Computing at School:
the state of the nation

A report of the Computing at School Working Group

For the UK Computing Research Committee

November 2009

Contact: Simon Peyton Jones (simonpj@microsoft.com)

1 Executive summary

Digital technology is everywhere. Our young people should be educated not only in the *application* and *use* of digital technology, but also in *how it works,* and *its foundational principles*. Lacking such knowledge renders them powerless in the face of complex and opaque technology, disenfranchises them from making informed decisions about the digital society, and deprives the UK of a well-qualified stream of students enthusiastic and able to envision and design new digital systems.

- Our current educational provision at school focuses strongly on the use of computers ("**ICT**"), but fails utterly to study how they work, or their underlying principles (which we will call "**Computing**"). ICT is like learning how to *read*; certainly a skill that everyone should have. Studying Computing is like learning how to *write* by engaging in the creative process of understanding, designing, and building new systems. Everyone should learn to write, even though only a minority will become professional authors (Section 2).

  No one disputes the importance of ICT; but an exclusive emphasis on ICT means that today's school pupils have *fewer* opportunities to learn Computing than they did 20 years ago.

- There is ample evidence to show that students are disenchanted with ICT (Section 3). Students arrive at school with a much richer background in ICT than they did a decade ago; ICT is (rightly) used pervasively across the curriculum. The place of ICT in the curriculum has moved from maths departments to design and technology and is now often found linked to business studies and office practice. Efforts to enrich the ICT curriculum beyond basic word processing and spreadsheet skills have had only limited success. As a result, students are taught skills they already know, sometimes several times over, and ICT courses are burdened with undemanding coursework that often onerous to teacher in terms of its bureaucratic demands. There is no GCSE-level qualification in Computing.

- This disenchantment has led to a collapse in student numbers applying for university courses in Computer Science (Section 4). This is a serious concern going far beyond the narrow interests of affected departments. Employer demand for Computer Science graduates is strong and growing, just as the supply is contracting. It appears that the negative perceptions of ICT at school have tainted our young people's entire attitude to Computing.

- Yet Computing is a rich discipline, on a par with Science, Engineering, and Mathematics (Section 5). Like these subjects, it is built on enduring principles, methods, and habits of mind. Through studying the discipline, students learn generic skills, such as problem-solving and abstraction. Computing leads to deep enquiry, such as what it means to think, and new insights into life itself.

This report makes a number of recommendations, which appear in Section 7. Briefly, we advocate the recognition of Computing as a STEM subject; the provision of Computing material at Key Stage 3 within the existing National Curriculum; the development of GCSE criteria for Computing; and CPD courses in Computing for hard-pressed teachers.

2 What is "Computing"?

Computing is the study of how computers and computer systems work, and how they are constructed and programmed, and the foundations of information and computation. Not only is computing extraordinarily useful, but it also intensely creative, and suffused with both visceral ("it works!") and intellectual ("that is *so* beautiful") excitement.

2.1 Computing is a discipline

Computing is a **discipline**, like mathematics or physics, that explores foundational **principles** and **ideas** (such as algorithms for finding shortest paths in a graph, or approaches to flow control in the Internet), rather than **technologies** and **skills** (such as the ability to use Excel), although it may use the latter to illuminate the former.

These technologies and skills are certainly useful, and go by the name of Information and Communication Technologies (ICT) in schools. But ICT is not Computing, just as numeracy is not mathematics. ICT is about the **technology** and **application** of computers. Computing is about their **principles** and **design**. To use the analogy of a car,

- **ICT** covers the skills of using popular applications (almost invariably spreadsheets, databases, and word processing), and then develops in the direction of business organisations and processes, health and safety, data protection and copyright, and so on. This is the equivalent of teaching how to **drive** a car safely, and how to use a car appropriately once you can drive it.
- **ICT** also includes some material on systems integration and management, computer networks, and creating simple web sites. This is like training car **mechanics**, who can maintain and repair cars, and add new optional components when necessary.
- The discipline of **Computing** is analogous to car **design**, including the principles of thermodynamics, friction, and fluid dynamics; how major components work (clutches, ignition systems, etc); the design space of cars (gasoline, diesel, electric etc); and how to design new cars and components.

