(Hello World)

THE MAGAZINE FOR COMPUTING & DIGITAL MAKING EDUCATORS

IN DEFENCE OF PSEUDOCODE
Evidence for its use in education

LEARNING GRAPHS
Maps that show progression

INCLUSIVE BY DESIGN
A shift in thinking to meet all your learners' needs

MACHINE
WHY STUDENTS SHOULD MASTER *THE INNER WORKINGS OF AI*
LEARNING

FROM TURING TO ALPHAGO
Inspire your students with the history of AI

BUTTERFLY OR BIRD?
Classification algorithms demystified

NATURAL LANGUAGE PROCESSING
Block-based coding activities to explore the tech

PLUS INSIDE AN OFSTED DEEP DIVE • **EVERYDAY ASSESSMENT APPROACHES** • PROJECT-BASED LEARNING **PICTURE BOOKS IN COMPUTING EDUCATION** • HIGHLIGHTS FROM WiPSCE • **COMPUTING POETRY**

The Gender Balance in Computing programme

**Are you a Computing teacher based in England?
There's an important new programme that your school can participate in...**

The Department for Education recently funded our Gender Balance in Computing (GBIC) programme, which gives us the amazing opportunity to investigate different approaches to engaging girls in computing in primary and secondary schools. Our research will also be used to find out how we can increase the number of female pupils who select Computer Science at GCSE and A level.

We want to create the widest possible community of schools to help us try different approaches and learn from what we find out. That's why **we are inviting all the primary and secondary schools in England to take part**.

We'd love your school to participate. Join our Schools Network today and help us to find out what **really** makes a difference to getting more girls into computing.

Find out more: rpf.io/gbic-info

National Centre for Computing Education

(HW)

HELLO, WORLD!

Machine learning and AI have an almost unfathomable reach into our lives. From everyday applications such as scrolling through social media or using GPS, to the large-scale collection of citizens' data for corporate or political aims, this technology affects us all. And if the goal of teaching computing is to put the power of technology into young people's hands, where they can understand and control it, rather than be passive users of it, then surely we must teach machine learning.

This is challenging, though, due to the sheer complexity of many of machine learning's underlying algorithms. Even David Spiegelhalter, a mathematician whose work in the 80s underpins a lot of machine learning today, admitted to me he would struggle following by hand every step of the calculations in many of them. Despite this, many teachers and organisations are not shying away from

teaching the inner workings of this technology. It's great to share some of their brilliant work in this issue of *Hello World*.

Last issue, we had a cover theme of inclusion and diversity. Reflecting on the emails we received and feedback in our survey, I'm really happy to say that Catherine Elliott will be continuing the discussion around this in her new column.

On page 95 we've included some of the feedback from our 2019 reader survey, and how we are responding to it. If you have feedback for us, or an idea for a feature, opinion piece, or activity that we could include in the magazine, please do get in touch.

Sian Williams Page
Editor

FEATURED THIS ISSUE

KEN KAHN

Ken is a researcher in the Department of Education at the University of Oxford. He was a member of Seymour Papert's Logo Group at the MIT AI Lab.

NAZIA FAKHRUDDIN

Nazia is a Royal Institution Computer Science Masterclass Speaker and Code Club Volunteer for at a library in the London Borough of Hounslow.

URSULA MARTIN

Ursula Martin has been an educator for 17 years. She is a District Level Technology Resource Teacher in Mobile, Alabama.

Hello World is a joint collaboration:

Raspberry Pi

(HW) CONTENTS

COVER FEATURE

15

THE HISTORY OF AI
Fascinating stories to share with your students

19

POPBOTS
An innovative project run by MIT

20

NATURAL LANGUAGE PROCESSING
Explore the tech behind Alexa with Snap!

NEWS, FEATURES, AND OPINION

CONVERSATION

REVIEWS

LEARNING

RESOURCES & LESSON PLANS

CORONAVIRUS OUTBREAK SEES A SHIFT TO ONLINE EDUCATION

The widespread implementation of online learning is taking place at an unprecedented rate, with impacts on student attainment and teacher well-being

Sian Williams Page

A s the number of Covid-19 cases continues to rise globally, around 372 million children are currently experiencing an upheaval in their education due to school closures. "The scale and speed of the current educational disruption is unparalleled and, if prolonged, could threaten the right to education," said UNESCO Director-General Audrey Azoulay. Schools are turning to distance learning programmes in an effort to continue providing an education for their students.

In the UK, most schools are open as of 17 March, but teachers are taking measures to prepare for widespread closures. James Bowen, director of policy at the NAHT school leaders' union, told Tes that his organisation is working with the DfE to "ensure they consider all scenarios and possible eventualities... We are urging them to continually review and update their advice to schools. This includes scenario planning in the event of regional or national school closures."

Which platform, and how to use it?

In many schools, the task of choosing and managing the correct online platform to use in the event of a closure falls on the shoulders of the Computing teachers. In the absence of more detailed current guidelines, teachers are facing challenging decisions around which technology to use and how to go about implementing it. Many ed-tech companies are now offering their products for free. UNESCO has published a list of nearly 30 free programmes that they say have "wide reach, a strong user base and evidence of impact".

School closures have brought extra challenges to teachers who are parents themselves. Stan Covington, a teacher in Bangkok, has described the challenges of implementing various online systems while juggling childcare. "My wife and I, both teachers, are struggling with balancing the demands of virtual schooling and the demands of managing our young

children's virtual learning. It's a tough balance that, on the first day of virtual learning, has already fried our nerves. Between Seesaw, the emails I've received from our kids' various and wonderful teachers about Seesaw links, and my own Flipgrids, email, Google classrooms, Google chats, and PowerSchool learning sites, I am already overwhelmed."

There are concerns that moving to online learning could increase the attainment gap between richer and poorer students. In the UK, seven per cent of homes do not have access to any kind of internet connection, and many students do not have access to a laptop or tablet. Many children do not have a suitable place to work at home.

There are also serious concerns about students going hungry due to school closures. In the US, 20 million meals are distributed for free in schools each day. The most recent data available suggests around 14 per cent of pupils in England claim free school meals. **(HW)**

LANGUAGE SKILL A BETTER PREDICTOR OF CODING ABILITY THAN MATHS

New research highlights the links between learning a foreign language and learning to code

New research has shown that aptitude for learning foreign languages is a stronger predictor of learning to program than basic maths knowledge. The study, led by Chantel Prat at the University of Washington and published in *Scientific Reports*, examined various aspects of adults' cognitive abilities, including language and maths skills, while they learned to code in Python for the first time.

The researchers tracked how quickly participants proceeded through programming tutorials, how accurate their code was, and how well they understood the programming language. Language aptitude was found to explain variance in all three measures of skill, once other areas of cognitive ability were accounted for. Numeracy was found to explain variance in learning rate only.

In many schools, a certain level of maths attainment is often a prerequisite for studying computer science. While Prat's study was carried out on adults, its results raise questions around this widespread practice. "I don't have a good reason to think the predictive utility of language, numeracy, general cognitive abilities or neural metrics would change in young learners," she says.

"Learning programming languages in and of itself doesn't require knowledge of linear algebra," she explains. But maths remains an important skill in many applications of programming. "I think the thing to keep in mind is that programming is still primarily taught in computer science and engineering departments. I don't have the expertise

or the data to suggest that complex math isn't required for computer science. In fact, I strongly suspect that there are many aspects of engineering that do rely on math!"

The researchers chose to carry out the study in Python because of its close resemblance to English structures such as paragraph indentation and its use of real words rather than symbols for functions. Prat believes, however, that her results will carry over to more complex languages, such as Java. "I believe we'll find that the programming languages that are anecdotally perceived as being intuitive or easy to learn or read may be those that most resemble human language structures, and learning those languages may be most strongly predicted by language aptitude. However, all programming languages emulate human communication structures and so I do think the relationship between language aptitude and programming aptitude will show up with different languages and in different learning environments. It's a question we want to pursue empirically." **(HW)**
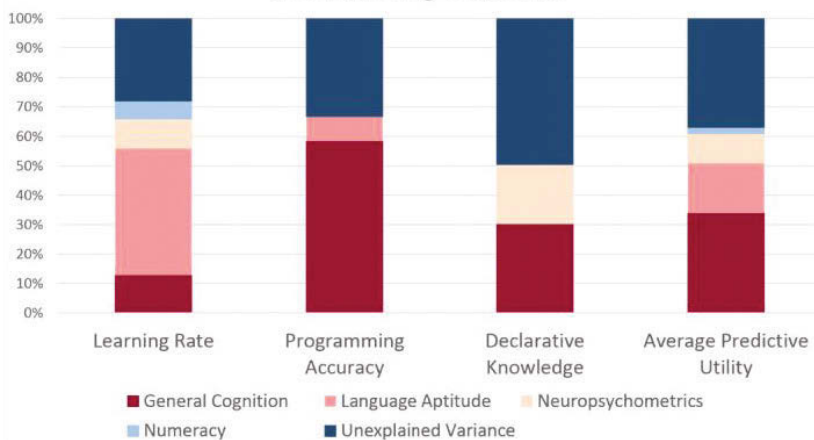

■ Participants learnt Python while wearing a specialised headset that measures electrical activity in the brain. Image credit: Justin Abernethy/U. of Washington

■ Cognition and language aptitude are greater predictors of learning than numeracy. Image credit: Prat et al./ *Scientific Reports*

YOUNG CREATORS WOW JUDGES AT COOLEST PROJECTS USA

150 attendees showcased their inventions in Santa Ana, California

Kevin Johnson

Coolest Projects USA returned on Saturday 7 March for its third celebration of the USA's creative and innovative young creators. Held at Discovery Cube Orange County, a science centre in Santa Ana, California, the technology fair allowed young innovators to showcase their work. It saw young people attend from Northern California, Texas, Lousianna, and Georgia.

Anyone under the age of 18 could enter their creations into one of the fair's categories of Hardware, Web, Mobile Apps, Visual Programming and Games. Last year's winner of the Mobile Apps award, Ashley Chu, featured as a special guest speaker. She gave advice for anyone feeling nervous, while using what she called the 'power pose' accompanied by a wiggle dance to shake any nerves away.

The judges were faced with difficult decisions in selecting the winners of each category. Kausthubh, Maria, Sarah and Eric won the Hardware award with Vest Buddy, 2nd Generation — a wearable technology project that helps non-verbal young people communicate. The group also attended Coolest Projects last year and impressed the judges by showcasing how they incorporated feedback into their project. The winner of the Visual Programming category was Nathan, who devised Memory Assistant, a system to remind older people to stay hydrated and take medication when needed.

The Girls Who Code Club from Maywood, California, showcased several projects, inspired by things that are important to them. Kayla, Carola, Cassandra, Natalia and Sibone from the club won the Games award for Deforestation Station, a game designed to help people better understand deforestation, as well as provide solutions.

The Overall Winner of Coolest Projects USA 2020 was awarded to Adarsh, for his project Vital Signs. Adarsh got involved with digital making when he started attending a local makerspace with his big brother. He created a system for monitoring vital signs to diagnose illness without touching the patient. It also provides limitless possibilities from preventing Sudden Infant Death Syndrome to diagnosing viruses. **(HW)**

> ## " ANYONE UNDER 18 COULD ENTER THEIR CREATIONS

■ Members of the Girls Who Code club from Maywood, California, won the Games award.

■ Vest Buddy is a wearable technology project that helps non-verbal young people communicate.
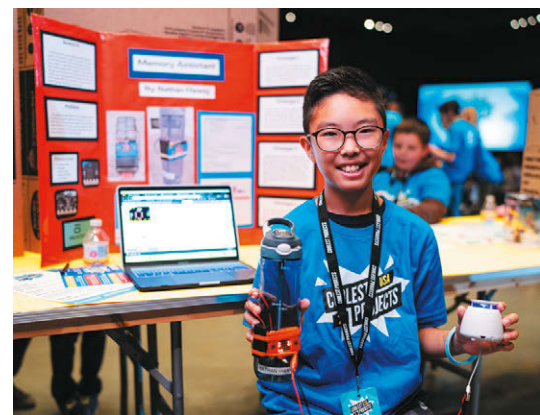
■ Young creator Nathan won the award for Visual Programming for his project Memory Assistant, which helps older people stay hydrated and gives reminders to take medication.

