LCERS
London Computing Education Research Symposium
Monday 11th June 2018

King’s College London
Welcome from the organisers

King’s College London is delighted to host the first research symposium as part of the Computing Education Research Centre. Our Centre is new and was only established this year; it gives us an opportunity to bring together academics from different disciplines with a common interest in computing education, and we are excited about starting to work together. We are interested in computing in a broad sense, for all ages, and in informal and formal settings.

Research in computing education is a young field with strong ties to national educational systems. There has been a steady stream of research focused on higher education and how students at university learn computer science, and how technology has impacted education. Back in the 1980s there was a considerable interest in school computing education, with the influential work of Seymour Papert, Logo and Basic being taught in school, and early personal computers such as the BBC Micro. Fast forward thirty years, the move in England to make computing a mandatory part of the school curriculum means that computing is no longer just for those who choose it at university, or those who choose an extra-curricular club, but computing is for all. We think this is important and that’s why it became the theme of our first research event.

We were very pleased to have a good number of submissions to this conference. We hope you will enjoy the 8 presentations and 14 posters you will see today. In addition, we are delighted to welcome two fantastic keynote speakers: Professor Ralf Romeike, one of the founders of WIPSCE, an established international conference, who will talk about a range of aspects of computer science education research, and Carrie Anne Philbin, award-winning author and video presenter, who leads CAS Include, who will talk to us about challenging diversity. We hope you enjoy the day!

Michael Kölling and Sue Sentance
June 2018

About CERC

At King’s College London we have a substantial co-location of computer science, digital humanities and education researchers in computing education, creating an ideal opportunity to establish a focal point for this area. As projects in this area typically benefit from expertise in a range of disciplines, it could be said that no one department on its own is a comfortable home for computing education research. We are thus setting up an interdisciplinary institute to enable collaboration, sharing of ideas and truly integrated work between researchers in all three departments. We hope to have more events like this one and work together on new and innovative research projects.

The CERC team are: Michael Kölling, Neil Brown, Alan Cribb, Mark Coté, Paul Curzon, Lulu Healy, Simon Humphreys, William Marsh, Jennifer Pybus, Sue Sentance, Elizabeth Sklar, Mary Webb, Steffen Zschaler

Administrative team: Carla Cretan & Erk Gunce
Programme
All presentations will be held in the Bush House Lecture Theatre 3, BH (NE) 0.01.
All refreshment breaks and the poster session will be on the 1st floor in the Exchange area.

<table>
<thead>
<tr>
<th>Time</th>
<th>Session/Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>09:00</td>
<td>Arrival and coffee</td>
</tr>
<tr>
<td>09:30</td>
<td>Introduction and welcome</td>
</tr>
<tr>
<td>09:35</td>
<td><strong>Keynote 1</strong>: Professor Ralf Romeike: From technical to didactical innovations in the computer science classroom: Understanding and creating with computer science (Chair: Sue Sentance)</td>
</tr>
<tr>
<td>10:30</td>
<td>Coffee break</td>
</tr>
<tr>
<td>10:50</td>
<td><strong>Session 1</strong>: Curriculum and content (Chair: Michael Kölling)</td>
</tr>
<tr>
<td></td>
<td>The impact of the new Computing curriculum on subject choices at GCSE Level</td>
</tr>
<tr>
<td></td>
<td><em>Eleanor Overland and Cathy Lewin, Manchester Metropolitan University</em></td>
</tr>
<tr>
<td></td>
<td>Using children’s literature as a springboard to teaching early computing concepts: a design-based research methodology.</td>
</tr>
<tr>
<td></td>
<td><em>Sarah Twigg and Lynne Blair, Lancaster University</em></td>
</tr>
<tr>
<td></td>
<td>A pilot study exploring spatial skills training and computer science outcomes</td>
</tr>
<tr>
<td></td>
<td><em>Jack Parkinson and Quintin Cutts, University of Glasgow</em></td>
</tr>
<tr>
<td></td>
<td>Unlocking the Secrets of the Binary Box</td>
</tr>
<tr>
<td></td>
<td><em>Duncan Maidens and Andrew Csizmadia, Birmingham City University, Newman University</em></td>
</tr>
<tr>
<td>12:30</td>
<td>Lunch</td>
</tr>
<tr>
<td>13:15</td>
<td><strong>Session 2</strong>: Equality, Diversity and Privacy (Chair: Lulu Healy)</td>
</tr>
<tr>
<td></td>
<td><em>Catherine Elliott, Sheffield County Council</em></td>
</tr>
<tr>
<td></td>
<td>How Freya built Sharkie: Initial explorations into the safety, security and privacy concerns of children's IoT devices.</td>
</tr>
<tr>
<td></td>
<td><em>Sophie Beck, Joe Finney and Bran Knowles, Lancaster University</em></td>
</tr>
<tr>
<td></td>
<td>Inspiring more girls into Computer Science</td>
</tr>
<tr>
<td></td>
<td><em>Martine Mannion, Nottingham Trent University &amp; Wellingborough School</em></td>
</tr>
<tr>
<td></td>
<td>Collaborative Creative Computing</td>
</tr>
<tr>
<td></td>
<td><em>Steven Bradley, University of Durham</em></td>
</tr>
<tr>
<td>15:00</td>
<td><strong>Session 3</strong>: Poster session (Exchange, 1st floor)</td>
</tr>
<tr>
<td>15:30</td>
<td>Coffee break (Exchange, 1st floor)</td>
</tr>
<tr>
<td>16:00</td>
<td><strong>Keynote 2</strong>: Carrie Anne Philbin (Chair: Sue Sentance)</td>
</tr>
<tr>
<td>16:45</td>
<td>Announcements and wrap up</td>
</tr>
<tr>
<td>17:00</td>
<td>Finish</td>
</tr>
</tbody>
</table>
Our keynote speakers