There is a big difference between a discipline and a skill. We are preparing young people for jobs that don't yet exist, requiring technologies that have not yet been invented, to solve problems of which we are not yet aware. Skills and technologies may seem topical, but quickly go out of date; principles and ideas may appear less up-to-the-minute (yes, even in computing!) but are transferrable and remain important and applicable a decade or two later.

For example, the 2009 report "ICT for the UK's future", published by the Royal Academy of Engineering states:

- *"There is an underlying confusion between IT as a fundamental life-skill and 'enabler' in the teaching of all subjects, and computing as a scientific discipline, with the present balance skewed towards teaching 'software use'. Students should be encouraged to explore what goes on behind the IT applications they use, from social networking and messaging tools, to computer graphics and computer games."*
- *"It is essential that a significant proportion of the 14-19 age group understands computing concepts – programming, design, problem solving, usability, communications and hardware. It is of particular importance to reform the teaching curriculum in schools to differentiate between the learning of genuine IT and the use of IT. Understanding the basis of the subject is fundamental."*

It is common ground that young people should learn some of the principles of Physics, even though very few will become physicists, because these principles illuminate our understanding of much of what we see around us, and empower us to make informed choices. Similarly, as well as gaining ICT skills, all students should learn some elements of Computing, and those that want to study the discipline in more depth should be provided with more opportunities to do so.

2.2 The content of Computing

In concrete terms, Computing includes (among many other things)

- The study of **algorithms** and **data structures**: efficient and ingenious ways to solve computational problems, together with a rich underlying theory of the "complexity" of such algorithms.
- An understanding of computer **systems** and **networks**: for example, how the internet works, and the protocols that keep data flowing smoothly despite all the control being decentralised.
- An appreciation of the challenges of **human-computer interaction**, which focuses on the challenge of making computers accessible to people.
- **How computers work**. Traditionally this means gates, binary arithmetic, and digital hardware. More broadly, biologically-inspired computation paradigms are in rapid development.

In all of these areas there is a fruitful interaction of theory, design, and experimentation. For example, information theory informs the design of compression algorithms, whose performance on real test loads is measured experimentally. Indeed, Computing is a fascinating blend of engineering, mathematics, science, and technology. Like engineering it involves design, construction and test. Like mathematics it involves theory, logic, and reasoning. Like science it embraces measurement and experiment. And of course, without technology no computer program could run and the subject would be dry indeed. In short, Computing is the quintessential STEM subject.

Although computing is not just **programming**, a working knowledge of programming is necessary for a thorough grounding in computing. In an educational context, programming plays a special role: with its focus on problem solving, creativity, sequencing and logic, programming helps foster the personal, learning and problem-solving skills required in the modern school curriculum. Furthermore, it is an extremely powerful motivator: nothing motivates students like making computers dance to their tune. For this purpose "programming" clearly includes scripting and other forms of "glue" that allow us to build large systems from software components.

Computing is not even primarily even about **computers**. Computers are pieces of machinery which permit the development of computing as pencil and paper permit the development of writing. We teach students the nitty-gritty of writing not primarily as a skill, but rather to unleash their creativity and self-expression. Similarly, computers offer students new fields of creative expression, and new modes of thought.

We use the terms "computing" and "computer science" interchangeably.

3 Current ICT provision is not meeting the needs of today's students

Computing is one of the most exciting subjects on earth. Yet the current arrangements for teaching computing at school leave many of our students feeling that it is irrelevant and dull. There are a number of contributory factors, many of which are inter-linked.

The pre-sixth-form school curriculum is a disaster as far as computing is concerned. The emphasis for the last few years has been on ICT literacy, often in a rather stereotypically narrow set of applications. Whilst the original intentions were good, the attractiveness of learning ICT skills has declined as computers have become ubiquitous. There is no joined up overall curriculum, which results in the same topics being taught repeatedly. There is a disconnect between the subject in schools and the subject as taught at HEIs.