■ The Megaprocessor is a major exhibit at the Centre for Computing History
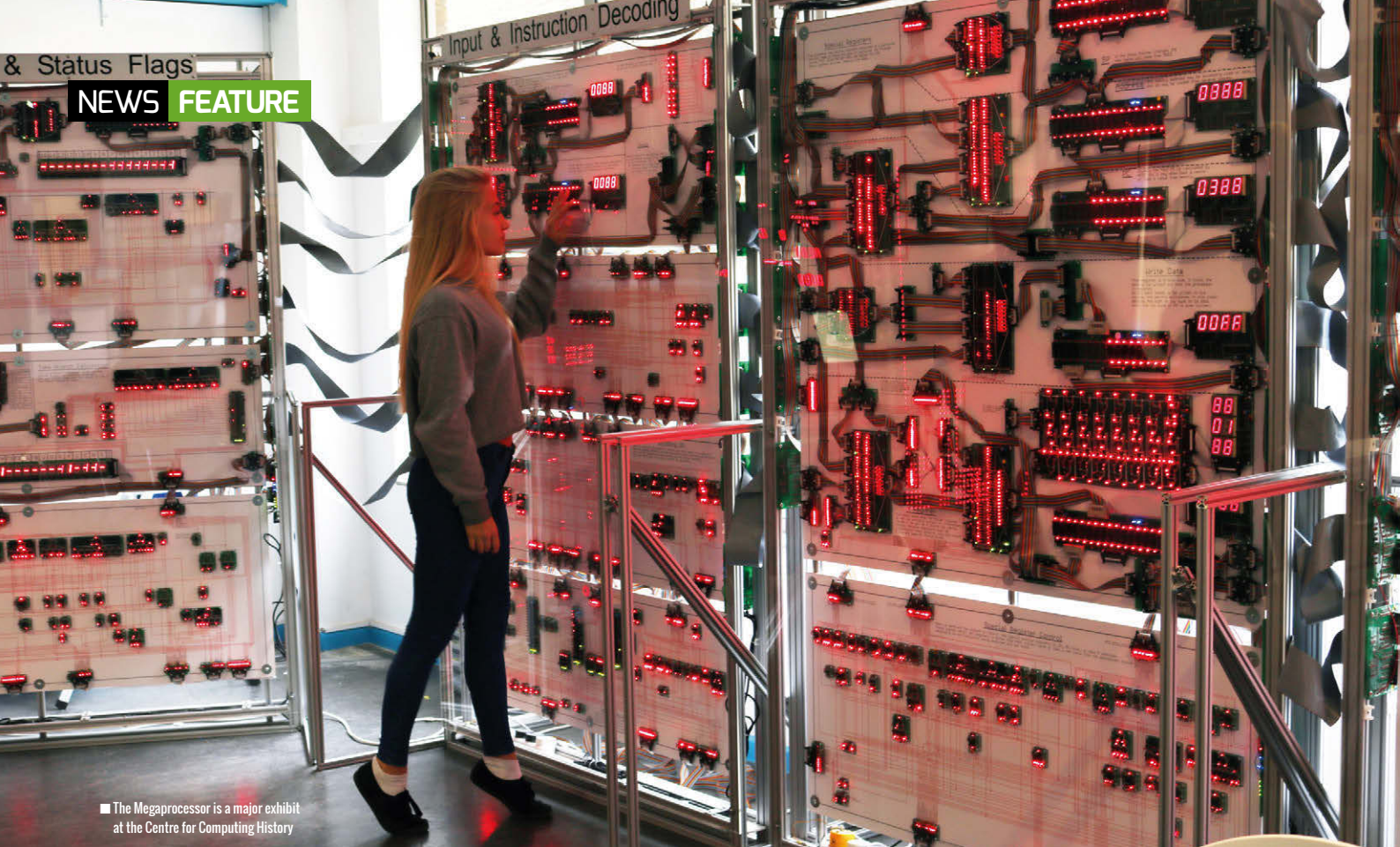
MUSEUM SEEKS NEW TRUSTEE

The Centre for Computing History in Cambridge, England, has
an opportunity for educators to guide its management

Liz Upton

The Centre for Computing History is seeking at least one new trustee for its Board. They welcome applications from people with a wide range of skills and experience, and are particularly seeking a trustee with a background in education.

The Centre for Computing History is a young museum based in Cambridge, England, with a growing visitor base and an established and successful learning programme. It offers a hands-on learning experience wrapped up in a historical context. Visitors can touch, program, and game on original hardware and learn about the people who created the technology should be we rely on today. The Centre welcomes visits from all schools, FE Colleges, and universities, and any organisations supporting the education of young people.

The role of a trustee

As the centre is a registered charity, trustees have a series of statutory responsibilities and also act as ambassadors for the museum. They steer the organisation in pursuit of its charitable aims. Trustees are expected to use their specific skills, knowledge, and experience to help the board of trustees to reach sound decisions for the benefit of all stakeholders. The role is not remunerated, but reasonable expenses can be met. Trustees are appointed for a term of four years, extendable for up to three terms.

Board members will need to attend trustee board meetings, normally held six times per year in Cambridge, and occasionally promote the museum at events and meetings with key people. They will generate and process key documents and papers. New trustees

■ There's lots to discover for all ages at the museum

are required to attend an induction process at the museum.

Further details on the role and how to apply are available at **computinghistory.org.uk/news** (HW)

■ Clare Parry-Jones, UK Enterprise Sector Director at Computacenter, presents Aoibheann and Avye with their awards

GIRLS RECOGNISED FOR INSPIRING LEADERSHIP

Avye Couloute and Aoibheann Mangan were recognised for their work in shaping the future of the technology industry at the FDM everywoman awards

Two young leaders have been recognised for their efforts in inspiring more girls into careers in tech. Avye Couloute and Aoibheann Mangan were announced joint winners of the FDM everywoman 'One to Watch' award at the 2020 FDM everywoman in Technology Awards earlier this month.

Avye Couloute, 12, attends Surbiton High School in London. She creates opportunities for girls, in particular, to explore coding and robotics, through her Girls Into Coding events. She has fundraised and secured support to provide girls with tech-themed books and physical computing kits. Avye is ambitious about the scope of her work: "It's a big goal but this year I want to reach one thousand girls through my events." Girls Into Coding is currently sponsored by Microsoft but Avye and her mum, Helene, are currently looking for new partners to expand their work.

Avye started to become interested in computing at the age of seven, after attending a workshop with her mum. "I really enjoy going to tech events as

there are loads of people who want to share their knowledge and ideas with you. They also allowed me to craft, code, design and to discover new things," she writes on her blog, **10tonolimit.com**.

Avye's work was also recognised last year when she won The Diana Memorial Award. The Diana Award is a charity founded in memory of Princess Diana and her belief that young people have the power to change the world.

Aoibheann, 13, is a student at Mount St. Michael Secondary School in Claremorris, Ireland. She runs a CoderDojo at her school, where she mentors over 50 children weekly, with 60 per cent of them being girls.

"I began coding when I was 8," Aoibheann explains. "I fell in love with it from day one. Mum tried to find a CoderDojo for us to attend but there was none in our area at the time so she set one up. That was something that inspired me when I went into first year in my secondary school, to try and set up a lunchtime coding club."

"When I first started a local Dojo, I can remember Mum meeting one of her friends

and asking whether her daughter was going to come along. Her friend said 'Ah, no, she thinks it's for boys...' A few weeks later and she was first in the door to come along. She had heard from her friends that it was a girl doing the workshops and she felt inspired to come! That was lovely to hear."

Aoibheann has also been at the forefront of a campaign for broadband equality in Ireland. "In Ireland there is a large divide in areas where you can have broadband and areas where you have dial-up or no internet at all," she says. "I live in rural Ireland where this problem is prevalent. I spent many hours sitting in my local car park in Tesco to get WiFi to complete projects. I found this very unfair and I'm mindful in times like this with the coronavirus worries that a lot of schools may have to use remote classroom learning in the coming weeks, that students may be faced with being unable to complete schoolwork due to lack of broadband. It's important everyone has a fair chance. I'd love to be able to remote work in this beautiful part of Ireland some day and not be faced with moving away to work!" (HW)
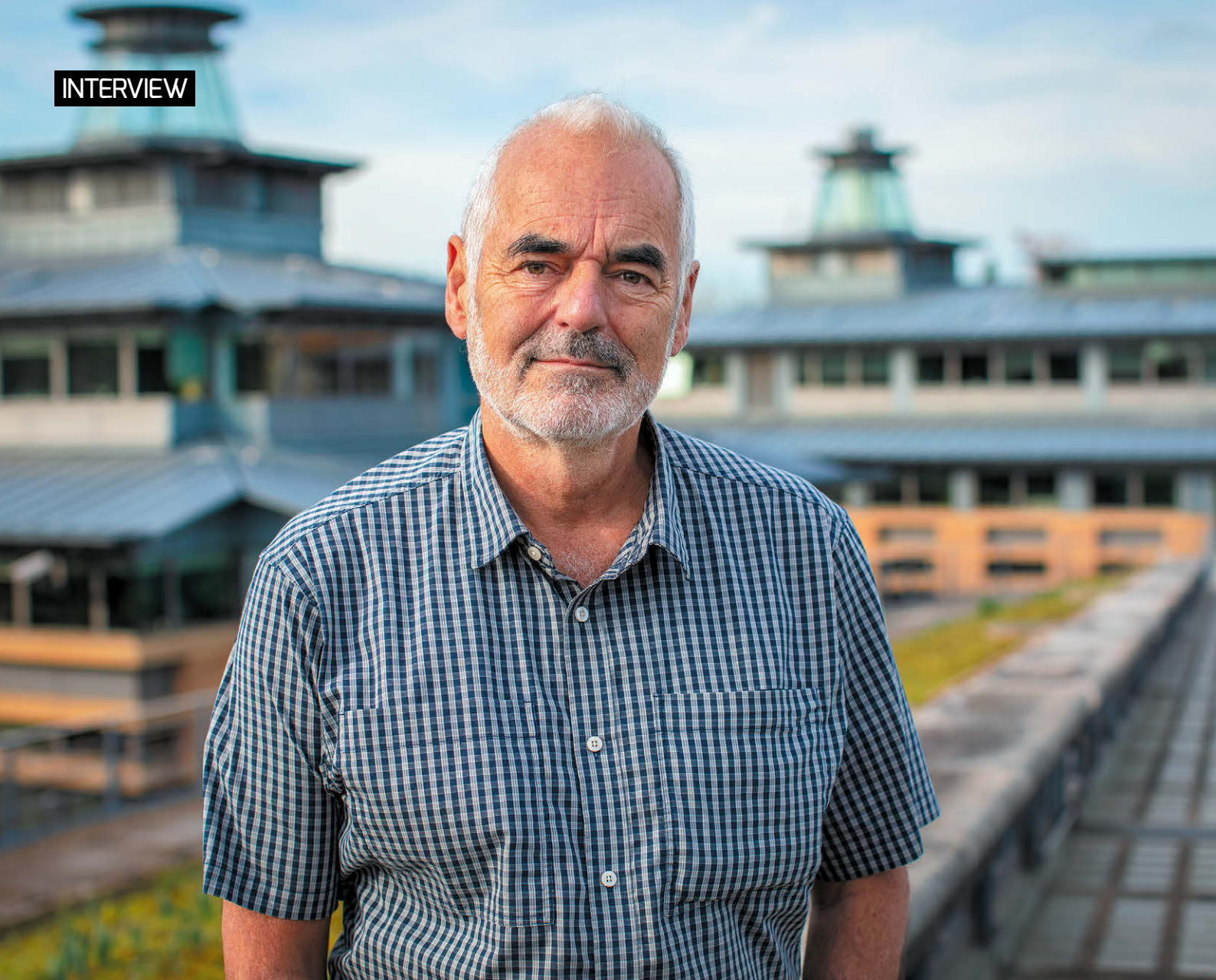
DAVID SPIEGELHALTER

ON TEMPERING THE HYPE AROUND AI

· ·

The statistician meets **Sian Williams Page** to discuss shoddy statistics, overblown algorithms, and a golden age for artificial intelligence

For someone who is arguably the most famous living statistician, David Spiegelhalter is surprisingly scathing of his field. "When I complain about statistics, I just think of old-fashioned education, where it appears like, 'Is it a chi-squared? Is it a t-test? Is it one-sample or two-sample? Is it a regression? Which test do I slap on and get some answer coming out of the program?' And that's dire, it really is."