Professor Ralf Romeike
Friedrich-Alexander University Erlangen-Nürnberg

From technical to didactical innovations in the computer science classroom: Understanding and creating (with) Computer Science

In Computer Science, continuous scientific developments harbour significant potential for the further development of teaching computing at schools. By presenting several research projects, the talk will discuss approaches for harnessing the possibilities of new topics, methods, and tools for better teaching and learning in the CS classroom.

About Ralf

Ralf is Professor for Didactics of Informatics at the Friedrich-Alexander University Erlangen-Nürnberg in Bavaria. His research interests include the use and encouragement of creativity in computer science education, and the use of agile methodologies. In 2012 he established the international conference WIPSCE, which is unique in focusing on primary and secondary computing.

Carrie Anne Philbin
The Raspberry Pi Foundation

Challenging diversity in computer science one 30-minute presentation at a time

STEM subjects in education have traditionally been dominated by middle-class white men. How can we create a more inclusive classroom to stop students from a more diverse background dropping subjects like computer science in KS3?

About Carrie Anne

Carrie Anne is the Director of Education at the Raspberry Pi Foundation and the Chair of CAS Include, a Computing At School group to support inclusion and diversity in computing in school. She was recently voted the 3rd most influential woman in UK IT in 2017 by Computer Weekly.
Presentations

Session 1: Curriculum and Content

The impact of the new Computing curriculum on subject choices at GCSE Level
Eleanor Overland and Cathy Lewin, Manchester Metropolitan University

Since the introduction of the National Curriculum in Computing in 2014, the implementation across the UK remains patchy and fragile (The Royal Society, 2017). Although 70% of pupils attend schools where GCSE Computer Science is an option, only 11% of pupils currently study it at this level.

The rapid introduction of the curriculum change has faced many problems. There is a shortage of specialist staff with the confidence to teach computer science across all key stages and indeed the number of trainees in this area is falling (Webb et al., 2017). As a result, not all schools offer GCSE Computer Science and where the subject is offered, there are significantly more boys than girls opting for GCSE and A-level Computer Science (Kemp et al., 2016). Therefore, it is imperative to develop a better understanding of what shapes young people’s decision-making in relation to subject and qualification choices for school-level computing qualifications given the limited research on the impact of this curriculum change to date and the pressing need to ensure digital skills for all. The curriculum changes could be constraining rather than opening up access to digital skills development for various reasons.

This project, currently in progress, aims to explore the impact of the new Computing curriculum on students’ attitudes to the subject, GCSE choices and career aspirations.

Our research questions are:

1. How do students position themselves in relation to Computing as a subject?
2. What factors influence whether a student chooses to study GCSE Computer Science and/or other qualifications in digital skills at Key Stage 4?
3. How does students’ experience of Computing relate to career knowledge and aspirations?

Two secondary schools in the North West, one urban and one rural, have been identified as case study schools; both schools currently offer GCSE Computer Science. In these schools, all students in their last year of Key Stage 3 (either in year 8 or year 9) will take a 15 minute survey and complete a mind mapping exercise. In addition, 4 students who have chosen to study GCSE Computer Science and 4 students who have been interviewed. Additional interviews are being conducted with class teachers, the head of department and a member of staff with responsibility for careers information, advice and guidance. Relevant document such as options evening booklets and photographs of displays are also being gathered.