- **Key Stage 3 (11-14):** The statutory Key Stage 3 curriculum in ICT has failed to develop imaginatively after the decision to drop external testing at the end of Key Stage 3, with the cycle of updates lagging far behind where the curriculum ought to be. The demise of the National Strategy has further widened this gap. Although the National Curriculum says nothing about word processors, spreadsheets, presentation software and databases, the reality is that these applications dominate the curriculum, and material that the students already know is often repeated. Although the recent revision to the ICT National Curriculum offers greater scope to develop introductory computing exercises, particularly in the areas of sequencing and modelling, there is little evidence that this flexibility is being adopted in practice.
- **Key Stage 4 (GCSE, 14-16):** a similar situation applies to the statutory Key Stage 4 programme of study in ICT, and the optional GCSE in ICT. The syllabi are simply boring and de-motivating. There is a GCSE in ICT, but no GCSE in Computing[1]. Many students are not even getting a full entitlement in ICT as required by statute. For example, *"the requirement that students must 'apply, as appropriate, the concepts and techniques of using ICT to measure, record, respond to, control and automate events' may be missed"* (The Importance of ICT OFSTED 2009 para 95). These are the elements of the curriculum that are closest and most relevant to computing.
- **14-19 Diploma**: From September 2008, the Secondary Curriculum Reform has introduced a new concept of curriculum planning, in which students are taught the core within employment sectors (lines of study). The IT and Telecommunication sector is represented by e-skills UK, which canvass IT employers' view and produced the subject criteria for the IT Diploma. Early experiences have indicated positive responses from students due to the problem solving emphasis and the opportunities for extended projects. The principal learning of the Diploma, however, revolves round Business, Technology and People. The choices available for the majority of the specialist learning options are restricted by the availability of technical courses available.
- **Key Stage 5 (A level, 16-18):** There are AS/A Levels in ICT but these contain very little computing. Examining Boards do offer syllabi in Computing but these have to follow a prescribed and constraining subject core shared with ICT AS/A Level, which has distorted the current AS/A Level Computing syllabi (2000-present). In 2005, however, AS/A Level Computing was granted its own subject core for the

[1] There is an IGCSE (International GCSE) in Computing Studies, but it contains little Computing. Furthermore it is only available to independent/private schools; state schools cannot teach it because it does not count in the school league-table scoring.

development of revised AS/A Level Computing syllabi for teaching from September 2008[2].

- In general, A-level Computing is not considered to be sufficiently aligned with **university courses in computing** to be given valued status by the universities. Why not? Because the current A levels contain little of the foundational material on which a first-year university course might build. (This is being partly addressed by the AQA syllabus rewrite.) Universities do acknowledge that in a subject with a high drop out rate at undergraduate level, students with Computing A-level tend to stay the course. However, recognition by universities of A Level Computing remains a challenge. They prefer Maths.

3.1 A pattern of decline

There is abundant evidence that, while the current ICT provision in schools serves some needs well, it is actually counter-productive for able students interested in computer technology. For example:

- There has been a precipitous fall in the number of students taking ICT or Computing courses over the last few years:
    - The number of students studying **A level Computing** fell 7% from 5,068 in 2008 to 4,710. This continues a sustained fall in every year since 2001, apart from a slight rally in 2003. Over the period 2001-2009 the total fell by 57% (10,913 down to 5,610)[3]. Not only is the number declining, but it is very small in absolute terms: just 0.6% of students take A level Computing.
    - The number of students taking **A level ICT** reached a high of 18,029 in 2003, but has fallen every year since then, to 11,948 in 2009, a fall of 33% in six years.
    - The number of students studying **GCSE in ICT** reached a high of 109,601 in 2006, but has declined particularly steeply to 73,519, a fall of 33% in only three years.

  These numbers have been widely reported in the press[4].

- A survey of 1000 students in July 2009 by Edge[5], an independent educational foundation, found that a majority (56%) were "unmotivated by three or more of their subjects". This result is not ICT-specific, but the same survey asked what other subjects the students would like to study instead. *The most popular choice was computer programming* (22%), beating criminology (21%) and film (18%).
- The 2008 CRAC report "Do undergraduates want a career in IT?[6]" surveyed over 1000 undergraduate computing students and found that "*Although the majority were happy with their choice, only 11% of computing students felt that the discipline had been strongly promoted to them as a degree choice while at school and over 40% felt that it had received very little promotion there.... The computing students cited a number of reasons for their choice of degree course. The overwhelming majority of*

[2] 14-19 Education and Skills white paper, page 65, section 8.26
[3] Source: Joint Council for Qualifications, http://www.jcq.org.uk
[4] E.g. "ICT and Computing A levels continue to slide" ITPro, http://www.itpro.co.uk/614179/ict-and-computing-a-levels-continue-to-slide
[5] http://www.edge.co.uk/news/gcse-students-disinterested-in-exams
[6] http://www.crac.org.uk/crac_new/pdfs/undergraduates_it.pdf

male students appeared to be driven by their personal interest or aptitude for computing (and a lower proportion, but still two thirds, of females)."