Spiegelhalter is letting rip at his field while sitting in the central court of the Centre for Mathematical Science, where he is the Chair of the Winton Centre for Risk and Evidence Communication. The Centre's buildings are beautiful, with sweeping curved walls and low, grass-covered roofs. "We hope the layout of the buildings and the way they work will actively encourage fresh dialogue and discovery," the award-winning architects explained to The Guardian in 2003. Though perhaps architecture can only go so far: just inside the entrance, there is a sign urging staff and students to smile at one another, and to try saying hello.

Spiegelhalter doesn't seem to need a reminder to smile and talk. Over the course of his career he has shifted from working on serious maths to improving how evidence is communicated to the public. In his latest book, *The Art of Statistics*, he applies what he calls 'simple numeracy' to messy, real-world problems. The questions he tackles include 'Do speed cameras reduce accidents?', 'What's the cancer risk from eating bacon sandwiches?', and 'Does going to university increase the risk of getting a brain tumour?' (The answer to the final question is a resounding no.)

He hopes the book provides a grounding in statistical principles for new graduates being lured into careers in data science — a multidisciplinary field that combines statistics with machine learning, design, and communication. He welcomes the advent of the subject and the facelift it has given statistics. "I think the ability to deal with data critically and to realise its strengths and limitations is the most important skill in the future world and it's an extremely marketable skill as well," he explains.

But in the shift from statistics to data science, has one bag of tools simply been replaced by another? Are machine learning algorithms being applied with little care for their inner workings, much like the statistical tests of Spiegelhalter's early years?

"Machine learning is even worse! Because you can get a set of data, and then you bang in a logistic regression, and classification trees, random forests, neural networks, small vector machines and off you go, bam, bam, bam. You apply them all and you've got no idea what they are doing and out they come with some error rate and you think 'Oh, that one's best.' And this is appalling, absolutely dreadful, because you've got no idea," he says.

"The robustness and explainability and fairness and transparency are absolutely vital when it comes to this work. Frankly, unless you've got a pretty good idea of what it's doing you've got no guarantee whatsoever that if you move into a slightly different domain that it won't just all fall apart. You slightly change the inputs, move into a different context, and you've got no idea why, suddenly, it doesn't work."

Spiegelhalter's is not just sceptical of how businesses apply algorithms, but also of the ambitious and broad claims made about the technology. "You have to recognise there are certain problems and challenges that machine learning and AI have been absolutely fantastic at: well-defined problems in terms of optimisation. I use Google Maps — it's a fantastic optimisation algorithm, there's fantastic image recognition, there's an amazing ability to play games. But these are deeply restricted classes of problems: you need no background knowledge, you just need a huge amount of information on past cases. The idea that 'Oh, because a program can learn Go [in 2016, Google DeepMind's AI program AlphaGo beat world master Lee Sedol at the board game], therefore you can let it free on medical records and it can tell you what's going to happen to a patient,' is complete nonsense, just nonsense! And there's so much overhype and over-claiming being made."

If anyone is qualified to raise an eyebrow at the current enthusiasm for machine learning and artificial intelligence, it's Spiegelhalter. Some 35 years ago, he and a colleague applied probability theory to artificial intelligence with significant consequences: the work has been cited thousands of times by other researchers and underpins many machine learning algorithms used today. He brightens when recalling the period. "When you look back on your career, you think there are certain times which have been a bit of a slog, and other times it was really exciting and energising. The eighties were really exciting."

The challenge educators face to convey a balance of enthusiasm for machine learning and artificial intelligence with a healthy dose of scepticism to their students is one Spiegelhalter is keenly interested in. Having previously advised England's Department of Education on the syllabus for GCSE Mathematics, he is now a member of a Royal Society panel that provides recommendations on data science in schools. In 2018, he co-authored a report that argued for a 'more detailed, updated, and coherent computing curriculum,' with recommendations for introducing content on machine learning and computational modelling and strengthening content on data representation.

"Data science certainly does not belong in maths. I don't even really believe statistics belongs in maths. Data science in all its manifestations — sorting data, getting conclusions from data visualisation, communication — is an essential skill in the world and it's one that should be part of schools' education, but I don't know where it should be. It has a broad spectrum of literacy that can be applied across all areas — geography, the humanities, computer science — nobody owns it, and it's wonderful. I wish I could say 'You must do it this way,' but I really don't know. But, I suppose it's quite right you should go into something not knowing the answers." (HW)

"DATA SCIENCE IS AN ESSENTIAL SKILL IN TODAY'S WORLD

Jonathan Howard/Raspberry Pi Foundation

THE INCLUSIVE COMPUTING CLASSROOM

Catherine Elliott launches her new column with a reflection on belonging

I **am delighted to bring you a regular column on all things related to special educational needs and disabilities (SEND) and Computing.** Over the coming issues I will go into more depth about the strategies and resources that teachers can use to ensure their SEND learners are accessing the subject in a meaningful way.

Why is this important? Many of our young people with SEND use technology to access information and entertainment, just like their peers. Some of them will rely on computers to communicate and to access their learning in school. We need to ensure that they are taught how to do this safely, responsibly, and effectively. Our students with SEND also need to have the same economic opportunities presented by studying the subject. Furthermore, we need them to be represented in our future workforce – the digital devices and products of the future will only be as diverse as the people who design and test them.

I have recently been reading a great deal around inclusive design, and this is one strategy for supporting students with SEND. You can read more about this at microsoft.com/design/inclusive/ and mismatch.design. "Inclusive design doesn't mean you're designing one thing for all people. You're designing a diversity of ways to participate so that everyone has a sense of belonging." Susan Goltsman, (quoted in *Mismatch* by Kat Holmes, 2018). This sense of belonging is at the heart of inclusion. If, as educators, we can provide multiple ways of engaging with the learning in the classroom, then we can ensure more of our young people are included in lessons. (Obviously some students with severe learning difficulties or physical disabilities will still require specific interventions.)

Consider how you measure success in your classroom. Some students may struggle to create a working program in Python, Scratch, or on the Bee-Bot. This is where the PRIMM model is really useful. If you haven't heard of PRIMM yet, then head to suesentance.net/primm-project to find out more. Essentially it provides multiple access points to interact with code. A student can read existing code and discuss what they can identify and predict what they think it will do. Another student can take a working program, and make small adaptations to it, for example change the background, alter the values in a variable, change the contents of a string. A young person with SEND can thus learn about the same computer science concepts as their peers, without the fear of failure, or the demand on working memory and recall that writing a program from first principles involves.

We could also offer a range of other programming-related activities during lessons, for example using unplugged activities to teach key concepts, spending time designing programs in groups, or working with physical computing devices that have a multimodal output (light/sound/movement). More on these in a future column!

So take time to reflect on what happens in your own classroom. How diverse are the methods of interaction in your lessons? Are all students included in the learning? (HW)

Catherine is the SEND lead for the Sheffield eLearning Service (sheffieldclc.net), and she has spent the last five years working on ways to make computing accessible for all learners. She is a member of the CAS Include working group, and co-leader of the Sheffield and South Yorkshire Secondary CAS Community.
@catherinelliott

SNAPSHOTS FROM THE HISTORY OF AI

George Boukeas introduces fascinating
stories to share with your students

The story of artificial intelligence (AI) is a story about humans trying to understand what makes them human. Some of the episodes in this story are fascinating. These could help learners catch a glimpse of what this field is about and, with luck, compel them to investigate further.

The imitation game

In 1950, Alan Turing published a philosophical essay titled *Computing Machinery and Intelligence*, which started with the words: "I propose to consider the question: Can machines think?" Yet Turing did not attempt to define what it means to think. Instead, he suggested a game as a proxy for answering the question: the imitation game.

In modern terms, you can imagine a human interrogator chatting online with another human and a machine. If the interrogator does not successfully determine which of the other two is the human and which is the machine, then the question has been answered: this is a machine that can think.

This imitation game is now a, fiercely debated, benchmark of artificial intelligence called the Turing test. Notice the shift in focus that Turing suggests: thinking is to be identified in terms of external behaviour, not in terms of any internal processes. Humans are still the yardstick for intelligence, but there is no requirement that a machine should think the way humans do, as long as it behaves in a way that suggests some sort of thinking to humans.

In his essay, Turing also discusses learning machines. Instead of building highly complex programs that would prescribe every aspect of a machine's behaviour, we could build simpler programs that would prescribe mechanisms for learning, and then train the machine to learn the desired behaviour. Turing's text provides an excellent metaphor that could be used in class to describe the essence of machine learning: "Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain. We have thus divided our problem into two parts: the child-programme and the education process."

It is remarkable how Turing even describes approaches that have since been evolved into established machine learning methods: evolution (genetic algorithms), punishments and rewards (reinforcement learning), randomness (Monte Carlo tree search). He even forecasts the main issue with some forms of machine learning: opacity. "An important feature of a learning machine is that its teacher will often be very largely ignorant of quite what is going on inside, although he may still be able to some extent to predict his pupil's behaviour."

The evolution of a definition

The term 'artificial intelligence' was coined in 1956, at an event called the Dartmouth workshop. It was a gathering of the field's founders, researchers who would later have a huge impact, including John McCarthy, Claude Shannon, Marvin Minsky, Herbert Simon, Allen Newell, Arthur Samuel, Ray Solomonoff, and W.S. McCulloch.

algorithm to be implemented into a program and Turing famously played a few games with human opponents executing the algorithm by hand. Claude Shannon wrote *Programming a Computer for Playing Chess*, in 1950, where he laid the foundation for many of the search techniques that would later be applied in games.

The main approach for playing many of these games is search: systematically

a solution "will act as a wedge in attacking other problems of a similar nature and of greater significance". Indeed, search was the driving force behind many of the landmark achievements in the field: making plans and schedules, proving theorems with logic, solving algebraic problems, making inductions, and so on.

> ## INTELLIGENCE IS THE QUALITY THAT ENABLES AN ENTITY TO FUNCTION APPROPRIATELY AND WITH FORESIGHT

The simple and ambitious definition for artificial intelligence, included in the proposal for the workshop, is illuminating: 'making a machine behave in ways that would be called intelligent if a human were so behaving'. These pioneers were making the assumption that 'every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it'. This assumption turned out to be patently false and led to unrealistic expectations and forecasts. Fifty years later, McCarthy himself stated that 'it was harder than we thought'.

Modern definitions of intelligence are of distinctly different flavour than the original one: 'Intelligence is the quality that enables an entity to function appropriately and with foresight in its environment' (Nilsson). Some even speak of rationality, rather than intelligence: 'doing the right thing, given what it knows' (Russell and Norvig).

Playing games: search

A lot of research in artificial intelligence focused on games. Over the years, programs for checkers, backgammon, and many other games have reached competence levels that surpassed the best human players.

However, chess was the most prominent game, right from the start. Alan Turing and David Champernowne developed a basic algorithm called Turochamp for playing chess back in 1948. It took years for that

generating and evaluating positions and moves. That may sound trivial for a computer but the number of combinations in non-trivial problems quickly explodes exponentially and a brute-force enumeration of all the possible outcomes is impossible. Shannon estimated the number of different chess games to 10^{120}. In these vast search spaces, a lot of thought needs to go into evaluating search states to guide the search effort and prune non-promising search paths.

In his paper, Shannon discussed the value of research in games, explaining that

Deep Blue

In 1996, the chess world champion Garry Kasparov played against Deep Blue, a purpose-built IBM computer. Deep Blue became the first chess machine to ever win a game and, a year later, a match against a world champion under regular time controls. Kasparov's defeat made the headlines and is considered a milestone in the history of AI.

The main driving force behind Deep Blue was search: it was able to generate and evaluate 200 million positions per second. The evaluation function was handcrafted by human experts and the only form of learning was the system's ability to fine-tune some of its parameters.

Modern chess-playing programs need to evaluate far fewer positions and do not require specialised hardware to vastly outperform human players. The last known win by a human against a top chess-playing machine was in 2005.

Deep Blue's victory was part of an impressive string of achievements, and yet there were problems that seemed elementary and yet proved extremely hard to tackle. This was eloquently summarised in Don Knuth's remark: "AI has by now succeeded in doing essentially everything that requires 'thinking' but has failed to do most of what people and animals do 'without thinking' – that, somehow, is much harder!" It is only very recently that artificial intelligence has made breakthroughs in the latter class of problems, such as image and speech recognition, and this is the main reason it has become so prominent.

Watson

In 2011, a computer system built by IBM and named after the company's founder, competed against two human champions in the game of *Jeopardy!* The highly publicised match resulted in an impressive win for Watson, in a context that would traditionally be considered extremely hard for a computer to tackle.