Data will be analysed through a sociocultural view of identity and agency, drawing on Holland and colleagues’ (2001) Figured Worlds. Figured Worlds views identity as fluid rather than fixed, with beliefs and behaviour strongly influenced by cultural models and narratives. Using this approach allows us to have a more nuanced understanding of how young people position themselves and are positioned by others such as teachers and parents. Identity plays a significant role in young people’s choices of qualifications and careers relating to computer science depending on whether or not they consider themselves to (not) be resilient, clever, anti-social, curious and creative (Wong, 2017). Similarly, young people may be influenced by schools, teachers, family, friends and the media. For example, in one of our case study school students are told that they need to be good at mathematics to do GCSE Computer Science. In this study, we try to present a balanced view by collecting data from young people who have interests in Computer Science, those who have interests in vocational ICT courses and those who do not. Particular attention will be paid to gender differences. We will report on the emerging findings from this project.
Using children’s literature as a springboard to teaching early computing concepts: a design-based research methodology
Sarah Twigg and Lynne Blair, Lancaster University

With the recent inclusion of Computer Science in many countries’ school curricula, including targets for younger children (e.g. England’s KS1-2 for ages 5-11), a particular need has arisen to explore innovative, creative and pedagogically sound methods to teach computing concepts to young children.

This research presents a creative children’s literature-based approach for engaging young children (KS1-2) in computational thinking and programming skills. At the heart of this approach is the recognition that many books already on classroom shelves exhibit the programming principles of sequencing, repetition and selection in creative ways, as well as supporting computational thinking skills such as logical thinking, decomposition and pattern matching. Our approach is iteratively developed through a design-based research methodology that is practice-based and situated in an educational (school) context.

To select books, we used a large online catalogue of children’s books from a popular UK bookseller (http://thebookpeople.co.uk), which was searched and filtered according to the following criteria: KS1 reading abilities (up to age 7), familiar children’s story books (classic fiction), picture books, single books (cf. collections or multiple books), and no duplicates. This filtering produced a list of 50 books, each of which was read and coded according to whether the storyline exhibited the three main programming constructs:

- Sequencing – most, if not all, books will illustrate this construct since stories generally move through a basic sequence of plot stages.
- Repetition – a pattern of repeated dialogue and/or actions in different contexts.
• Selection – dialogue and/or actions following a repeated pattern, but where changes occur based on the context; or alternatively a test of context to determine whether a desired goal is reached (e.g. termination).

From the 50 books, 2 exhibited the pairing of Sequencing and Selection, 6 exhibited Sequencing and Repetition, and 16 books exhibited all three programming constructs. These 16 books became our ‘working set’ of books for the iterative and participatory design of a set of learning resources.

Our design-based research methodology (Ethical approval for this project has been granted by our University Research Ethics Committee) was set up as follows. Focus groups were chosen based on geographically clustered groups of up to 8 primary school teachers with associations to the researchers’ Computing At School (CAS) Regional Centre. Initial questions to participants sought to reveal the current challenges of teaching computing concepts to young children (KS1-2). The children’s literature-based approach was then presented to participants, including a set of prototype lesson plans and resources. The lesson plans aim for a gradual build-up of subject knowledge, supported initially by Unplugged [2] activities. Sequencing is introduced through an activity to sequence Scratch Vector Scratch Blocks [3] and/or images from the storyline, Selection is introduced by identifying sections of the story that only happen due to a certain event, and Repetition requires the children to identify any repeated sections of the storyline and also identify the terminating condition. Finally, moving on to a Scratch editor [4], pairs of children work together to drag and drop (pre-loaded) blocks to create an executable storyline. Through each of these stages, computational thinking concepts such as patterns, algorithms, abstraction and decomposition are identified, along with opportunities for tinkering, creating, debugging, persevering and collaborating [1].

In the focus groups, participants discussed the prototypes, identifying opportunities and challenges to using them in their own classroom contexts. Participants were asked to take the resources to adapt or develop for use in their own classrooms. A follow-up interview was conducted to gather details of the adaptations and feedback, including consideration of learning progression and any necessary scaffolding. Through these steps, we claim that this innovative and creative idea has been evaluated and honed into a pedagogically sound approach for the teaching of computing concepts to young children.

References


A pilot study exploring spatial skills training and computer science outcomes
Jack Parkinson and Quintin Cutts, University of Glasgow

Spatial skills (SS), also known as spatial reasoning, have been connected with STEM and computer science (CS) success for over sixty years [6]. While in some fields, notably engineering [5], this relationship has been well explored and exploited, there is limited research investigating the relationship between SS and CS [1,2,3]. While existing research indicates a correlation between SS and CS success, a cognitive model explaining this relationship has not been proposed. Unlike Sorby's work in engineering, we do not believe there is sufficient evidence to launch into full-scale SS training for our students, but yet are tantalised by the possibility that this may be effective in improving CS skills, enough to warrant a small-scale pilot study.