- The 2008 "IT & Telecoms Insight Report[7]" published by Eskills UK says *"The image of IT-related degrees and careers was that they would be repetitive, boring, and more-of-the-same; for example use of IT office applications such as word processing, spreadsheets, and databases"*. The next bullet says *"The ICT GCSE had a major part to play in creating their (negative) impressions"*.
- The 2007 report "Developing the future"[8], sponsored by Microsoft, City University, the British Computer Society, and Intellect says *"With no GCSE in Computing or Computer Science (only the GCSE in ICT which is not about the subject of computing) learning to use a computer and learning Computer Science become indistinguishable as far as students are concerned. The skew in emphasis has a direct bearing on a student's view of the IT industry; one that results in many negative perceptions".*

This pattern of decline is replicated in other countries. For example, in the USA, the 2009 CSTA National Secondary Computer Science Survey surveyed over 1000 teachers, finding that "*only 65% of schools offer an introductory CS course, compared with 73% in 2007 and 78% in 2005. Likewise, the number of advanced placement CS courses has also declined, with 27% in 2009, 32% in 2007, and 40% in 2005*". The USA is taking this issue very seriously. In October 2009 the House of Representatives established (by 405 votes to 0) a National Computer Science Education Week. The text of the resolution[9] clearly distinguishes the discipline of Computer Science from IT (e.g. "*the field of computer science underpins the information technology sector of our economy*").

3.2 Rethinking ICT

The March 2009 Ofsted report "The importance of ICT"[10] has a particularly detailed assessment of ICT. The chapter "Rethinking ICT qualifications and progression routes" amounts to a strong critique of the educational value of teaching ICT skills with a focus on their vocational usefulness

- "*The assessment requirements of some vocational qualifications may actually be limiting students' achievement. In many of the schools visited, higher-attaining students were insufficiently challenged....much of the work in ICT at Key Stage 4, particularly for the higher attainers, often involved consolidating skills that students had already gained proficiency."*
- *"Too many of the lessons seen during the survey emphasised the development of skills in using specific software at the expense of improving students' ICT capability."*
- *"Students had plentiful opportunities to use ICT for presenting their work well and communicating their ideas....[but] coverage of control, sensors and databases was limited in many of the schools, as was the provision for students to learn the logical thinking necessary to program, write scripts or macros, which was cursory and superficial."*
- [Concerning vocational courses at Key Stage 4] *"Most of the competencies related to spreadsheets and databases that students are required to demonstrate for*

---

7 http://www.e-skills.com/Research-and-policy/Insights-2008/2181
8 http://www.microsoft.com/uk/developingthefuture/default.mspx
9 http://thomas.loc.gov/home/gpoxmlc111/hr558_eh.xml
10 http://www.ofsted.gov.uk/content/download/9167/101177/file/The%20importance%20of%20ICT.pdf

accreditation have already been covered at Key Stage 3. Many students therefore repeat work; the expectations for progressing further in using control, data logging, spreadsheets and databases are low.

"The predominance of vocational GCSEs also has ramifications for post-16 study. The vocational courses are poor preparation for the demands of A-level computer studies and ICT courses. Consequently, the number of students choosing these sixth-form courses is low. Compared to 2004, in 2007 around 25% fewer students were entered for A-level ICT, with the decrease comprising boys and girls equally. Over the same period, the decline in A-level computer studies was more severe with a 32% drop in entries (45% drop in girls' entries and 31% for boys)."

Treating Computing as a discipline, on a par with mathematics and science, rather than merely as a useful skill (and one which many students now acquire early, or even at home), directly addresses these concerns.