Watson is not really a computer system for playing *Jeopardy!*, though. It is a system that uses multiple different techniques to answer questions posed in natural language. In order to answer a question, Watson generates multiple hypotheses and seeks to support them by drawing evidence from a body of sources. In other words, Watson is able to provide justification for its answers. There are many areas where Watson is now being applied, the most prominent being in assisting doctors with diagnosis and suggested treatment.

Watson's level of complexity is astonishing and it would be impossible to develop such a system without some form of learning. Echoing Turing's comments about learning machines, Grady Booch, who was involved in building Watson, remarked that "building a cognitive system [like Watson] is fundamentally different than building a traditional software-intensive system of the past. We don't program them. We teach them."

Neural networks

A neural network receives input values and computes output values, which are influenced by a set of parameters called the weights. The function computed is a composition of simpler functions – represented by individual neurons. To build a neural network boils down to how these simpler functions are organised and composed (the network 'topology'), also taking the weights into account. A neural network can 'learn' in the sense that its weights can be modified, swaying the output in more desirable directions.

Behind every recent breakthrough in artificial intelligence, you will find a neural network. Teaching a neural network of sufficient complexity requires a significant amount of training instances and computational power. Even though neural ▶

GEORGE BOUKEAS
George is a Learning Manager for secondary education at the Raspberry Pi Foundation (**@pythonies**).

" GO WAS CONSIDERED THE HOLY GRAIL OF GAME AI

Neural networks are behind nearly all AI technology

▶ networks have been around for decades, it has only been in the last few years that their potential has been realised, as the amount of available training data has skyrocketed and computing power, along with dedicated hardware, has become more readily available.

AlphaGo

Go is an ancient strategy game for two players, who take turns placing black and white pieces (stones) on a 19x19 board.

It is a notoriously difficult game for computers. The number of possible board positions is estimated at an astronomical 10^{170}. Traditional search techniques in such a vast space of possibilities is pointless and it has proved very hard to develop functions that reliably evaluate positions in order to guide the search. Researchers estimated that it might take decades for machines to beat humans at Go, which was considered to be the holy grail of game AI.

Enter AlphaGo, a computer program by DeepMind. In 2017, AlphaGo beat Ke Jie, the world's top-ranked player at the time, following victories over other high-ranking professional players.

AlphaGo combines previously known methods in a novel way. It studies human games or uses self-play, in order to learn how to evaluate positions and moves. It uses **neural networks** to compute its evaluation functions and modify them while learning. It searches through the vast space of possible positions by **taking random samples,** instead of searching systematically.

This is such a promising generic approach that AlphaZero, a generalised version of the program, used only self-play to achieve a superhuman level of play in the games of chess, Shogi and Go within 24 hours. This is a step closer to Turing's vision of a blank slate child-programme, endowed with the ability to learn. (HW)
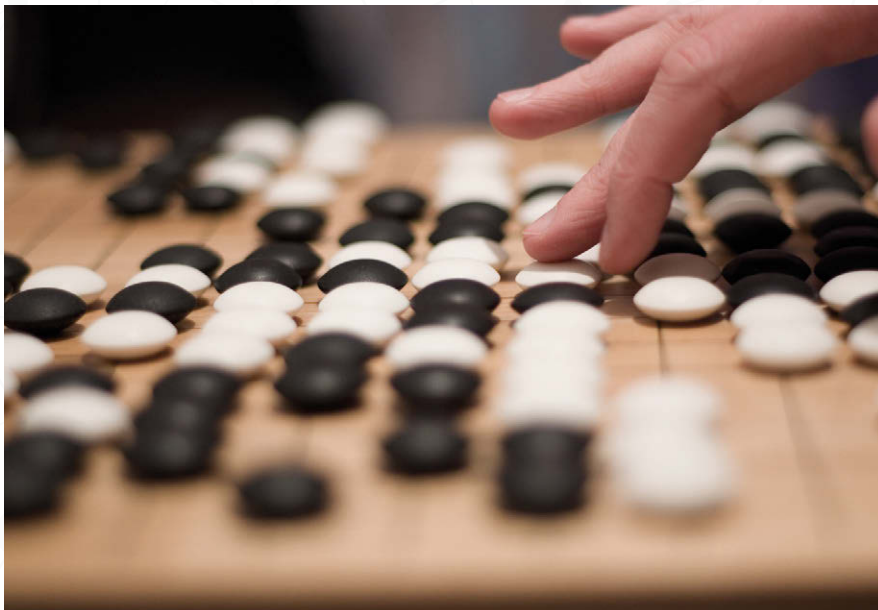
FURTHER READING

Artificial Intelligence,
Michael Wooldridge. Ladybird, 2019.

Machine Learning for Humans,
Vishal Maini and Samer Sabri, 2017.

*The Quest for Artificial Intelligence:
A History of Ideas and Achievements.*
Nils J. Nilsson, Cambridge University Press, 2010.

Machines Who Think,
Pamela McCorduck, A K Peters, 2004.

POPBOTS OPEN AI TO THE YOUNGEST LEARNERS

From Siri to gaming, artificial intelligence is all around us.
Ben Hall explores an innovative project run by MIT that makes
this technology accessible to preschoolers

Very often, the most vivid examples of how quickly children learn and progress can be seen in the youngest age groups. A great example of this is how children use construction-based resources such as wooden blocks, train tracks or LEGO. Often, children experience these for the first time when they enter the reception classroom. Initially, they will use these resources in quite a free, unstructured way. Over time, with guidance and support, they construct increasingly complex and imaginative models, often linked to other areas of learning. The models from the end of the year are unrecognisable from those constructed at the beginning.

Can the same be said of children's first encounters with technology in the classroom? Many schools will provide access to tablets or computers as part of their early years provision, but how much of that is about consuming the technology, and do children relate this technology to real-world applications? In my experience, simply having the devices available is not enough — most children start school able to navigate their way around a tablet, and merely exposing them to it will not introduce new concepts. How can we develop children's computing knowledge so it will encourage them to be more than just passive users?

AI literacy

Could AI provide an answer? A team at MIT has addressed how to make AI accessible by developing PopBots, small robots aimed at introducing AI to young learners. They have also developed easy-to-use resources to complement the hardware.

One such project explores using machine learning, where a robot can be programmed to sort healthy or unhealthy foods. Children begin by classifying foods for the robot, but soon realise that it would take too long to do this for every single food type. Through some supervised machine learning, children can quickly train PopBots to classify foods, also developing their own understanding of healthy eating. Through this simple activity, children become AI-literate creators — turning a passive device into something that makes informed, intelligent decisions.

Randi Williams, a PhD student at MIT, who worked on the PopBots project, provides her view of introducing children to AI: 'Children's views of themselves relative to the technology change. Their views of how much they can participate in technological invention change. I love the fact that early AI education makes children feel more curious about their world and empowered to change it.'

Research-led

PopBots are still in the early stages of development and not yet widely available. Despite this, there is already a growing body of research investigating the inclusion of AI in the curriculum at an early age. Williams would like to see PopBots developed as an open-source platform that students could build from classroom materials. Research supports further development.

A recent paper by Williams and colleagues investigates how AI can influence young children's perceptions of robots **(helloworld. cc/robots)**. They found that perceptions of robots can be shaped at an early age, so for children to be AI-literate, their earliest experiences should be meaningful and informed. As it is an emerging technology, the research is at an early stage. There is no doubt that our world will be increasingly shaped by AI, so helping children develop a conceptual understanding at an early age needs to be at the forefront of curriculum development. **(HW)**

WHAT IS A POPBOT?

PopBots are constructed using LEGO and use a mobile phone with additional LEGO or Arduino peripherals. Users interact with them via a programming interface on a tablet or computer.

BEN HALL
Ben Hall is a Learning Manager at Raspberry Pi, where he develops resources for the NCCE. He is a CAS Master Teacher and a Raspberry Pi Certified Educator (**@hengehall**).

NATURAL LANGUAGE PROCESSING WITH SNAP!

Students can write programs to solve puzzles like 'Hat is to head as glove is to X' with the block-based software

Today's AI programs can generate recommendations, answer questions, summarise text, analyse sentiment, and perform translations. To do this, they rely on a technique that treats words as long lists of numbers. These AI programs are based on neural networks. Neural networks cannot deal directly with textual data – input text must be converted to numbers, and output numbers are often converted back to text. The list or vector of numbers associated with each word can be understood as coordinates. The technique is called 'word embedding' because it places each word somewhere in a high-dimensional space. This is a space that is impossible to visualise and very hard to understand. But putting words into this space enables AI programs to do some very impressive things.

What might school students do with word embeddings? Can they be made accessible to non-experts? What might they learn in the process?

We attempted to answer these questions by building a Snap! library that contains 20,000 word embeddings in 15 languages. Using a block that reports a list of 300 numbers for any of the known words, students can create programs that search for similar words, find words that are the average of other words, explore cultural biases, and solve word analogy problems. These programs can work in a single language or rely upon the alignment of the word embedding spaces of different languages to perform rough translations.

To compute with word embeddings, we need to perform vector arithmetic. This can be accomplished by providing vector arithmetic blocks. More advanced users can instead take advantage of Snap!'s support of higher-order functions to use list mapping blocks to perform the vector operations. Students may have encountered linear algebra in their studies (students in the UK, for example,

KEN KAHN
Ken is a researcher in the Department of Education at the University of Oxford. In the 1970s he was a member of Seymour Papert's Logo Group at the MIT AI Lab.

are expected to learn to apply addition and subtraction of vectors and multiply vectors by a scalar between the ages of 11 and 14). They may be surprised to discover how useful it is in dealing intelligently with words, too.

Word embedding Snap! blocks

A surprisingly wide range of applications is possible with only two blocks: a block that, given a word, reports a list of numbers; and a block that, when presented with a list of numbers, reports which words are closest. We also provide blocks reporting all known words, and blocks for helping to visualise the locations of words. Some functionality

reached from **helloworld.cc/craft**. The AI blocks are described in an article in Hello World issue 5.

> " **WORD EMBEDDING PLACES WORDS IN A HIGH-DIMENSIONAL SPACE THAT IS IMPOSSIBLE TO VISUALISE AND HARD TO UNDERSTAND**

is provided with both simple and full-featured versions of blocks.

The blocks are available as a Snap! library, as a Snap! project that illustrates typical uses, and as an interactive guide. These webpages are part of a larger library of AI Snap! blocks that was produced as part of the EU-funded eCraft2Learn project. All this can be

Finding biases in AI

Determining how close words are to each other relies on data from AI programs trained on all Wikipedia articles, and a large web crawl. These pages reflect societal biases. We can create Snap! programs to uncover these biases. For example, if we look at the distances between the word 'president' and male

words such as 'man' and 'boy', and compare these to the distances between the word 'president' and female words such as 'woman' and 'girl', we find that 'president' is closer to the male words than the female words. The opposite is true for the word 'maid'.

Solving word analogy problems

An amazing aspect of word embeddings is the way they can be used to solve word analogy problems. For example, we can solve the problem 'Cat is to kitten as dog is to X.' We can solve for X as (cat − kitten) + dog. In (cat − kitten), we are calculating the youth of the word. If we then add that to 'dog', we get a 'young dog' – or in other words, a puppy!

Word analogy problem-solving using word embeddings is not limited to the meaning of words. Word embeddings also encode how words relate to one another. So, for example, they can solve 'Fast is to X as slow is to slower.'

Biases can also be found in word analogy solutions. Soon after researchers discovered how word embeddings could be used to solve analogy problems, it was ▶

■ Technology such as Amazon's Alexa is based on natural language processing
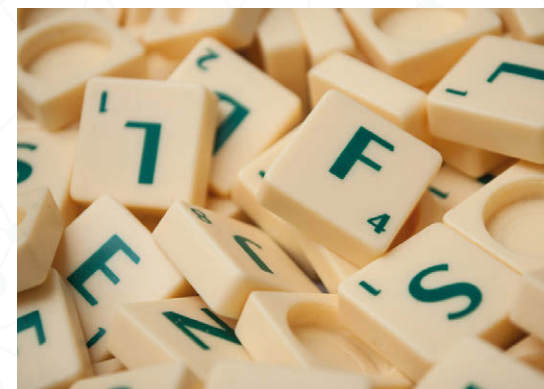
pointed out that their database declared that 'Man is to doctor as woman is to nurse.' In an effort to eliminate this and other similar biases, our Snap! blocks are based upon a newer database.