The aim of our work on SS therefore is to provide a more solid foundation concerning the connection between them and CS for future studies. We survey the literature on SS and investigate the various underlying cognitive skills involved. We postulate a model for the relationship between CS ability and CS, exploring ways in which the cognitive processes involved in each overlap, and hence may influence one another. In our model, SS are not directly related to CS, but practice in either, as well as in other STEM subjects, improves underlying cognitive skills of value to all these areas. Such a relationship is supported by a study involving physics students [4].

Strengthening the case for a connection between SS and CS, our earlier work, under review, shows that SS increase as the level of academic achievement in CS increases. Final year students and academic staff in CS have significantly greater SS than first year students, with a large effect size.

Further, we present here the findings of a pilot study, replicating at a smaller scale the kind of SS training course run by Sorby in engineering. The purpose is to investigate the effects of SS training on CS students and identify the practical considerations which need to be made for running such a training course on a larger scale. In our study, 113 students in a first programming module took SS and programming assessments at mid-term. 52 students failed the SS test and were invited to take an accelerated SS training over four weeks with on-line and off-line components, with 18 taking part. Intervention students’ who retook the SS test demonstrated significantly increased skills. Considering class rankings between the mid-term programming assessment and another at the end of the course, all CS students who attempted the SS training moved up the programming ranking significantly compared to those who had failed the SS test but did not take the training; those who did not take the training showed on average no significant movement in either direction.

While only small scale, this pilot study supports larger scale experiments to determine whether SS training can lead to improved CS skills. Our model may be interpreted by some to argue that the skills we are discussing are all just part of a general intelligence quotient. It is immaterial whether this is the case, as long as we accept that aspects of intelligence can be developed. It is certainly the case that SS can be improved with training, and students in our study report the training to be enjoyable. Furthermore, the number of repetitions can be raised significantly via a SS course over, say, the number of problems solved in a programming lab, and so the development of the underlying skills accelerated. Hence, if our model holds, the application of SS training may be an effective way to improve outcomes in CS, as well as other STEM disciplines.

References

Data communications underpins the computing curriculum taught in both English primary and secondary schools and influences and impacts the way in which society works, lives and plays. However, despite its pivotal role in computing, it is often regarded as the Cinderella of the curriculum. Taught data communication sessions in schools can range from rather dry and unengaging theory lessons through to hands on, but somewhat disjointed, unplugged activities. In order to ensure this Cinderella takes their rightful place on the computing stage, the authors planned, prototyped and produced the Binary Box. The Binary Box is a tangible physical resource, for teaching the principles and concepts of data communications. Additionally, a series of associated teaching and learning activities to engage, educate and enthuse learners about data communications were developed.

In this paper, the authors will share their trails, tribulations and triumphs of the initial design and continuing development of the Binary Box which has spanned a three-year period. Additionally, they will share the theoretical frameworks which have influenced and guided their thinking and trajectory of work. They will reveal the impact of utilising the Binary Box as a teaching and learning tool, and future developments of this resource.

The authors will outline how the Binary Box was designed with the constructionist’s approach of “low ceiling, high floor and wide walls” (Resnick, 2017) and the concept of the notional computer (du Boulay, O’Shea and Monk, 1989) guiding them. Over a three-year period, they have designed, tested and refined this tangible physical artefact and created a series of associated teaching and learning activities.

These scaffolding activities have been designed in accordance with the principles of Creative Learning (Resnick, 2014) in order to introduce the principles of data communication in Key Stage 1, develop these principles in Key Stage 2, and introduce and reinforce more complex topics at Key Stages 3 and 4. The data communication principles that learners are introduced to through discovery
include: binary representation, coding schemas, sequencing data, serial transmission, Transmission Control Protocol (TCP), encryption, image coding, bit rates, eavesdropping and man-in-middle attacks. In addition, the activities were designed to develop computational thinking, promote computational learning and encourage computational participation as learners collaborate to solve an open-ended problem, they are presented with. To ensure coverage each activity has been mapped to the Progression Pathway framework and categorisation system for mapping computer science content and computational thinking (Dagiene and Sentance, 2016).

The delivery of each activity aims, not to initially teach the underlying data communication principle associated with that particular activity, but to lead the learner, working collaboratively, through a process of investigation and ‘trial and error’ to unlock the underlining principle. This is in order that they appreciate the need for both a communication system and an agreed data communication standard. Then, using practical, hands on (unplugged) methods, the learner discovers the solution, experientially learns the underpinning data communication principle and engages in a meaningful discussion regarding that principle.

The authors will not only present a critical reflection of the Binary Box using the lens of Creative Learning. They also will report on the impact of the Binary Box as a teaching tool for learners, as a tool for continuous professional development for in-service computing teachers and its adoption by the computer science education community in England.

In conclusion, the authors will outline future developments of the Binary Box and potential research to investigate the impact of the discovery-based approach adopted in the associated teaching and learning episodes.