4 Universities and employers

Computing has an immense impact on modern life. The job prospects are excellent and the field is rigorous, intellectually vibrant, and multi-faceted. Yet, computing is in danger of disappearing from schools, with a critical skills shortage developing. This skills gap leaves the UK vulnerable, and unable to supply or control a key technology.

The number of students applying to computing courses at university level has halved in the last 10 years, despite increasing take-up of university education, and strong employer demand. Ironically, many at universities directly attribute this fall in numbers to the increased spread of ICT at school. The dot-com bust, ill-founded myths about outsourcing, the perceived public image of computing, general ignorance about the shortage of well-qualified specialists in computing, and the everyday familiarity of digital technology, may also have had an effect. It is worth noting here that the number of girls applying for such courses has reduced even more dramatically over the past 15 years.

A recent study[11] by the UK Council of Professors and Heads of Computing illustrates the problem: it predicts that demand for IT professionals will increase by up to 15% in the next eight years, while the number of students aiming for jobs in the industry has fallen by 50% since 2001. They further identified:

- The IT labour market is set to grow by 163,000 from 2007 to 2016 (from 1,069,000 up to 1,232,000).
- 179,800 appointments are made each year in the IT labour market, the majority (78.5%; 141,300) of which will go to "new entrants" (people who are not currently in the IT labour market). Of this annual requirement of 141,300, 26,800 will be joining direct from education
- In 2005, an IT "Skill-Shortage Vacancy" was experienced by 5% of all employers. This equates to 28% of all employers with a vacancy
    - 38% of IT Managers have a technical skill gap, as do 12% of Networking Staff, 10% of Programmers, and 10% of PC Support Staff.
    - It is crucial to understand that technical staff and managers are the two areas where the largest employment growth will take place over the coming years. A technical skill gap amongst such people is a serious problem.
- UK university applications to read Computer Science are down over 60% since 2000.
- At a recent job fair for computing students in Cambridge there were more employers than students. Many of these employers had vacancies for the entire cohort.

[11] http://www.cphc.ac.uk/docs/reports/cphc-itlabourmarket.pdf

5 Computing as a discipline

The invention of the computer in the 20th century is a "once in a millennium" event, comparable in importance to the development of writing or the printing press. Computers are fundamentally different from other technological inventions in the past in that they directly augment human thought, rather than, say, the functions of our muscles or our senses. A fundamental understanding of computing enables students to be not just educated users of technology, but the innovators capable of designing new computers and programs to improve the quality of life for everyone.

5.1 Computing is Important Intellectually

Computers have already had enormous impact on the way we live, think, and act. It is hard to overestimate their importance in the future. We live in a digitized, computerized, programmable world, and to make sense of it and influence it, we need computing. An engineer using a computer to design a bridge must understand the limitations of the numerical methods used, how the maximum capacity estimates were computed and how reliable they are. An educated citizen using a government database or bidding in an eBay auction should have a basic understanding of the underlying algorithms of such conveniences, as well as the security and privacy issues that arise when information is transmitted and stored digitally.

These are computing, not ICT issues. Computing students learn logical reasoning, algorithmic thinking, design and structured problem solving—all concepts and skills that are valuable well beyond the computing classroom. Students gain awareness of the resources required to implement and deploy a solution and how to deal with real-world and business constraints. These skills are applicable in many contexts, from science and engineering to the humanities and business, and have already led to deeper understanding in many areas. Computer simulations are essential to the discovery and understanding of the fundamental rules that govern a wide variety of systems from how ants gather food to how stock markets behave.

Computing is also one of the leading disciplines helping us understand how the human mind works, one of the great intellectual questions of all time. What does it mean to be "conscious"? Can a computer "think"? If a computer plays chess better than a human, does that make it smarter?

5.2 Computing Leads to Multiple Career Paths

The vast majority of careers in the 21st century will require an understanding of computing. Many jobs that today's students will have in 10 to 20 years haven't been invented yet. Professionals in every discipline—from art and entertainment, to communications and health care, to factory workers, small business owners, and retail store staff— need to understand computing to be globally competitive in their fields. Movies like *The Incredibles* and *Lord of the Rings* required the development of new computing techniques. Progress on understanding the genetics of disease or of creating an AIDS vaccine requires professionals to think in terms of computing—because the problems are unsolvable without it. Those who understand the technology can make the new movies and invent the new techniques, and they are the professionals who will go beyond simply using what others have invented. Studying computing will prepare a student to become a professional software developer or to pursue a career in one of many related fields. Despite the depressing reports in the media, the reality is that professionals with computing training have never been more in demand in the UK and worldwide than they are today. Network managers need computing expertise to install new kinds of routers. Professional computer scientists rarely spend their days writing program code. More often they are working with experts in many fields, designing and building computer systems for every aspect of our society.

