subject for students in the scientific stream in general high schools. The May 2015 plan includes “coding” for elementary schools, “programming” for all students in middle schools (how and by which teachers is still to be defined, but within interdisciplinary activities), and there will be an elective course in computer science for all students (not only those in a scientific stream as now) and for the three levels	Grandbastien 2015

Country	CS	Source
Germany	Recommendations of the Federal Government are for improvement of the implementation of ICT-education in school curricula. CS equipment is provided by the States and the local communities. The curriculum is the responsibility of States. Informatics in primary education is mostly at the state of educational experiments. CSE is mostly not a mandatory school subject, but it can be chosen as an elective course. It can also be part of the Abitur examination. In some States, e.g. Bavaria, informatics is a mandatory course in the 5th grade of the gymnasium. Most States offer CSE courses as elective courses starting at grade 7 through to grade 13	Magenheim 2015
Ireland	In primary schools, activities are incorporated into the timetable informally by 'champion' teachers, including those programming with Scratch. In secondary schools, computing activities are mainly in the 4th Year (optional year – 15/16 year olds), including coding clubs and competitions, e.g. Coderdojo, Google Call to Code, and the ICS Skills 4-module Computing Curriculum	Leahy 2015
Italy	For computer science education, a policy is present for specialised education (e.g. upper secondary technical schools). A national optional initiative ("*coding the future*") has been launched with the aim of introducing primary school pupils to basic computer science concepts through coding	Bottino 2015
Japan	No special course for learning informatics or ICT is stated officially at the elementary stage. Lower secondary schools have informatics and programming curricula as a part of 'Technology and Home Economics', which is one of the compulsory subjects determined by the Ministry	Saito 2015
Lithuania	In secondary schools, a large number of ICT learning objectives are included in central steering documents, which include less common objectives such as programming skills and knowledge of computer hardware	Dagienė 2015
Portugal	Most classrooms have wi-fi, a computer, an interactive whiteboard or multimedia projector. A course is available in the 7th and 8th grades, which includes use of the programming language Scratch or Kodu. Informatics is available as a course in secondary education. A project to start next year is: The introduction to programming in the 1st Cycle (primary school)	Carvalho 2015
Republic of Korea	Basics and how to use algorithms, programming, problem solving, representing information, computer networks and computer ethics are taught as an elective course within the curriculum revised in 2009, but new national curricula with a compulsory course from 2017 will be introduced	Kimn 2015
South Africa	Computer science, informatics or ICT taught or used in secondary schools focuses on programming in Java and Delphi, but there is pressure to drop Java in favour of Delphi. There is a large focus on database development	van Niekerk 2015
Spain	The Madrid Regional Government introduced the CODE Madrid programme to teach programming and robotics at ESO (secondary education) level	Kloos 2015

Table 1: Current approaches to CS in the compulsory school sector in thirteen countries

As can be seen from details in Table 1, CS is, or will be, included in many curricula and curriculum practices in compulsory education sectors in these thirteen countries. In some countries, CS is now being introduced for younger learners, while in other countries ICT continues to be developed alongside the introduction of CS. It is also clear that in some countries, policy decisions are not universally accepted by educators, who see different arguments and concerns from those who are making substantive decisions. In other countries, working groups or consultation groups involving interested parties in computer societies and those in policy groups have been established, and these are leading in some cases to outcomes that range from curriculum documentation to project developments or even technology developments to support the compulsory education sector. Recent reports from other countries indicate such practices being involved (in the Netherlands, reported by Barendsen 2015, and in Sweden, for example).

In terms of patterns of implementation of CS, from the 13 countries represented in Table 1, and with England added as a fourteenth country, there are:

- 3 countries where CSE is a compulsory subject (including one to be introduced in 2016) at the primary education level.
- 2 countries where CSE is an optional subject at the primary education level.
- 5 countries where there is a compulsory CS course at the secondary education level (although this is not necessarily the case for all schools or all states).

- 7 countries where there is an optional CS elective course at the secondary education level.
- 5 countries where there are CS curriculum elements but no specific CS courses at the secondary education level.