References


Session 2: Equality, Diversity and Privacy

Catherine Elliott, Sheffield County Council

Computing replaced ICT as a subject in the National Curriculum in England in 2014 to better equip young people for living and working in an increasingly digital world. Computing is essential for learners with special educational needs and disabilities (SEND) who require technology in order to influence and access the world around them. We should also be teaching SEND learners appropriate digital skills to provide them with the same "economic and social mobility opportunities as their peers" (Wille, Century & Pike, 2017, p.41).

This study investigates the current picture of computing provision in special schools and similar settings. It also collates the resources and strategies that are considered successful in teaching the subject to SEND learners in order to help teachers improve their practice. Although aimed at supporting practitioners in specific SEND settings, it can also inform teachers working in mainstream provision, where over a million of young people with special needs are currently educated (DfE, 2017). Furthermore, good practice in teaching computing to students with SEND can benefit all students.

A survey of teachers in special schools and other SEND settings in England was conducted in the autumn of 2017. It was advertised through targeted emails, social media, on the Computing At School website, and by word of mouth. This resulted in 65 responses representing approximately 6% of special schools, of whom 71% were computing or ICT lead teachers. Follow up interviews were then conducted with a small number of teachers to gather more detail. On the whole the responses paint a positive picture of the subject in these schools, with 66% of teachers rating their confidence in teaching the subject highly. There were also indications of a great deal of innovative work being done to adapt the curriculum for the specific needs of the young people. It must be noted, however, that fewer than 50% of the schools surveyed are teaching the 2014 Computing curriculum. There are a number of challenges in delivering the curriculum, many of which are common to teachers in mainstream settings (Sentence and Csizmadia, 2016). The largest barrier, however, is the lack of SEND specific resources. Content is often aimed at too high a level or is not age-appropriate for students working well below age expectations.

A number of pedagogical approaches were identified by the teachers as beneficial to teaching computer science around the following themes: unplugged activities, relevant personalised tasks, physical computing, and the chunking of tasks into smaller parts. Further research is required into the effectiveness of these approaches for students with a range of needs in terms of reducing cognitive load and providing multi-modal means of access.

A final aim for the study was to investigate the teachers’ perspective on the relevance of the computing curriculum in their setting. Around 65% considered computing or ICT relevant for their learners, notably where students rely on technology for communication and access to learning. It is also an area that engages these young people, and where many individuals can excel. The teachers were split in their opinion about the relevance of the computer science elements, with many teaching in settings where the cognitive level of the students precluded meaningful programming activities. However, the more experienced and confident computing teachers consider these aspects more relevant and have been using computational thinking as a framework for teaching problem-solving across the curriculum. There is much debate as to the transferability of computational thinking skills (see for example Tedre & Denning, 2016), but the responses indicate a number of benefits of this approach for learners with specific communication and learning disabilities and further investigation would be recommended.
How Freya built Sharkie: Initial explorations into the safety, security and privacy concerns of children’s IoT devices.
Sophie Beck, Joe Finney and Bran Knowles, Lancaster University

There are many benefits for children interacting with Internet of Things technology, which is exemplified by the popularity of physical computing in education. However, there is a need to explore the risks that are likely to emerge as children become ever-more exposed to IoT technology [1,2]. Through exploring the possible privacy, safety and security risks that children might encounter, new ways in which children can be educated around these risks can be identified.

This presentation reports on the findings from the first stage of IoT4Kids, which is a one year PETRAS project in partnership with The Micro:bit Foundation, NSPCC and FOSI. The project draws on children’s imagined uses of the BBC micro:bit, which is an IoT device designed with the intent to engage children with programming and computing. It will also highlight how these findings are informing future research.

This presentation highlights the key findings of two workshops with primary school children (aged 9-11) which captured the children’s imagined uses of the micro:bit and a Team and Stakeholder Workshop, which explored the children’s use cases in relation to how children might realise some of their designs and possible risks which may emerge from children building their designs.

The data analysis from the first stage produced three high-level use categories; Assistance, Play and Companionship. Within each high-level use category, two initial speculative use scenarios were developed in order to inform the second stage of the project. These speculative use cases drew on the data collected from the events, as well as desk research drawing on current knowledge on known risks regarding online safety for children [3,4]. This resulted in the production of narratives of
imagined children which included; their motivations for building an IoT device, how they built their design, and the possible risks that they might encounter during the build and use of their device.

Our study highlighted that a high proportion of children desire assistance technology, most likely influenced by their exposure to current IoT technologies. Whilst some of the more simplistic designs pose low risks, there was a proportion which highlighted children’s desires for tracking and surveillance. A number of girls highlighted the desire for technology to be utilised for companionship purposes, combatting loneliness and provide a mechanism for emotional support. These designs related to capabilities available in IoT ‘connected toys’, whereby a child’s favourite toy will respond to their questions and emotions.