5.3 Computing Teaches Problem Solving

Artists, philosophers, designers, and scientists in all disciplines are united in the intensely creative activity of problem solving. Every painting by Picasso is an attempt to solve the problem of capturing an active, three-dimensional world on a flat canvas. Every TV commercial is an attempt to solve the problem of how to entice people to want, and then purchase, a product. And every well-designed scientific experiment provides data to support or refute a theory.

Computing teaches students to think about the problem-solving process itself. In computing, the first step in solving a problem is always to state it clearly and unambiguously. Often a computer scientist works closely with business people, scientists, and other experts to understand the issues, and to define the problem so explicitly that it can be represented in a computer.

This co-operative process requires people with different expertise and perspectives to work together to clarify the problems while considering each other's priorities and constraints. Computer programs must be designed, written, and tested. New hardware or devices may need to be made. Existing software systems and packages may be modified and integrated into the final system. Building a system is a creative process. The process requires computational thinking. With each fix of a bug or addition of a new feature, there's a hypothesis that the problem has been solved. Data is collected, results are analyzed, and if the hypothesis is untrue (alas, often!), the cycle repeats.

A computer scientist is concerned with the robustness, the user-friendliness, the maintainability, and even the formal correctness of computer solutions to business, scientific, and engineering problems. These issues often require intense analysis and creativity. Computer specialists draw on their training and experience to avoid problems and to create the best possible solutions. Often this involves creating new programs and systems. That takes computing skill.

Computer programs and systems are amongst the most complex entities constructed by man. They require precise teamwork and management of uncertainty for which techniques such as agile engineering have been developed. These are transferable skills.

5.4 Computing Can Engage All Students

Computing applies to virtually every aspect of life, so computing can be explicitly tied to a myriad of student interests. Students may be fascinated with specific technologies such as cell phones or have an innate passion for visual design, digital entertainment, or helping society. Computing teaching nurtures students' interests, passions, and sense of engagement with the world around them and offer opportunities for them to find purpose and meaning in their lives.

Pedagogically, computer programming has the same relation to studying computing as playing an instrument does to studying music or painting does to studying art. In each case, even a small amount of hands-on experience adds immensely to life-long appreciation and understanding, even if the student does not continue programming, playing, or painting as an adult. Although becoming an expert programmer, a violinist, or an oil painter demands much time and talent, we still want to expose every student to the joys of being creative, for example by having students design and write programs that control their cell phones or robots, create physics and biology simulations, or compose music. Students will want to learn to use conditionals, loops, and parameters and other fundamental concepts just to make these exciting things happen. Examples include:

- **Computing and Digital Media**. Manipulating and creating digital media is a context that engages students and easily integrates with computing learning goals. Instead of iterating over an array to compute an average, students might write a program to iterate over an array of pixels to compute a negative image or a grey-scale image or try new forms of image manipulation. Students can learn that combining two arrays is the technique used to splice and mix digital sounds. Processing pictures and sounds in new ways needs new programs. Similar contexts are robotics and story-telling with digital media.
- **Team work to solve large uncertain problems**. Computer programs are some of the most complex structures built by mankind, and are rarely built in isolation. Software engineering has worked out paradigms for designing and building such structures by a co-operating team, often in the presence of real world uncertainty, for example by using agile engineering methodology. Generally, students much prefer to collaborate than to work alone, and computing can give them the disciplines and methods they need to work successfully on large uncertain projects, and in collaborative teams.
- **Computational Thinking**. How can one accurately simulate a system consisting of millions of objects evolving over billions of steps? How does one prevent a computer from creating many thousands of e-mail accounts that can be used to send spam to millions of people? How can one design an electronic auction system that fairly represents the interests of all parties involved? How can one be certain a program will perform correctly, in life critical systems such as avionics or medicine? To deal with these problems, and many more similar ones, requires a type of thinking characteristic of computing: computational thinking. Computational thinking involves a clear focus on tangible problems; a large collection of proven techniques such as abstraction, decomposition, iteration, and recursion; an understanding of the capabilities of humans and machines alike; and a keen awareness of the cost of it all. Emphasis on computational thinking rather than just programming has greatly improved introductory courses and is starting to become a motivating principle in other parts of our curriculum.
- **Computers and Biology.** Computing has become essential for solving biological and biotechnology problems. In molecular biology, for example the fragment assembly problem was a central computational task in sequencing the human genome. This problem is a nice vehicle for introducing the fact that some computational problems seem to have no efficient solution—a deep insight of computing.