Our Snap! blocks used the databases found at fasttext.cc, an improvement over the original word embedding work called word2vec **(helloworld.cc/word)**. Both projects involved training a machine learning system to predict which words had been removed from sentences. One or two words per sentence were randomly replaced by a blank token and the program was trained to compute what were the most likely missing words. After training with billions of sentences, a part of the neural network was extracted to create the word embedding databases.

Possible projects

Some suggestions for projects using these word embedding blocks are:

- Use word embeddings to explore the similarity of sentences. One idea is to average all the words in the sentence. This is called the 'bag of words technique' as it ignores the order of the words, just as if they were put in a bag.
- Find a chain of similar words by finding the nearest word to the starting word. Then repeatedly find the nearest word to that word, while making sure never to repeat the same word. Use this to repeatedly change random words one at a time in a famous poem or text (such as 'roses are red and violets are blue').
- Make word games using word embeddings. For example, try something like Semantris **(helloworld.cc/semantris)**, a semantic version of Tetris. Working towards a bilingual version of Semantris might be a good idea.
- Create a program that searches for new word analogies. Hint: If A is to B as C is to D, the distance A–B 'is close to' the distance C–D.
- Search for biases that arise from the way people write about things. See if any biases you come across also apply in other languages.
- Use word embeddings as a way to input textual data into the deep neural networks. The deep learning models can be created with other Snap! AI blocks. This opens up a wide range of natural language processing possibilities.

Availability

All of these resources work in modern browsers. They are free and open-source. They can be run on a local web server without any internet connection. Visit **helloworld.cc/craft** for more information. (HW)

20000	A	B
1	dog	-0.917162
2	dogs	-0.798859
3	cat	-0.758230
4	pet	-0.740171
5	puppy	-0.739180
6	canine	-0.711131
7	pets	-0.709589
8	cats	-0.705311
9	animal	-0.683428
10	kitten	-0.678030
11	puppies	-0.652868

■ Snap! blocks answer questions about words by doing arithmetic

MACHINE LEARNING IN THE PRIMARY CLASSROOM

Nicholas Hughes explains why it's important every child explores the technology for themselves, and how he's broaching the topic with his classes

Machine learning is all around us, in our homes, on the websites we use, and in our pockets. More parts of our lives are being affected by it, from suggestions on streaming services and shopping sites to automatic tagging of our photos. How does all this happen? How do we know that the answers are right? What data is being used to make suggestions? We all need to have an awareness of data quality and bias. I believe all children and adults should have an understanding of what machine learning is and how it works.

After hearing Dale Lane talk about the resources on his website, **machinelearningforkids.co.uk**, I was keen to get my students exploring the technology. I was struck by how accessible the activities are and how they can be used to explore some of the wider implications of machine learning in class. It is also a fun and exciting way to engage children with computer science concepts, through ideas such as making their own version of a smart assistant.

I started simply and used Dale's 'Make Me Happy' resources to create a program that would recognise nice and nasty phrases (see pages 80–82). This linked well to some work we had just been doing: programming a face to react to such phrases. In this activity, a face reacted to specific phrases that the class offered up. I had asked the class how we could make the face react to any phrase, and they agreed that it was impossible to write everything, so this was a great opportunity to introduce machine learning as a solution.

We spent most of the lesson creating the model. We started by discussing that we needed to collect data for nice and nasty things and place them in the right buckets. They learnt that machine learning models don't always get it right, so more data is required to retrain the model.

NICHOLAS HUGHES

Nicholas is a Computing teacher at Latymer Prep School and an Educational Consultant at 3BM Education. He is a CAS Master Teacher and a Raspberry Pi-Certified Educator (**@duck_star**).
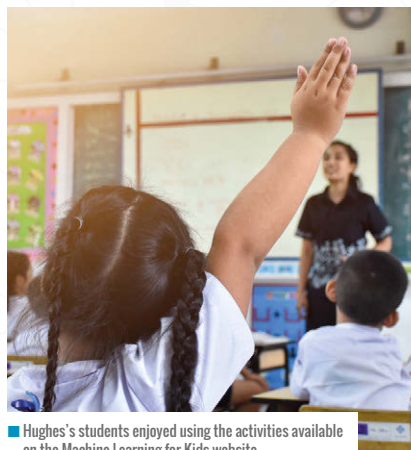
■ Hughes's students enjoyed using the activities available on the Machine Learning for Kids website

MACHINE LEARNING RESOURCES

If you want to learn more about machine learning and investigate some resources, I recommend:
- Dale Lane's (@dalelane) Machine Learning for Kids - **machinelearningforkids.co.uk**
- Code.org AI for Oceans - **code.org/oceans** - great videos that are easy to understand
- Quick Draw - **quickdraw.withgoogle.com** - can Google recognise what you have drawn?
- Teachable Machine - **teachablemachine. withgoogle.com** - train a computer to recognise your own images, sounds and poses

It provided an excellent springboard to discuss where else machine learning models exist, e.g. spam filters, photo tagging, smart speakers, etc.

The Machine Learning for Kids website contains prebuilt Scratch projects to demonstrate how the machine learning model can be used. There are also activities from the other resources mentioned above. These provide children with opportunities to explore and start thinking about the implications for machine learning use.

Machine learning is all around and providing opportunities for children to explore what it is and to make their own projects is an important step in ensuring children have a greater knowledge and understanding of this technology. (HW)

MACHINE LEARNING COURSES TO DIVERSIFY COMPUTING

Tayler Kirk explains how new courses from Apps for Good could encourage more girls and under-represented groups of students to choose to study computing

With an already brimming Computing curriculum, some teachers may be hesitant about the prospect of teaching machine learning. The technology may seem too daunting for many educators to broach with their students. The concept centres on a computer rather than a programmer, deciding and developing its own set of instructions on how to sort, organise, classify, and use information. There's a lot of terminology that can be off-putting, and at its heart there are algorithms that may be beyond most people's grasp.

However, machine learning is revolutionising our interactions with technology, from running complex algorithms to utilising secure biometric data, and is now entering the educational landscape. By exposing students to machine learning, we unmask the everyday mysteries of how technology is influencing and arguably improving our lives. By teaching students how Spotify is able to pick the perfect song, how Netflix knows what you want to watch, and how chatbots have averted the need to ever speak to a human again, we can foster a real enthusiasm for technology.

TAYLER KIRK
Tayler is Education Outreach and Engagement Manager at Apps for Good.

And this inspiration is crucial. Although figures from the Joint Council for Qualifications (JCQ) have shown a rise of 7.2% in GCSE Computing qualifications, just 21.4% of these were achieved by female students in 2017. This figure is even worse at A level: only 9.8% of those completing a Computing course were girls.

If we can make teaching technology relevant to our students' lives rather than abstract and dry concepts, this trend can be reversed. Interventions that highlight the relevance of science to students' lives have been shown to enhance their interest in the subject, particularly for students who are most at risk of being disengaged from school. And a 2011 study of students in California showed that the strongest predictor of girls' interest in computing is the extent to which they see its value.

Depth and breadth

At Apps for Good, we recognised how powerful machine learning courses could be in engaging students who might not otherwise have considered computing to be a subject for them. Over the course of 2019, we have developed two courses in which students work in teams to build and use a machine learning model to help solve a real-world problem. Students begin by learning the core concepts of machine learning, before going on to design and code their own models.

Our courses are designed with an interdisciplinary approach that allows students to develop technical skills such as design and coding while also nurturing soft skills such as teamwork, problem-solving, and presenting. While the course content doesn't shy away from exploring

■ Apps for Good have developed two machine learning courses in which students work in teams

the inner workings of machine learning algorithms (students work on activities such as decision trees and k-means clustering, among others), we believe that it is also important to expose students to the ethical considerations of such technology. So our activities also encourage and facilitate debates such as whose safety should a driverless car prioritise, the implications of storing biometric data, and whether the police should be allowed to use facial recognition technology. Opportunities to present their thoughts and findings are interspersed throughout the courses to allow students to practise sharing their ideas in an articulate and confident way.

Two students from Speyside in Scotland recognised the immense power that

■ Isla and Phoebe from Speyside designed and began to prototype a machine learning model (Ultra IR) that would use image recognition to identify abnormalities in foetal ultrasound scans

" MACHINE LEARNING IS NOW REVOLUTIONISING OUR INTERACTIONS WITH TECHNOLOGY

machine learning could yield and how this could be used to assist in maternal care for women in rural communities. Isla and Phoebe designed and began to prototype a machine learning model (Ultra IR) that would use image recognition to identify abnormalities in foetal ultrasound scans. As the machine was exposed to more examples of ultrasounds it would become more attuned to pinpoint anomalies, particularly those that could be missed by human error.

Their teacher, Marc McWhirter, said, "I was thrilled to take part in the first year of Apps for Good's machine learning course. I was initially worried about the learning curve, but the team at Apps for Good presented the material and practical activities in a way which made the topic far more accessible than I had anticipated. The topic certainly fired up the imaginations of my pupils, who are always keen to be tech pioneers."

Lilli Hutchinson is a teacher who has been delivering Apps for Good courses for over eight years. Her department has

chosen to begin delivering machine learning via the programme.

"We're really excited to be delivering App for Good's machine learning course to five year nine classes this year. We have been delivering the app development course for the past eight years and wanted to try something new. The machine learning course is cutting edge and includes a good amount of programming and computation theory to fit in with our overall ambition of encouraging our girls to both enjoy and develop their computing knowledge and skills. The fact that the students can speak with experts in the field through the Apps for Good programme adds a real boost to the team's engagement, too."

Free to access
We work with over 650 schools and colleges across the UK and recognise that every teacher is in a unique position and that our Standard courses, which last between ten and twelve hours, may not be suited to everyone. For these teachers who may be pressed for time, our 'In a Day' workshops allow them to explore skills of app design or machine learning in a condensed five-hour course that goes through the same process of ideation, design, build, and present, but within a single school day.

Our courses are free of charge to access and published under a Creative Commons licence. They come fully resourced with

student workbooks, slides, lesson points, and primers so busy teachers can quickly access and deliver our materials completely free of charge. All our materials have been designed by a former Computer Science teacher and are mapped against the Computing curriculum, so they can be delivered in lesson time or as an extracurricular activity. Each session has a lesson primer so teachers don't have to spend time researching topics and can confidently deliver sessions on topics that they may not have previously come across.

Although all our courses, including our 'In a Day' workshops, are designed to be as easy to deliver as possible, we know that some schools may not feel confident to deliver a course. Since February 2018, members of Apps for Good's staff have been going into schools in challenging circumstances across the UK to deliver 'In a Day' workshops so that teachers can use the experience as a training opportunity and then feel comfortable to deliver the workshop to additional classes. Although we are only a small team of ten, with two members of staff facilitating sessions, we are hoping that 2020 will see an expansion of our delivery capacity and we will be able to reach more students than ever before and support educators in their delivery of our courses and workshops.