A smaller proportion of boys highlighted their desire for risky play. Such examples included pranking, whereby IoT devises could be utilised to elicit fear in others, such as switching lights on and off in order to scare siblings. Based on the findings, use scenarios have been developed such as Freya who builds ‘Sharkie’ her own connected toy. Through this process a number of possible risks have been highlighted to explore further in the second stage of the project.

Future work aims to draw on this current study in order to design and create workshops-in-a-box. The aim of these resources will be to educate children around the potentials that they might encounter whilst programming IoT devices. These will include educational resources to support teachers in the exploration of IoT technologies when delivering the computing curriculum. The vision is to develop innovative approaches for delivering physical computing education, whilst raising awareness of safety, security and privacy concerns associated with children’s interaction with IoT technologies.

References


How do we inspire more girls into Computer Science?
Martine Mannion, Nottingham Trent University & Wellingborough School

With the ever-widening Computer Science (CS) industry ‘gender gap’, despite being in the age of the ‘digital native’ (Prensky, 2001), experts predict females pursuing a CS career will continue decreasing. My research investigates what drivers would inspire more UK girls to take GCSE Computer Science (DfE, 2015). Statistics demonstrate since 2015 a yearly decline, with only 20% female students taking GCSE-CS in 2017, indicating barriers including gender stereotypes, accessibility or un-inspiring content. Being a female with 20+ years in the IT industry, now a Computing teacher and Regional Hub Leader for CAS, I am only too aware of the critical need to make computing accessible to all. The ‘After Reboot’ report (The Royal Society, 2017) makes several recommendations including teacher-led research into improving female participation in CS.

I collected data via electronic surveys, informal interviews and observations, identified emerging trends from girls in Years 3, 6 and 9 at a Northamptonshire co-educational school into what would inspire them to study CS. Comparing these findings to recent national surveys including ‘Girls’ Attitudes’ by The Girl Guiding Association (2018), three strong themes emerged; A clear appetite for using technology in creating innovative solutions to everyday problems. Being able to work collaboratively on projects using technology. A passionate plea to promote more female role-models to champion girls in CS. Reflected in my Year 6 & 9 sample groups, the 12 participants identified potential career aspirations of, combating cyber security threats, developing robotic devices, building specialist software applications for the AI/VR markets.

The girls surveyed also grasped the importance of protecting personal data, a need to understand ‘big data’ and the wider connotations which GDPR seeks to address. Applying similar methodologies, a small sample group of 32, combining parents and females working, researching or teaching in the field of CS, focused around the GCSE-CS syllabus and accessibility of CS outside school. Their viewpoints highlighted core interventions. Female Industry representatives surveyed insisted the GCSE-CS should be more stripped back, removing the technical content relating to specialist CS areas, as this should be nurtured in a commercial situation. Indeed, the whole group felt the current Computing curriculum including GCSE-CS fell short in providing girls the opportunity to explore their creative and critical thinking skills, with too much emphasis on programming, algorithms and pseudocode. It’s small wonder then why we are seeing this annual decline.

Other interventions included tapping into the emerging school’s STE(A)M movement, providing the mechanism to integrate CS, as passionately campaigned by Wing (2010) ‘putting the ‘C’ (computer science) into ‘STEM.’ With affordable easy to use resources, including the BBC Microbit central to many STEM projects, would not only compliment CS but enable the GCSE-CS to critically engage girls creatively, developing core computational thinking (CT) skills industry is crying out for.

Although The Royal Society (2017) considers ‘many out-of-school STE(A)M club schemes exist and have been shown to be effective’, parents’ data highlighted a lack or no accessibility to STEM / Coding clubs in their wider community, with the consensus such initiatives would provide vital pathways where schools don’t offer such opportunities to complement a CS curriculum, raising awareness of the subject and towards further study. Indeed, with sufficient support club leaders of established associations (Girl-Guides) could incorporate ‘STE(A)M’ activities.

In conclusion my findings propose shifts to be investigated further establishing an ‘OPP’; overhauling, not just GCSE-CS but all GCSE STE(A)M related subjects, thus a theory of quality rather than quantity, with CS a natural embedded synergy within wider curriculum areas, promoting, awareness of inclusivity to disperse the stereotypes attached to CS via a rebrand unified approach of champions on all levels, providing, teachers and club leaders to resources and a rolling CPD programme which is manageable and achievable within a continuing cycle of pedagogical self-review.
References


Collaborative Creative Computing  
Steven Bradley and Stephen Church, University of Durham