6 Opportunities

Although we have painted a gloomy picture, there are reasons to be hopeful. This section articulates some of the opportunities that we see.

6.1 The Key Stage 3 opportunity

The biggest opportunity at KS3 is the enthusiasm and curiosity evident in 11 year olds, and the abundant evidence that they are able to grasp the concepts and practices of programming. They represent the "digital generation" and take for granted the role of computers in all day-to-day activity. They may have had some exposure to the use of computers at a home and simple programming, perhaps with LOGO and floor turtles at KS2 in primary school.

Furthermore, the National Curriculum Programme of Study at KS3 is admirably non-prescriptive. Much that we have described as "Computing" could readily fit within it, and some is mandated (e.g. "Use ICT to make things happen by planning, testing and modifying a sequence of instructions, recognising where a group of instructions needs repeating, and automating frequently used processes by constructing efficient procedures that are fit for purpose") .

A third opportunity concerns the nature of computing: it is not a dry academic discipline, but rather one where students can write programs that embody and animate their ideas. This is tremendously motivating, an opportunity that is wildly under-exploited today. Moreover, there are a number of programming environments aimed specifically at engaging the interest of young students, while helping them to learn algorithms, data structures, and generic techniques like problem-solving and abstraction. Prominent examples are Alice[12], Scratch[13], Kodu[14], and Greenfoot[15].

6.2 Guerilla outreach

There is much exciting and innovative work going on, across the UK and elsewhere, to encourage the delivery of Computing at school. Examples include:

- **Cs4fn**[16], a a magazine produced at Queen Mary and Westfield College. *"Welcome to the fun side of computer science! Computer Science is no more about computers than the music industry is about microphones. Explore how computer science is also about people, solving puzzles, creativity, changing the future and, most of all, having fun ..."*. Cs4fn is a major activity: they distribute 20,000 hard copies of each issue of the magazine world-wide, the web site gets 15 million hits/yr, and the team presents live shows to thousands of students each year.
- **CSinside**[17], a programme of workshops for students and teachers, run by the University of Glasgow, funded by EPSRC. Also **Computing Scotland**[18], is "a portal website, from a student's perspective, for information about computing and

[12] http://www.alice.org/
[13] http://scratch.mit.edu/
[14] http://research.microsoft.com/en-us/projects/kodu/
[15] http://www.greenfoot.org/
[16] http://www.cs4fn.org
[17] http://csi.dcs.gla.ac.uk
[18] http://computingscotland.org/home

computing degrees and careers available in Scotland and wider". The main message is that computing is a vibrant career path.

- **CS unplugged**[19] is a *"is a collection of activities designed to teach the fundamentals of computer science without requiring a computer"*. Developed at the University of Canterbury, New Zealand, it has proved very popular across the world. It is aimed at 5-12 year olds, but is entirely appropriate for the age range up to 18 in UK.

[19] http://csunplugged.com

7 Recommendations

7.1 Computing should be recognised as a STEM subject

Computing is a rich mixture of Science, Technology, Engineering, and Mathematics; if any subject is a STEM subject, Computing surely is.

The 2008 CRAC report "Do undergraduates want a career in IT?[20]" puts the point well: "*To us it seems self-evident that the arena of IT and telecommunications (ICT) is one of the knowledge- and technology-intensive sectors of industry mentioned above, and therefore ensuring that people enter relevant education and subsequently jobs in that sector is important. However, we find it puzzling that current Government policy does not always appear to recognise Computing/IT as a STEM subject in terms of provision of additional funding, so many STEM initiatives do not include Computing/IT. The Government also provides funding support to certain 'Strategically Important and Vulnerable Subjects' (SIVS), but Computing/IT appears to be recognised as being strategically important but not vulnerable, so is apparently not eligible for support or initiatives under that scheme.*"

Being recognised as a STEM subject will not transform Computing overnight, but it will help to align it with the disciplines of science rather than with bureaucracy and business processes.