6. EVIDENCE OF OUTCOMES OF CS IN COMPULSORY CURRICULA

Voogt et al. (2015), in their recent article, indicate the paucity of evidence of how computational thinking has led to any particular measurable learning outcomes, whether they are positive or negative. There is similarly a paucity of evidence about learning outcomes of CS in school curricula. Lye and Koh (2014) provide some evidence from their review of 9 studies, but also highlight points of caution in interpreting the results.

Hubwieser (2012) provides some longer-term evidence from his 9-year-long study of computer science in gymnasia in the State of Bavaria, Germany. In terms of a single year group graduating with their final 'Abitur' end-of-school examination in 2011, it is clear from the data presented that while there was 100% inclusion of learners in grades 6 and 7 where the CS subject was compulsory, the rate of inclusion from grade 9 onwards continued to fall (49.3% in grades 9 and 10, 14.2% in grade 11, 7% in grade 12, and 2.4% graduating with CS as part of their Abitur examination in 2011). However, it was noted that the level of outcome grades for CS in those Abitur examinations was higher than the average grade level for all subjects. Of particular note, however, was that data reported from teachers indicated girls' performance in CS declined across grades 6 to 12, while the performance of boys increased. According to Hubwieser (2012) this, and other decreases in performance, were attributed to the fact that "the introduction into object-oriented programming in this grade represents one of the most crucial points of our concept" (p.30).

In terms of the effects upon CS applications for university, Hubwieser (2012) found that the first cohort of learners graduating from gymnasia in Bavaria led to a vast increase in proportions of applications for CS courses at his University. It will be important to see if this indicator will be sustained, and whether numbers of applicants for CS courses will increase in the future.

7. CONCLUSIONS: FUTURE NEEDS AND FUTURE RESEARCH

Accepting that the six arguments outlined above constitute reasons for curriculum change, it is clearly important to explore to what extent the curriculum can, and indeed already has, considered these arguments and addressed them appropriately. While research needs to start to explore this new curriculum strategically, the starting point of existing evidence needs to be well understood and established. This point is echoed in the EDUsummIT 2015 research report, which identifies a number of key research issues and questions.

The learning argument implies that learners are gaining skills and competencies of value to them, for the future as well as in the present (problem-solving and creativity skills, as well as programming skills). Schools and teachers are likely to be initially immediately concerned with identifying these skills. Whether the constructionist forms of pedagogies are most appropriate to support the development of these skills, and how teachers and learners can identify problem-solving and creativity abilities, are both of clear concern. Research questions that would help to address these needs are:

- What forms of pedagogy can teachers deploy to best support the development of computational thinking, problem-solving and creativity through programming skills, for the 5 to 16 year age range?
- How should schools describe curricula to enable development of CS skills across their full age range?
- In what ways can schools, teachers and learners identify and monitor CS abilities and skills?

The learner argument implies that schools, teachers and learners are concerned with a curriculum that supports engagement and maintains interest. Research questions to help address these needs would be:

- What approaches enable learners of different ages to engage with CS or computing?
- In what ways can this interest and engagement be maintained?
- How can different groups of learners be supported most effectively in these respects?

The organisational argument implies that schools, teachers and learners understand how CS skills are currently used within organisations, and what this means in terms of the organisation of lessons and activities to enable skills to be developed in ways that match future and current employment needs. Research questions to help address these needs would be:

- How can a school gain understandings of how CS and computing skills are deployed and managed in organisations?
- What forms of team work or group work activities should teachers and schools offer?
- In what ways can contributions of learners be developed, monitored and assessed when they are involved in CS group work activity?

The economic argument implies that schools, teachers and learners will not just recognise that future employment will require more CS skills, but enable and understanding of where those skills might be needed, and how they are used and applied. Research questions to help address these needs would be:

- How can a school gain understandings at a useful local as well as other levels about the ways CS are being used and developed in employment situations, and what future needs are predicted?
- In what ways can this knowledge be best deployed to support future skills and employment possibilities for learners?