1 WHY?  
We all know that there are not enough girls and women in computing, which is disadvantageous both to women and to computing. In the UK in 2014 the proportion of girls choosing to study GCSE computing (14-16) was only 15%[1], and similarly low (17%) on entry to university in 2015 [4]. This trend appears start between the ages of 11 and 15 [5] and although there is no one simple answer, a large part of the problem appears to be in the existence and perpetuation of negative stereotypes of the culture [2] such as:

- socially isolated  
- not collaborative  
- masculine interests

In our work we focus particularly on workshops for primary or early secondary school pupils (9-13), and in particular on the collaborative creation of music using computers. Our approach is to focus less on the syntax of the code and to think more about the structure of the network: both the logical network that governs which notes are played when, and the physical network and related protocols that join the different components together. In this way we hope to be able to address not just coding issues, but broader computer science issues that we discuss below.

2 WHAT?  
"Node-RED is a flow-based programming tool, originally developed by IBM’s Emerging Technology Services team and now a part of the JS Foundation." [3]. Designed as a tool to allow quick and easy development of Internet-of-things applications, it allows integration between internet services such as http requests, email, social media, and to connect them to physical devices by a range of protocols. Editing takes place in a web browser, without the need for code to be typed in (although nodes in the flow network can be defined with arbitrary javascript functions). The code itself runs on a nodejs server, which can either be deployed remotely or locally. We have defined new node types to facilitate music-making in node-red:

- beat: which generates regular beats, like a metronome  
- divider: which allows beats to be aggregated into bars, bars into phrases, phrases in sections etc  
- sequencer: which plays rhythms based on the beats that it receives  
- synth: which creates SuperCollider messages to play sounds (to be forwarded via OSC)  
- sampler: which allows sounds to be recorded and played back, creating a sophisticated programmable “loop pedal”.

These nodes can then be combined with standard node-red features, including browser-based UI components, to make beautiful music! Figure 1 shows how some standard node-red components can be combined with custom music nodes to generate a simple drum beat with a volume and speed control.
3 RESULTS

So far we have used the system in schools workshops three times, including primary and secondary. The most recent workshop was run as part of a Y7/8 after-school club and had an enforced 50/50 gender split, resulting in a group size of 12. We used a self-completed questionnaire at the end of the session to gather quantitative and qualitative opinion data from the students on technical and motivational aspects of the workshop.

Students were unanimously strongly supportive of the statements “It was fun to create music” and “I learnt new things using this system” with more variability around “node-RED was easy to use” and “It was simple to create music”. Looking at the qualitative data the main thing that the students liked about the session was the chance to experiment and make their own music. We have larger-scale events planned, which should support more detailed analysis of the impact and its variation across gender.

REFERENCES


1. **Widening the Frame: Adding Stride to BlueJ**  
Amjad Altadmri, Neil Brown, Hamza Hamza, Michael Kölling and Davin McCall  
King's College London

Frame-based programs editing is a blocks-like paradigm, with usability improvements from text-editing. This fusing removes awkward aspects of manipulating larger and more complex programs, ‘raising the ceiling’ of the size and scope of programs that can be written with block-like interfaces. Implementing it into a general-purpose educational IDE, BlueJ, opens up this paradigm to a broad range of educators who are looking to target a wider variety of domains, but do so with a more approachable user interface than text-based programming.

2. **PRIMM and Paired Programming**  
Roger Carpenter and Priyadarshini Bakre, Tiffin Girls School

Our poster describes a practitioner research project to encourage girls to investigate and discover programming through ‘tinkering’. We introduced a four-lesson module based on principles of PRIMM Predict-Run-Investigate-Modify-Make methodology along with “pair-programming” to explore whether these methodologies will work at KS4 level.

3. **The Computing Curriculum: A survey to investigate what student teachers observed or taught during their last Block School Experience (BSE) in primary school.**  
Lynda Chinaka, University of Roehampton

So, what do we really know about the teaching of computing in primary schools in England? This poster sets out a research plan and methodology with a particular focus that investigates ways to determine what teaching methods work at primary level.

4. **Writing simple algorithms: making the implicit explicit for novice programmers.**  
Rosina Crisa, Lady Eleanor Holles School

This poster describes a teacher-led action research project in a mixed co-educational secondary academy in South West London. The INPUT-PROCESS-OUTPUT Timeline Tool was trialled with a group of five Y11 GCSE Computer Science students who had been identified as lacking confidence with basic programming and simple algorithm design.