You can access all the Apps for Good course materials for free after registering, for free, at **appsforgood.org**. (HW)

■ The Get Creative With Data course was launched with a preview event at Tapton School in June 2018
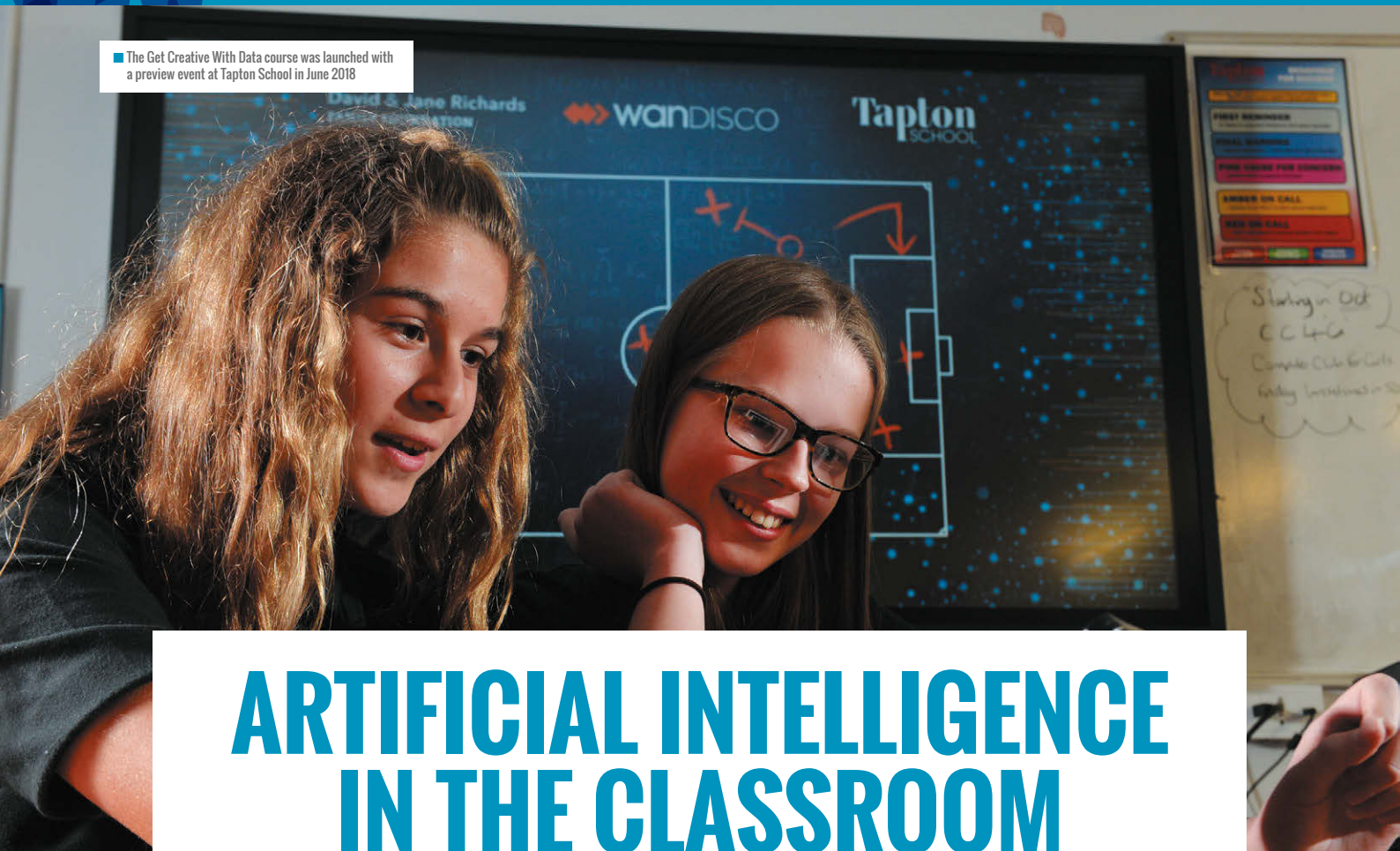
ARTIFICIAL INTELLIGENCE IN THE CLASSROOM

Emily Dreimann shares some ideas for kick-starting discussions on the technology

T here's little doubt that artificial intelligence has captured our collective imagination. TV series and films increasingly explore the implications of this technology, from family favourites like WALL-E to the distinctly darker Black Mirror. Robotics companies, meanwhile, are beginning to transform the more benign of these visions into reality: Hanson

EMILY DREIMANN

Emily works for the David and Jane Richards Family Foundation as project manager for the Get Creative With Data course (@ **em_dreimann**).

Robotics' lifelike robot Sophia, for example, has become a familiar sight on talk shows and has even starred in music videos.

At the same time, our understanding of how we personally interact with AI in our day-to-day lives, and how we can use it to our advantage, remains limited. Ask a class of 13-year-olds what they think of when they hear 'artificial intelligence', for example, and the answers tend towards a common theme: 'creepy'; 'sinister'; 'taking over the world'. How do we retain interest levels while grounding AI in reality and preparing students for the workplace of the future?

For young people to be able to lead and succeed in the data-driven economy, a strong understanding of this ever-evolving technology is paramount. In order to engage students with this topic, lessons should not only highlight the many forms that artificial intelligence can take in the real world, but also offer tangible experience of

and interactions with the technology. Here are just some of the angles from which this topic can be approached, and suggestions for resources that can complement them.

AI in action

With the recent proliferation of smart speakers and virtual assistants, this technology can be a useful framework for an initial discussion around the key tenets of artificial intelligence. Most young people will have been exposed to these devices in some form; fewer, however, are likely to identify them as an example of AI. You could ask students:

■ What does AI look like?
■ What does it sound like?
■ To what extent should it mirror human behaviour?

Google's Duplex AI assistant is a great example of the capabilities and potential of this technology. In a popular video of the

■ Students from Astrea Academy Sheffield saw the production of custom-made sensor boards

assistant in action, it is heard making calls to a number of different businesses, sounding sufficiently humanlike in its interactions to fool the real humans on the other end of the line. The somewhat unnerving potential of the AI is likely to hook students, while it remains grounded in reality as an aid and time saver, rather than a replacement, for humans.

Quick, Draw! is a great resource for highlighting to students another manifestation of AI. The game challenges the user to create a series of doodles, while a neural network attempts to guess what they are drawing. Coupled with its hands-on, accessible nature, this activity has the potential to engage even the most reluctant of students.

Not so intelligent?

At the same time, we need students to be critical in their appraisal of AI technology. The mantra that 'Machine learning is written in Python; artificial intelligence is written in PowerPoint' is a great starting point for this discussion. You could ask students what they understand by this. The aim here is to draw out the idea that machine learning can already be seen in action in industries across the globe, while AI arguably remains a theoretical concept. Has anyone truly created an intelligent machine?

There is a wealth of resources that we can draw on to assist students in forming their own opinions in this debate. The Turing test, for example, is an important concept for students to understand and remains a useful benchmark against which to measure the capabilities of AI technology. Encourage your students to read aloud some of the transcripts for entries to the most recent Loebner Prize, an annual Turing test competition. Would any of these convince them that they were speaking to a real human?

On the website AI Weirdness, meanwhile, research scientist Janelle Shane publishes the entertaining results of her experiments

> " **AI ARGUABLY REMAINS A THEORETICAL CONCEPT, WHILE MACHINE LEARNING CAN ALREADY BE SEEN IN ACTION IN INDUSTRY**

training neural networks on existing content across a range of topics, from cat names to knitting patterns. Taking Halloween costumes as an example, you could allow students to explore; with suggestions such as 'sentient stone' and 'a skunk in a moose suit', it should quickly become apparent to them that this technology has some way to go in capturing the uniquely human traits of creativity and humour.

Branching out

AI and machine learning can also be ideal starting points for generating lively debate around other key topics in Computing. An unplugged activity in which students create their own algorithm to guide a visitor from the school reception to their classroom

is an ideal catalyst for a discussion about the differences between how people and machines make sense of instructions. The ability of humans to apply common sense when determining a course of action can be highlighted as a strength that machines are unable to emulate, which can encourage students to understand AI as a tool to complement us rather than compete with us.

Meanwhile, developments in self-driving vehicle technology presents a unique opportunity for students to explore ethics in the context of computer science. Moral Machine, developed by the Massachusetts Institute of Technology, is an interactive tool that asks the user to judge the most acceptable outcomes of a series of moral dilemmas faced by a driverless car. By engaging with this modern take on the classic trolley problem, students develop a deeper and more personal understanding of the ethical challenges surrounding artificial intelligence.

These activities have been popular in our partner schools, with teachers commenting that pupils were "fully engaged" and found the topic "really interesting". By providing students with an interactive forum in which to discuss and explore AI, we have an excellent opportunity to support the next generation in confidently claiming their place in the modern world. (HW)

■ Students explored equipment such as circuit-building robots and laser cutters

YOU AND AI: RESOURCES FROM THE ROYAL SOCIETY

Michael Jones explores how AI and machine learning can be embedded in Computer Science lessons and the wider curriculum

Is artificial intelligence the same as machine learning? As a computer science teacher, I might be expected to have an answer to that question. However, the more I consider it, the less sure I am of my answer. The term 'AI' is embedded in our language and consciousness, as is, to a lesser extent, 'machine learning'. Yet, like me, most people do not have a clear idea of what the concepts mean. Working with AI in my role as a teacher, I have some understanding of the mechanics of AI. Like many, I have a feeling of unease that machines may be making decisions for me.

Looking back to the 2018 Facebook–Cambridge Analytica scandal, it is all too clear that algorithms can be used to influence how we think and conduct our lives. With this concern about AI and machine learning

in mind, the Royal Society, supported by DeepMind, published 'You and AI', a summary of a 2019 public lecture series that explored topics as diverse as the future of work, bias, and how medical diagnosis could be impacted by AI and machine learning.

The full lecture series can be found here: **royalsociety.org/topics-policy/projects/machine-learning/you-and-ai**.

Getting to grips with AI

For teachers wanting to develop a wider understanding of AI and machine learning, these lectures are a great first stop. Viewed alongside the 'You and AI' report (**helloworld.cc/youandai**), they will give anyone interested in the topic a balanced and deep appreciation of the technology involved and the challenges it brings.

Teachers in the UK may want to consider using the full-length series for A level Computing and Key Stage 5 courses such as Cambridge Technicals, for example Unit 17 'The Internet of Everything', or BTEC Computing (Unit 9 'The Impact of Computing'). This could be undertaken in conjunction with projects focusing on the use of AI hardware and software such as the Jetson Nano and Amazon DeepLens.

The response to the report and lecture series has enabled teachers from the Royal Society Schools Network to create a body of classroom resources. These come in two forms: short video clips and lesson resources.

Moral decisions in the classroom

The short clips from the lecture series (the longest is four and a half minutes) each focus

on a particular topic. The beauty of these is their wide-ranging use. They are suited to a cross-curricular purpose; for example, 'You'll need your hat and coat' fits neatly into the wider science curriculum focusing on how AI can be used to forecast weather. Clips such as 'What kind of world do we want?' and 'When is a criminal not a criminal?' provide a PSHE focus. The relatively short length of the clips and their 'big question' focus make them well-suited to debate and starter scenarios. As with most 'big question' topics, there is likely to be a diversity of answers. That can only be to the good and helps to ensure that our students are exposed to questions and ideas that will undoubtedly impact their lives.
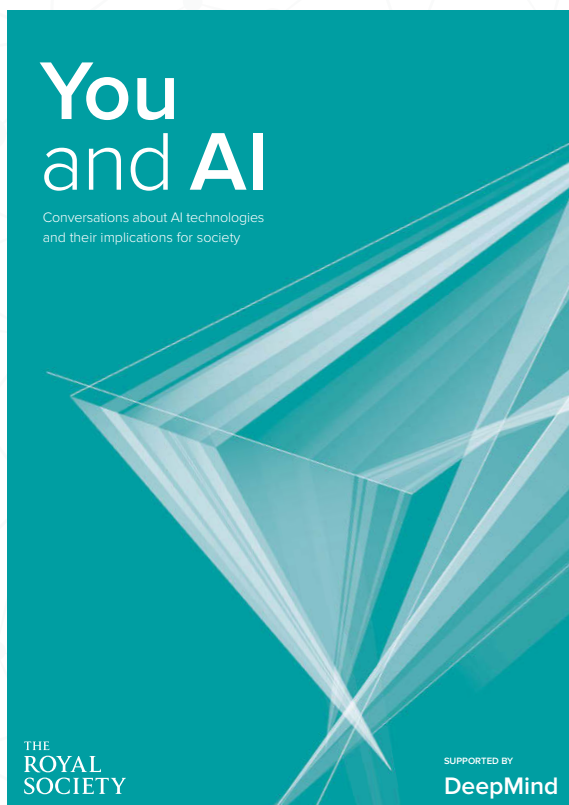
The lesson resources consist of lesson plans spanning Key Stages 3 and 4. These are highly detailed and designed to engage and raise interest. The KS3 package 'How do we stop machines enslaving us?' provides students with an opportunity to investigate decision-making algorithms and the life-and-death consequences they may have. I encourage you to test your moral judgement using the MIT Moral Machine (**moralmachine.**

change?' package is an example of where computer science and science converge. The resource is linked to the AQA, Edexcel, and OCR science specifications. Linked to the 'Practical applications of AI', the resource investigates ocean acidification and looks at how AI can be harnessed to provide us with greater understanding and perhaps with the means to prevent the damage occurring. The resource is supplied with ready-to-go examination questions.

> ## MOST PEOPLE DO NOT HAVE A CLEAR IDEA OF WHAT AI AND MACHINE LEARNING MEAN, DESPITE THEIR PREVALENCE ISOCIETY

mit.edu). Starting from an individual viewpoint, students contribute to a wider group rubric about the value of lives and the meaning of responsibility.