7.2 The need for a consistent curriculum and body of knowledge

One of the criticisms of ICT is that the same material is repeated. In computing whilst excitement is generated at primary level with floor turtles and the like, it is dissipated in secondary school, and the subject is taught again from scratch at HEI, as no reliance can be placed on prior knowledge. In the USA and other countries there are well established bodies of knowledge and curricula, such as the ACM K-12 CS model curriculum. The UK needs to develop its own approach, defining expectations for computing education at each Key Stage.

7.3 A GCSE in Computing

There should be a GCSE in Computing, with its own, QCA-approved Subject Criteria.

The new A level in Computing is a step in the right direction, but it is hamstrung by the absence of a GCSE in Computing. After five year of education that computers = spreadsheets and word processing, many students have simply ruled out computing as subject of interest. The absence of a GCSE in the subject strongly reinforces this perception. It also has a knock-on effect at Key Stage 3: it is hard for a teacher to present material on Computing, because in school terms it leads to nothing. In these days of league tables, this is a killer blow.

We make the case for a GCSE in Computing in a companion paper[21].

7.4 The need for training and resources

In our discussions we have found many, many teachers who are highly motivated to teach computing, and are desperate for the teaching material and institutional framework that

---

[20] http://www.crac.org.uk/crac_new/pdfs/undergraduates_it.pdf

[21] http://www.computingatschool.org.uk/files/GCSERationale.pdf

would allow them to do so. At the 2009 teacher's conference[22] organised by the Computing at School Working Group, a member of an exam board expressed concern that teachers would be unable to deliver a GCSE in Computing. A teacher responded, to widespread applause "We are *teachers*; we can learn, and we want to".

We need

- **Teaching materials for Computing** that teachers can use "off the shelf", much as they currently use material from the National ICT Strategy resources. At Key Stage 3 there is considerable flexibility in the required provision, but it is seldom exploited because there are no resources for avenues other than mainstream ICT.
- **A national programme of Continuing Professional Development for Computing teachers**. Teachers are eager to learn; universities will help with material; but scaling up existing local efforts to a national programme is a big step. The human resources available are the single most important factor in determining the quality of the curriculum delivery. Given that there is the flexibility in the curriculum to teach Computing, and the statutory programmes of study do not mention particular applications never mind particular products, why is such a narrow interpretation adopted in so many schools? The answer has to lie in the skills and knowledge of teachers often reinforced by popular pressure from the media and external influences. Many teachers are co-opted into teach ICT as simple ICT users rather than professionals with training in computing. This is the equivalent of mathematics being taught by teachers who have the arithmetic skills to get by in everyday life. Given that £500m was invested in ICT teaching resources through Curriculum On-line, if investors in people means anything, similar amounts need investing in ensuring teachers have the skills and knowledge to deliver the hi-tech understanding and transferrable skills a twenty first century economy demands.

[22] http://www.computingatschool.org.uk/files/conference2009/report_noemail.pdf

8 Appendix: The Computing at School Working Group

Computing at School is an informal Working Group, consisting of individuals concerned to promote the discipline of computing at school.

- It is an open group: anyone can join
- It has broad representation, including school teachers, university academics, members from industry, members of professional societies, and exam boards.
- It is non-partisan; CAS is not there to promote the interests of any particular group (e.g. the universities, or employers), but rather to argue for the discipline as a whole.

The CAS Working Group is active at many levels, including developing a Body of Knowledge for school-level computing; in spinning up “hubs” that bring teachers together in local groups; and in developing new material that teachers can use in the classroom, especially at Key Stage 3.

More details can be found at http://www.computingatschool.org.uk/.

9 Appendix: Summary numbers

Figures for GCSE in ICT (from JQA): 2003: 92,054, 2004: 98,833, 2005: 103,400 2006: 109,601, 2007: 99,656 2008: 85,599 2009: 73,519