5. **Torino: Motivation in Programming for Visually Impaired Learners**  
Alex Hadwen-Bennett, King's College London

Programming is challenging to learn and for visually impaired (VI) learners there are numerous additional barriers that need to be overcome. Physical programming languages (PPLs), which use physical blocks or pods to represent instructions, have the potential to facilitate programming education for VI learners. This poster focuses on a study which evaluates the potential of the Torino PPL to positively influence the motivation of VI learners towards programming and computer science.
6. **The Ada approach to project based learning with industry partners**  
Alexandra Hemming, Tina Götschi, Ian Dickerson and Steve Rich,  
Ada National College of Digital Skills  

Inspired by the ground-breaking work at High Tech High, each term Ada runs an industry project with our students across three days. This poster outlines the Ada Approach to PBL which is a framework for the effective development and running of industry-led projects.

7. **Challenges in introducing AI programming to children**  
Ken Kahn, University of Oxford  

As part of the EU-funded eCraft2Learn project we have been adding new blocks to Snap! (a more powerful variant of Scratch) for speech synthesis, speech recognition, image recognition, and machine learning. This poster describes the challenges to providing beginner programmers a means to create programs that incorporate AI.

8. **The Roehampton Computing Education Report 2017**  
Peter Kemp, University of Roehampton  

This poster will explore the uptake of computing qualifications in English schools at KS4 and KS5. Using data from the DfE’s national pupil database we can study the results of all students who have sat a qualification at KS4 and KS5 since the implementation of the new computing curriculum in 2014. We argue that the change in curriculum will likely impact females and minority groups disproportionately, and that there is a need for new qualifications at GCSE and A-level covering a broader interpretation of the computing curriculum.

9. **Investigating Severity of Programming Errors**  
Davin McCall, King's College London  

We show results of a manual categorisation of errors in programs written by novices, and rank errors by frequency. We also show a measurement of error severity which includes difficulty of resolution combined with frequency. It can be seen that a manual categorisation which assesses severity rather than frequency gives a substantially different and potentially more accurate view of the types of error (and associated program constructs) that cause the most problems for novice programmers, compared to a ranking based on compiler diagnostics (error messages) using frequency alone.

10. **Mental models to support competence in computer programming**  
Richard Millwood, Trinity College Dublin  

Mental models are the mind's 'mechanisms' for explaining and predicting phenomena and are private to our minds: interconnected, fluid, faulty and ultimately unknowable. The notion of mental model is applied here to the design thinking needed for teaching of programming. Five are proposed: 'problem comprehension', 'programming language', 'notional machine', 'microworld/domain' and 'interactive development environment'.
11. **Collaborative Problem Solving through Worked Examples in Code Club**  
Oliver Quinlan and Lucia Florianova, Raspberry Pi Foundation

Collaborative problem solving is believed to be an essential skill to develop in employment/in order to achieve more successful learning. We studied whether it can be encouraged in the context of Code Clubs by using worked examples, and whether there are other benefits worked example could have on learning programming for children.

12. **What is the role of creative computing education in developing a ‘voice for change’ with secondary school, low-income students in Antigua and St Lucia in the context of climate science and disaster preparedness?**  
Genevieve Smith-Nunes, Ready Salted Code

Technology is rapidly reshaping the world we live in, and the nature of the global workforce. In the US for example, the tech sector employs 4% of the total workforce, with workers earning more than double the national average, and generates 8% of the national economy. But innate tech talent, amazing startup ideas, and natural innovators are not unique to Silicon Valley. The Caribbean region is facing some unique challenges from climate change that tech and innovation may play a key role in managing; our island states need to be ‘ahead of the curve’ in green tech to thrive into the future. The aim of this entry level certificate is to provide young people across the Caribbean with the access to tech and innovation skills and to ‘future proof’ their emerging careers and develop the much needed solutions to cope with, and redress, climate change, green technologies and awareness of digital civics.

13. **Efficacy of computing learning opportunities on attainment: A case study**  
Allen Tsui, Willow Brook Primary School Academy

This case study draws on the classroom based experiences of Willow Brook Primary School Academy in Leyton, East London. The practitioner-led research element considers the efficacy of the teaching resources deployed and attempts to evaluate their impact on attainment elsewhere in the curriculum specifically on learners as they end Key Stage 2 (aged 10-11) and prepare to transfer to secondary schools. The research is part of a dissertation for the MA in Educational Leadership at Manchester Metropolitan University which is expected to be presented in Summer 2019 so is at a very embryonic stage. It has however attracted interest from University College London (UCL) Institute of Education Knowledge Lab and the ASPIRES2 project studying young people’s science and career aspirations.

14. **Integrating computational thinking and digital storytelling: Design-based studies in computing education**  
Robert Whyte, University of Nottingham

This poster reports the first iterative design of an intervention which seeks to integrate computational thinking (CT) and literacy activities in the context of an afterschool program for primary school children. The synergies between literacy, through the composition of digital multimodal projects, are therefore explored. By drawing on constructionist theories of learning, as well as research into multimodal practices in primary school, it tests how the ‘hybridization’ of ‘new’ and ‘old’ literacy practices can enhance learning in both.