The uses of neural networks and classification are explored in the 'How do machines learn?' package. Here, students learn what an algorithm is and how classification rules are weighted. The example given focuses on the classification of butterflies (see p.83 for more on this) and the features that may lead to an object being classified as a butterfly or not. The resources are suited to adaptation, recognising facial expressions or trees versus teacups, for example. Mathematical skills are incorporated in this via the concept of weighting a decision: if a feature exists, it assumes a weight of 1. Non-existence assumes a weight of 0. The closer the average of the weights is to 1, the more likely it is that the object is a butterfly. This is a highly practical activity and could be used as an unplugged classroom activity.

Taking Computer Science into Science at GCSE
At Key Stage 4, the 'Can AI predict the future of climate

Supporting the videos and lesson packages are an informative animation and two interactive infographics. Give the identification activity a go and see if you can beat Google. The immediately accessible content in the infographics will work well from Key Stage 2 upwards. They act as stand-alone materials in which students can explore the topics and questions and test themselves as they progress.

The Royal Society is committed to supporting schools, not just in computer science but as part of a much wider science remit. The Schools Engagement team has a growing schools network across the UK, providing opportunities such as access to funding opportunities for student research projects, student conferences, and lesson resources at both primary and secondary level. To become part of the network and receive their newsletter, sign up at **royalsociety.org/schools-network**. (HW)

YOU AND AI RESOURCES

Lecture series summary: **helloworld.cc/youandai**
Video clips: **helloworld.cc/society**
Teaching suggestions: **helloworld.cc/resources**
Interactive infographic: **helloworld.cc/info**

You and AI

Conversations about AI technologies and their implications for society

THE ROYAL SOCIETY

SUPPORTED BY
DeepMind

MICHAEL JONES
Michael is Director of Computer Science at Northfleet Technology College. He is a Specialist Leader of Education, a Raspberry Pi Certified Educator, and CS Champion for the NCCE (**@MikeJonesCSTalk, community.computingatschool.org. uk/users/222**).

#INSIGHTS

STORYTELLING IN COMPUTING EDUCATION

STORY BY Hayley Leonard

Children's traditional stories, nursery rhymes, and literature provide a rich source of sequences and repetition. At the 14th Workshop in Primary and Secondary Computing Education (WiPSCE 19) in Glasgow last October, Sarah Twigg and colleagues presented an approach that uses children's literature to teach Computing to primary school pupils. The team from Lancaster University hopes the familiar contexts of children's stories will engage pupils and raise the confidence of non-specialist teachers in delivering the curriculum.

Identifying computing concepts in children's literature

The team reviewed 50 popular children's picture books to identify key computing constructs, namely sequencing, repetition, and selection. The books included programming constructs to varying degrees, but 16 books included all three constructs, and all 50 books could be used to demonstrate sequencing. Several books were then used as the basis for sample teaching resources, which were trialled in some classrooms and code clubs.

The teaching approach: Read, Act, Model, and Program (RAMP)

An approach named RAMP builds up subject knowledge and appropriate vocabulary in a storytelling context. The format begins by reading through the story; the teacher asks questions about what

is happening and introduces computing terminology. Children then act out the story and are asked about repeating patterns of behaviour and what triggers them. The model element of the approach then involves constructing the sequence of events in the story, using either images from the book or printouts of lines of code or blocks from Scratch. Children are asked to identify repeating patterns and choice points in the story, and links are made to the computing terminology throughout.

The program step is supported by the sample lesson resources developed for specific books. It involves using the computer to produce the program that has been designed through the previous unplugged activities.

Teachers' evaluation of the resources

Responses from teachers who were asked about their experiences were highly positive. In particular, teachers said that the first three stages (read, act, and

---

LESSON EXAMPLE

Computing concepts represented in *We're Going on a Bear Hunt* (Rosen, Oxenbury 1989):

Sequencing: A list of events to be followed in order.

Example: The characters on the bear hunt go through six different environments in order. On their way back home, they go through the environments in reverse order.

Repetition: At least one example of a pattern of repeated dialogue, actions, or environment.

Example: Repetition of particular phrases in each environment, for example "We're going on a bear hunt."

Selection: At least one example of a choice of dialogue, actions, or environment.

Example: The end of the repeated dialogue varies depending on the environment, for example, in the river they say, "Splash splosh!".
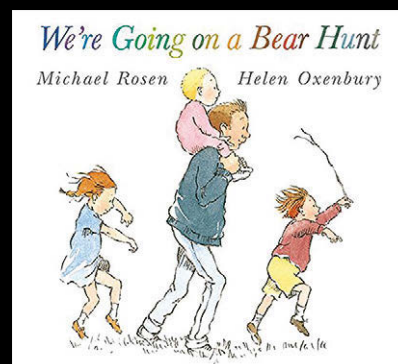
> ## " THE APPROACH IS LOW-COST AND USES FAMILIAR CONTEXTS

model) were very engaging for the pupils. They provided multiple opportunities for differentiation and working together at different levels of ability.

Some teachers suggested, however, that some non-specialist computing teachers might find the step up to the program stage daunting. The authors are using this feedback to help them develop the teaching resources. They are continuing to work with teachers to investigate whether more support is required for the transition to the final stage, or whether this should be separated from the other elements of the approach.

An inclusive approach

Teaching computing principles to young children can be a challenge. A creative storytelling approach is low-cost and uses familiar contexts that are intuitive to teachers and parents. Twigg and her colleagues suggest that this approach has benefits for diversity in computing: they are particularly interested in its use with disabled students. The collaborative activities and active discussion involved could also be beneficial in engaging more girls in computing. A pilot study that tests this approach more widely will start in England in September 2020, as part of the Gender Balance in Computing programme of research. (HW)

FURTHER READING

✔ Twigg, S., Blair, L., and Winter, E (2019), Using children's literature to introduce computing principles and concepts in primary schools: work in progress (helloworld.cc/literature)

VIRTUAL TEACHING ASSISTANTS HELP STUDENTS LEARN

STORY BY Thom Kunkeler

L earning in classrooms is usually guided by a teacher, but new research has shown the promise of using virtual assistants. A study has tested the effects of an open learner model, in which students can view and reflect upon their level of mastery in different aspects of problem-solving. Students can then choose which problems to work on in order to progress. Yanjin Long and Vincent Aleven, who conducted the research in two US classrooms, found that students who had control over their learning in this way showed significantly better equation-solving skills and enjoyment levels than their counterparts.

Over the past decades, the field of Intelligent Tutoring Systems has emerged with the goal of investigating how artificial intelligence techniques can be beneficial to the personalised needs of individual learners, while helping teachers out in classroom instruction. In this respect, open learner models are learning environments that allow students to access content with varying levels of interactivity. For this study, Long and Aleven investigated how an open learner model can support students' self-assessment of equation-solving skills, and whether shared control over problem selection ensures better learning gains and enjoyment of learning. They designed a user interface using Cognitive Tutor Authoring Tools (CTAT) in which each problem is broken down by the student into steps, providing step-by-step guidance. The tool was also used to provide feedback for each step, and students could request hints for what to do next.

As learners advanced through the problems, their skill bar progressed, turning gold when students completed over 95% of the assignments. In addition, the researchers added self-assessment prompts where students could express their confidence in solving a type of problem. To understand how the open learner model relates to learning outcomes, the classrooms were divided into four groups, with students who had access to the open learner model and control over their problem selection, and students who did not.

The researchers found that students who learned with the open learner model needed significantly fewer steps and problems to reach the mastery criterion, and had fewer incorrect attempts. In addition, the self-assessment prompts might have led students to reflect on the skills they just practiced, or stay more focused on practising. The presence of an open learner model can therefore lead to significantly better learning gains when shared control over problem selection is granted to the student. **(HW)**

> " STUDENTS ACCESS CONTENT WITH VARYING INTERACTIVITY

VIRTUAL TEACHING ASSISTANTS

- Cognitive Tutor Authoring Tools: helloworld.cc/tutor

- Lynnette: helloworld.cc/smarter

- MagicBox: helloworld.cc/box

FURTHER READING

✓ Long, Y. and Aleven, V. (2017). Enhancing learning outcomes through self-regulated learning support with an open learner model **(helloworld.cc/open)**

Level 1	Level 2	Level 3	Level 4	Level 5
Examples:	Examples:	Examples:	Examples:	Examples:
$x+3=5$ $y+1=4$	$3x+1=7$ $2y+1=9$	$3x+1=x+5$ $4y+2=y-1$	$2(x+1)=4$ $3(y-1)=6$	$2(x+1)+1=5$ $3(y-1)-1=5$
Mastered	Mastered	22% Mastered	25% Mastered	25% Mastered
Problems solved: 4	Problems solved: 3	Problems solved: 2	Problems solved: 0	Problems solved: 0
		Get One Problem	Get One Problem	Get One Problem

Lynnette is an intelligent tutoring system with an open learner model that displays skill mastery in the forms of skill bars

ACADEMIC INSIGHTS FROM THE WIPSCE CONFERENCE

STORY BY Oliver Quinlan

One of the key events for Computing Education research in Europe is the Workshop in Primary and Secondary Computing Education (WiPSCE) which took place last October in Glasgow. Sue Sentance and I, from the Raspberry Pi Foundation, headed to Glasgow to hear from colleagues in computing education and share some of our work.

Academic conferences have a particular format and although they often have keynote speakers covering big issues, they largely involve participants presenting their own work and inviting feedback and discussion on it. Presenters submit papers about their research projects and findings, and then present the content of these papers for other presenters and attendees to learn from but also to critique.

WiPSCE's two keynotes started each day off with a big issue. On day one, Beth Simon from the University of California raised the big questions about training a new generation of computing teachers, and how we should be looking to the future. On day two, my colleague, Sue Sentance, identified the potential for embedding computing into a national education system, with the case study of the National Centre for Computing Education in England.

The conference started with a focus on primary education, and several papers explored young children's perceptions about programming and technology. There is a lot of rhetoric about children growing up around technology and developing expertise with ease, but the authors of these papers took an objective view of how children's perceptions are developing with some very interesting discussion.

On the second day I presented the work we have been doing at the Raspberry Pi Foundation, exploring approaches to assessment during informal learning. Presentations at academic conferences are often met with more challenge than at other types of events, with much of the value coming from discussing and developing the ideas. The diverse perspectives of attendees were really helpful in pointing out some aspects of our paper that I hadn't considered, and I found this constructive challenge very helpful.

As well as paper presentations there were also workshops, and I attended the session focusing on ScratchMaths. This project aims to teach maths skills through computing, and specifically programming in Scratch. Celia Hoyes showed us how they have developed materials to successfully address both subjects. Looking at programming through the lens of another subject, and discussing this with others, was a good chance to be put in the position of the students we teach.

As with any conference, it isn't just the formal sessions that are valuable, but also the chance to meet and discuss with new people working in similar, but different areas to your own. WiPSCE was a very friendly conference in this sense, and I enjoyed some thought-provoking conversations from people working on school-based computing education across the world. Despite it being an academic conference, there were also lots of practising teachers at the event, which brought a real intersection of views from theory and practice together.