The community argument implies that schools, teachers and learners understand the social or community contexts in which CS and ICT will be used and deployed. School provision is often formal by nature; the community argument is based on supporting activities in non-formal or informal, as well as formal, ways, however. Research questions to help address these needs would be:

- What support activities can be undertaken in non-formal or informal situations, linking with community groups or organisations, to address their problem-solving and creative needs?
- How can outcomes of community-based activities be monitored and assessed to support learner needs?

The educational argument implies that schools, teachers and learners have access to facilities that will enable educational outcomes to be realised. These facilities clearly concern not just computing facilities, but also the facilities that teachers can bring to the classroom, and the activities that learners will engage in. Research questions to help address these needs would be:

- In what ways should a school most effectively deploy facilities and resources to support a curriculum providing CSE activities for all learners across all age ranges?
- When and how should a school review its facility and resource provision to ensure future needs for learners are accommodated?

A balance of formal, non-formal and informal activities appears to be crucial to the future success of CS in the compulsory education sector. There is a need to develop fundamental skills and to practice these through formal learning activities, to develop working in pairs and teams through collaboration and project approaches through non-formal learning activities, and to develop practices that solve and address real-life problems in family or community settings through informal learning activities. Research has a major role to play in helping to achieve these goals; it is vital that we formulate an appropriate long- as well as short-term research and evaluation strategy to explore the arguments and potential of this new curriculum area fully.

ACKNOWLEDGEMENT

The author would like to thank Jari Koivisto for the opportunity to develop and present the initial ideas that formed the basis of this paper. Jari's interest in, feedback on and translation of an original paper from English to Finnish are all very gratefully acknowledged. Thanks also go to the IFIP TC3 national representatives who submitted national education and technology reports, providing an important international perspective for an earlier version of this paper, and to the IFIP TC3 2015 Conference organisers in Vilnus, Lithuania, for their encouragement in writing that earlier version, and for its subsequent publication. Lastly, my thanks go to Claire Johnson, whose PhD thesis provided invaluable, further insight into this topic, and to the advisers and

doctoral students at the second NORDNice meeting and the sixth doctoral consortium meeting in Druskinikai, Luthuania.

REFERENCES

ACARA. (2013). *Draft Australian Curriculum: Technologies*. http://consultation.australiancurriculum.edu.au/Static/docs/Technologies/Draft%20Australian%20Curriculum%20Technologies%20-%20February%202013.pdf. Accessed 5 January 2015.

Bottino, R. (2015). *National Report on Education and Technology: Italy*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.

Bureau of Labor Statistics. (2012). *Economic News Release, Employment Projections: 2010-2020 Summary*. Washington, DC: United States Department of Labor.

Bureau of Labor Statistics. (2015). *Occupational Outlook Handbook, 2016-17, Web Developers*. Washington, DC: United States Department of Labor. http://www.bls.gov/ooh/computer-and-information-technology/web-developers.htm. Accessed 28 December 2015.

Carvalho, A. (2015). *National Report on Education and Technology: South Africa*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.

Chowdry, H., Crawford, C., & Goodman, A. (2010). *The Role of Attitudes and Behaviours in Explaining Socio-Economic Differences in Attainment at Age 16: IFS Working Paper 10/15*. London: Institute of Fiscal Studies.

Computing at School. (2015). *Switchedon: Computing at School Newsletter Summer 2015*. London: CAS and BCS.

Dagiené, V. (2015). *National Report on Education and Technology: Lithuania*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.

DFE. (2013). *Statutory guidance – National curriculum in England: computing programmes of study*. https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study. Accessed 5 January 2015.

EDUsmmIT 2015 (2015). *TWG 9: Curriculum - Advancing understanding of the roles of CS/Informatics in the Curriculum: Research Plan*. http://www.curtin.edu.au/edusummit/local/docs/twg9-research-plan.pdf. Accessed 27 December 2015.

e-Skills UK. (2012). *Technology Insights 2012*. https://www.e-skills.com/Documents/Research/Insights-2012/TechnologyInsights_2012_UK.pdf. Accessed 9 October 2015.

e-Skills UK and the BCS. (2014). *The Women in IT Scorecard*. London and Swindon: e-Skills UK and BCS. http://policy.bcs.org/content/women-it-scorecard-it-profession-numbers. Accessed 6 June 2015.

Ferrari, A. (2013). *DIGICOMP: A Framework for Developing and Understanding Digital Competence in Europe*. Seville, Spain: Institute for Prospective Technological Studies. http://ftp.jrc.es/EURdoc/JRC83167.pdf. Accessed 5 January 2015.

Fuller A., Turbin, J., & Johnston, B. (2013). Computer Club for Girls: An initiative in search of the right problem? *Gender and Education*, *25*(4), 499–514.

Furlong, A., & Davies, C. (2011). Young people, new technologies and learning at home: taking context seriously. *Oxford Review of Education, 38*(1), 45-62.

Futschek, G. (2015). *National Report on Education and Technology: Austria*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.

Glover, J., & Guerrier, Y. (2010). Women in hybrid roles in IT employment: A return to 'nimble' fingers?' *Journal of Technology Management & Innovation*, *5*(1), 85-94.

Grandbastien, M. (2015). *National Report on Education and Technology: France*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.

Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher, 42*(1), 38–43.

Hanover Research. (2011). *A Crosswalk of 21st Century Skills*. Washington, DC: Hanover Research. http://www.hanoverresearch.com/wp-content/uploads/2011/12/A-Crosswalk-of-21st-Century-Skills-Membership.pdf. Accessed 27 December 2015.

Higgins, S., Xiao, ZM., & Katsipataki, M. (2012). *The Impact of Digital Technology on Learning: A Summary for the Education Endowment Foundation - Full Report*. Durham and London Durham University and Education Endowment Foundation.

Hubwieser, P. (2012). Computer science education in secondary schools – The introduction of a new compulsory subject. *ACM Transactions in Computer Education, 12*(4), Article 16, 41 pages.

Johnson, C. (2014). 'I liked it but it made you think too much': A case study of computer game authoring in the Key Stage 3 ICT curriculum. PhD thesis, Norwich: University of East Anglia.

Joint Council of Qualifications. (2014). *Examination Results*. http://www.jcq.org.uk/examinationresults/. Accessed 29 June 2015.

Kafai, Y., & Burke, Q. (2013). Computer programming goes back to school. *Phi Delta Kappan, 95*(1), 61–65.
Kay, D. S. (1991). Computer interaction: Debugging the problems. In R. J. Sternberg & P. A. Frensch (Eds.). *Complex problem solving: Principles and mechanisms* (pp. 317-340). Hillsdale, NJ: Lawrence Erlbaum Associates.
Kimn, H.J. (2015). *National Report on Education and Technology: Republic of Korea*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.
Kirkup, G. (2011). Preparing women for dead-end jobs? Vocational education and training (VET) for Information and Communication Technology (ICT) jobs'. *International Journal of Gender, Science and Technology, 3*(2).
Kisiel, J.F. (2014). Clarifying the complexities of school–museum interactions: Perspectives from two communities. *Journal of Research in Science Teaching, 51*(3), 342-367.
Kloos, C. (2015). *National Report on Education and Technology: Spain*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.
Koivisto, J. (2015). *National Report on Education and Technology: Finland*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.
Leahy, D. (2015). *National Report on Education and Technology: Ireland*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.
Livingstone, I., & Hope, A. (2011). *Next Gen. Transforming the UK into the world's leading talent hub for the video games and visual effects industries: A Review*. London: Nesta.
Lye, S.Y., & Koh, J.H.L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior, 41*, 51–61.
Magenheim, J. (2015). *National Report on Education and Technology: Germany*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.
McCormack, J., & d'Inverno, M. (eds.). (2012). *Computers and Creativity.* Heidelberg, Germany: Springer Verlag.
McDowell, J., 2015. A black swan in a sea of white noise: Using technology-enhanced learning to afford educational inclusivity for learners with Asperger's Syndrome. *Social Inclusion*, *3*(6), 7-15.
McEwan, T., & McConnell, A. (2013). Young people's perceptions of computing careers. In *Proceedings of the 43rd Annual Frontiers in Education Conference,* pp. 1597-1603. Piscataway, NJ: IEEE.
Papert, S. (1980). *Mindstorms - children, computers, and powerful ideas*. New York, NY: Basic Books.
Papert, S., & Harel, I. (Eds.). (1991). *Constructionism: research reports and essays 1985 – 1990.* Norwood, NJ: Ablex Publishing Corporation.
Passey, D. (1999). Strategic evaluation of the impacts on learning of educational technologies: Exploring some of the issues for evaluators and future evaluation audiences. *Education and Information Technologies, 4*(3), 1-28.
Passey, D. (2012). *Independent evaluation of the Little Big Planet 2 project in Wolverhampton's Local Education Partnership schools: Outcomes and impacts – Final report*. Lancaster: Lancaster University.
Passey, D. (2013). *Inclusive technology enhanced learning: Overcoming Cognitive, Physical, Emotional and Geographic Challenges*. New York, NY: Routledge.
Passey, D. (2014). Early uses of computers in schools in the United Kingdom: shaping factors and influencing direction. In A. Tatnall & B. Davey (Eds.). *Reflections on the History of Computers in Education: Using Computers and Teaching about Computing in Schools from the late 1970s to the early 1990s.* Heidelberg, Germany: Springer.
Passey, D. (2015). Computer science (CS) or information and communication technologies (ICT): the curriculum needs both. In A. Brodnik & C. Lewin (Eds.). *IFIP TC3 Working Conference "A New Culture of Learning: Computing and next Generations": Proceedings.* pp. 26-44. http://www.ifip2015.mii.vu.lt/file/repository/IFIP_Proceedings.pdf. Accessed 20 July 2015.
Reynolds, N. (2015). *National Report on Education and Technology: Australia*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.
Royal Society. (2012). *Shutdown or restart? The way forward for computing in UK schools*. https://royalsociety.org/education/policy/computing-in-schools/report/. Accessed 29 June 2015.
Saito, T. (2015). *National Report on Education and Technology: Japan*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.
Tatnall, A., & Davey, B. (2014). *Reflections on the History of Computers in Education: Using Computers and Teaching about Computing in Schools from the late 1970s to the early 1990s.* Heidelberg, Germany: Springer.
The Princes Trust. (2014). *Digital Literacy Survey 2013*. http://www.princestrust.org.uk/about_the_trust/what_we_do/research/digital_literacy_research_2013.aspx. Accessed 29 June 2015.
The Tech Partnership. (2015). *Tech Partnership Progress update April 2014 – March 2015*. London: The Tech Partnership.
U3A Online. (n.d.). *U3A Online*. http://www.u3aonline.org.au/. Accessed 5 January 2015.
UCISA. (2010). *Survey of Technology Enhanced Learning for higher education in the UK*. http://www.ucisa.ac.uk/groups/ssg/~/media/groups/ssg/surveys/TEL%20survey%202010_FINAL.ashx. Accessed 29 June 2015.

UCISA. (2012). *Survey of Technology Enhanced Learning for higher education in the UK*. https://www.ucisa.ac.uk/~/media/groups/ssg/surveys/TEL_survey_2012_with%20Apps_final.ashx. Accessed 29 June 2015.

UCISA. (2014). *Survey of Technology Enhanced Learning for higher education in the UK*. https://www.ucisa.ac.uk/~/media/groups/dsdg/TEL%20Survey%202014_29Sep2014.ashx. Accessed 29 June 2015.

UK Digital Skills Taskforce. (2014). *Digital Skills for Tomorrow's World*. http://www.ukdigitalskills.com/wp-content/uploads/2014/07/Binder7-REDUCED2.pdf. Accessed 9 October 2015.

Van Niekerk, J. (2015). *National Report on Education and Technology: South Africa*. http://ifip-education.ning.com/page/national-reports-on-education-and-technology. Accessed 1 July 2015.

Voogt, J., Fisser, P., Good, J., Mishra, P., & Yadav, A. (2015). Computational thinking in compulsory education: Towards an agenda for research and practice. *Education and Information Technologies, 20,* 715–728.