For educators, academic conferences might seem like something rather different, but those like WiPSCE can provide experiences to engage with the deeper and more emerging themes in your subject. You will rarely find quick fixes or off-the-shelf resources at such conferences, although there were some tools on offer here which I could see being used in the classroom by enquiring teachers. What you will find are presentations that really make you think about the underlying issues that our subject faces. **You can find out more about future WiPSCE conferences at wipsce.org.** (HW)

■ WiPSCE 2019 was held at Glasgow University

SUBSCRIBE

SIGN UP TODAY FOR **FREE**
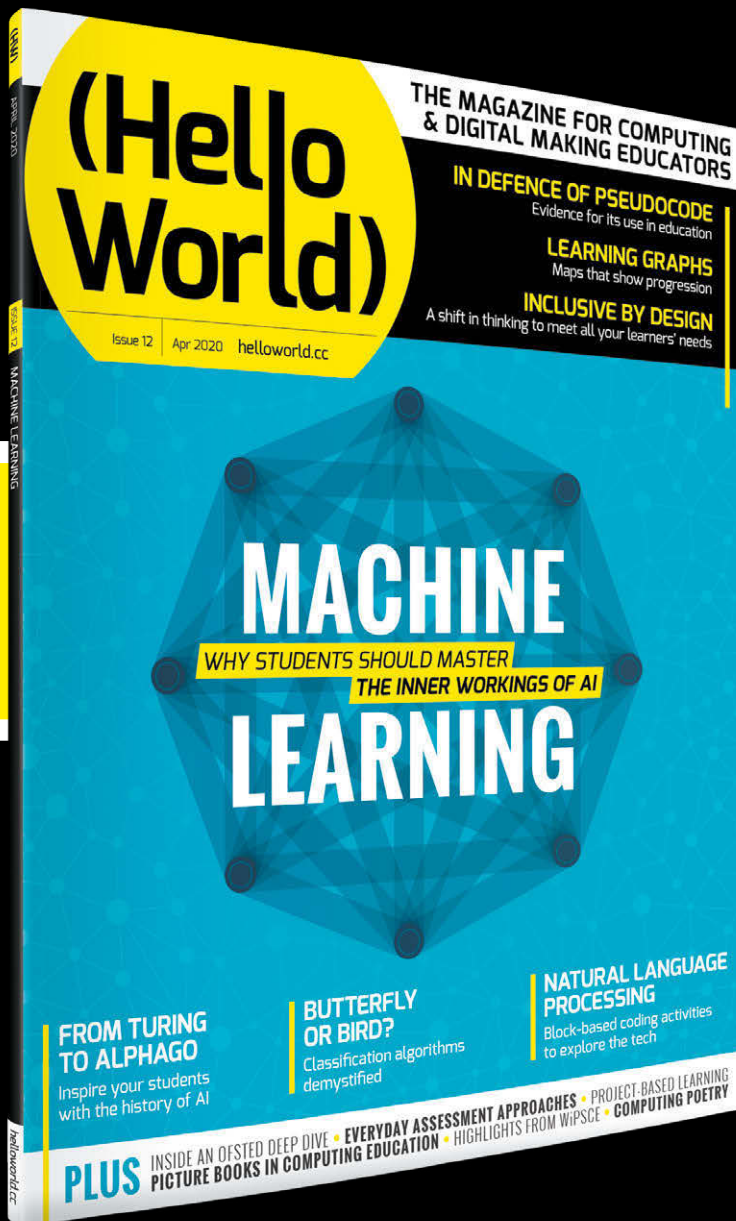
FREE
IN PRINT
for UK-based
educators

SUBSCRIBE TODAY

- **Published four times a year**

- **Have them delivered directly to your door**

- **Hello World is not available in stores**

COMPUTING AND THE REVISED OFSTED INSPECTION FRAMEWORK

Ofsted's new inspection framework emphasises 'quality of education'. **Neil Rickus** examine what this means for teachers, subject leaders, and pupils

S chools in England have been inspected under a revised Ofsted inspection framework since September 2019 and, while there have been various reports about what an inspection now entails, including subject-specific 'deep dives', only a few schools have actually been inspected under the new framework. In this article, we discuss the key changes to the framework and share the details of a visit to Wembley Primary School in London to find out about their recent experience of being inspected.

Quality of education

Chief Inspector Amanda Spielman recently outlined the need for schools to have a broad and balanced curriculum, which is reflected in the new judgement category of quality of education. This category outlines how schools should focus on the intent, implementation, and impact of the curriculum, which can be briefly defined as:

■ **Intent:** The focus here is on the curriculum and aspirations for learners. All learners should study an ambitious curriculum that is rich in both knowledge and skills and offers suitable progression across the school.

■ **Implementation:** This is about how teachers do their job and how leaders support them, and focuses on assessment and feedback, pedagogy, subject knowledge, and the quality of teaching.

■ **Impact:** This looks at how learners develop their knowledge and skills across the curriculum and, as a result, their achievement. It is less focused on data than the previous inspection framework, and success may not be purely academic.

The new framework also includes an emphasis on cultural capital, with the inspection handbook stating:

"It is the essential knowledge that pupils need to be educated citizens, introducing them to the best that has been thought and said and helping to engender an appreciation of human creativity and achievement."

Within computing, this could perhaps mean focusing on:

■ **Technology:** How it can be disruptive and improve lives, along with negative impacts
■ **People:** Ada Lovelace, Alan Turing, Grace Hopper, Bill Gates, Steve Jobs, etc.
■ **Opportunities:** Extracurricular activities, going on school trips, entering events and competitions, talks by or visits to employers and educational providers

Intent

In order for schools to have a broad and balanced curriculum, foundation subjects such as computing need to be studied at appropriate depth and given sufficient curriculum time. In order to discuss and evidence how computing is delivered, thorough and accessible schemes of work, including the progression in skills and knowledge, could be part of a subject leader folder (see box on p.38). Cross-curricular links and opportunities to develop cultural capital could also be identified.

To find out more about the realities of a recent inspection, I visited Eleanor Haines and Satya Maremanda, computing coordinators at Wembley Primary School (WPS) in London, whose September 2019 Ofsted report **(helloworld.cc/ofsted)** outlined how well-thought-out the school's computing curriculum is (see p.2).

At WPS, a curriculum map (see photo on p.38) specifies the topics and technologies used by each year, per half term. As some staff members lack confidence in delivering computing, lesson plans with step-by-step instructions are also provided, which link directly to the programming environment used throughout the school. Providing planning in this way helps teachers focus on the required skills and knowledge within each session, rather than the underlying technology, to avoid 'We're doing Scratch today.'

■ Eleanor Haines and Satya Maremanda are computing coordinators at Wembley Primary School, London, which had a recent Ofsted inspection under the new framework

> ## SUBJECTS SUCH AS COMPUTING NEED TO BE STUDIED AT APPROPRIATE DEPTH AND GIVEN SUFFICIENT CURRICULUM TIME

Time is spent in other curriculum subjects revisiting various key skills, although coordinators work closely with staff to ensure that technology is being used appropriately, rather than just for the sake of using it. Online safety content is predominantly delivered within computing lessons at WPS, with a focus at the start of every academic year on the school's acceptable use policy (AUP). The views of both pupils and parents on the computing curriculum and technology usage in school are regularly sought, with online safety being discussed with parents at a range of events taking place throughout the academic year.

To enrich the curriculum, WPS holds computing clubs for both Key Stage 1 and Key Stage 2 pupils, which are well attended and provide an opportunity for all pupils to use the school's technology away from the formal classroom environment, along with occasional enrichment days.
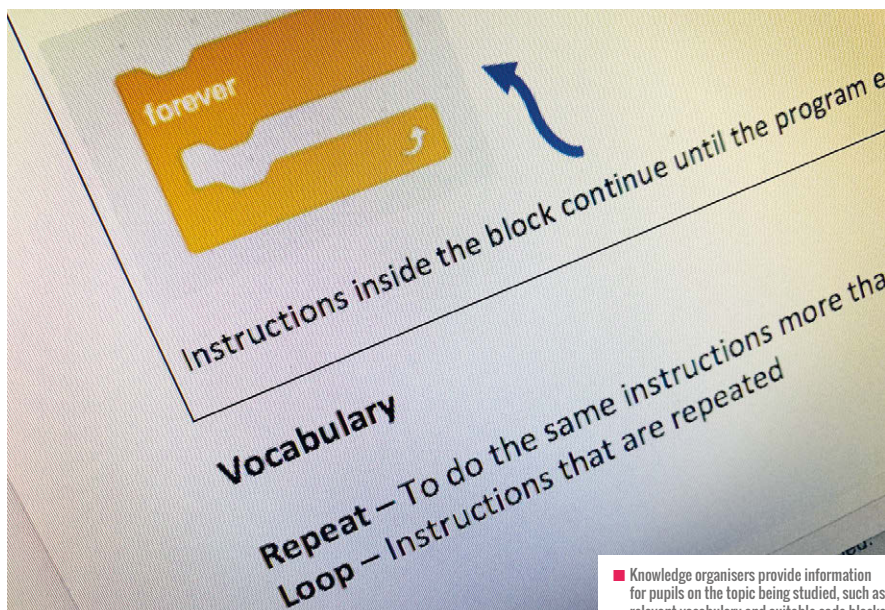
■ Knowledge organisers provide information for pupils on the topic being studied, such as relevant vocabulary and suitable code blocks

NEIL RICKUS
Neil is a Senior Lecturer in Computing Education at the University of Hertfordshire, a primary education consultant for the BCS, and the founder of Computing Champions. He is a CAS community leader, a CEOP Ambassador, and a Raspberry Pi, Google and Microsoft Certified Educator (**@computingchamps**).

SUBJECT LEADER FOLDER

A detailed, well-organised subject leader folder can enable intent, implementation, and impact to be evidenced, along with outlining how the subject is to be developed and facilitating discussion during an inspection. Many CAS community meetings this term will focus on the new inspection framework and enable attendees to discuss subject leadership in their school.

The contents of a subject leader folder could include:

- Development/action plan
- Observations of lessons
- Self-evaluation
- Professional development records
- Staff audits and surveys
- Pupil and parent questionnaires
- Budget and resource documents
- Progression documents
- Schemes of work
- Cross-curricular topic links
- Moderation/examples of students' work
- Online safety guidance

▶ Implementation

The implementation of the curriculum appears to be focused on what happens in classrooms on a day-to-day basis, rather than examining paperwork or consulting with the Senior Leadership Team (SLT). For example, inspectors may wish to view computing lessons delivered by a non-specialist teacher, or to discuss the subject with pupils. The coordinators at WPS outlined how inspectors were keen to find out more about the computing lessons taking place, including discussing with teachers where the current lesson fitted into a scheme of work, along with previous content taught and the next steps in pupils' learning.

To enhance teachers' subject knowledge, regular continuing professional development is provided at WPS, along with support from the coordinators, other staff and digital leaders (a small group of pupils confident in computing) as required.

■ Wembley Primary School's computing curriculum map outlines the topics studied and the technologies used within each year group

> ## " ALL LEARNERS SHOULD STUDY AN AMBITIOUS CURRICULUM THAT IS RICH IN KNOWLEDGE AND SKILLS AND OFFERS PROGRESSION

Further guidance is provided by the use of knowledge organisers, particularly for programming units of work, which outline the relevant blocks of code and vocabulary examined within lessons (see photo), along with scaffolding within the programming environment itself, such as tutorial videos.

Impact

Finally, the impact of the curriculum examines pupils' achievements in school, which is related to how they develop their skills and knowledge across the curriculum. Schools could consider how they can evidence pupils' work and link this to the relevant skills and knowledge. End points for pupils could also be identified, such as programs and content produced by children of varying abilities. For example, the inspection report for WPS praised the school for enabling Year 6 pupils to use their knowledge of computing to design their own games (see p.2).

Next steps

Many Computing at School (CAS) community meetings have the revised Ofsted framework as the focus for their next event. Details of your nearest event can be found here: **community. computingatschool.org.uk/events**.

CAS have created a Next Steps document, which includes a number of tasks for subject leaders and possible questions they may be asked during an inspection. The document can be found within the CAS resources pages: **community.computingatschool.org. uk/resources/.**

A more extensive interview with Eleanor and Satya from Wembley Primary School is available on the CAS website: **computingatschool.org.uk/.**

Has your school recently been inspected under the new framework? Perhaps the inspection outlined above differs from your experiences? Maybe you lead computing in a different way and would like to share your approach? Do get in touch on Twitter at @computingchamps. **(HW)**

Disclaimer: The information outlined in this article does not reflect the official views of Ofsted, Computing at School (CAS), or the authors, is subject to change and should be used for guidance only.

HELPING BUILD THE ENGINEERS OF TOMORROW

Stew Edmondson explains how Electronics Everywhere is helping students develop the computing skills required for future careers

The UK has a great heritage of technological innovation and a world-class electronics sector. However, there is now a shortage of electronics engineers, meaning that there are too few engineers and designers to develop the next generation of products and help produce creative technological solutions needed by society. The UK Electronics Skills Foundation (UKESF) wants to encourage more young people to consider engineering as a career, and to be aware of the opportunities available in the electronics sector, for the benefit of the UK and wider society.

We know that computing is an increasingly popular A-level subject with 10,000 students nationwide sitting the exam in 2018; this was up 23 per cent on 2017. Despite this, there are few hands-on teaching resources available to schools to teach fundamental computer engineering. Therefore, working closely with academics from the University of Southampton's prestigious School of Electronics and Computer Science, the UKESF has developed a Logic and Arithmetic Kit. The kit features hands-on activities with a circuit board that teach core electronics concepts to A-level Computer Science students.

Along with the Music Mixer Kit, aimed at A-level Physics students, the Logic and Arithmetic Kit is part of the Electronics Everywhere project, which aims to show young people how engaging electronics can be and the exciting career opportunities that are available within the sector. Importantly, the project also provides professional development for teachers to
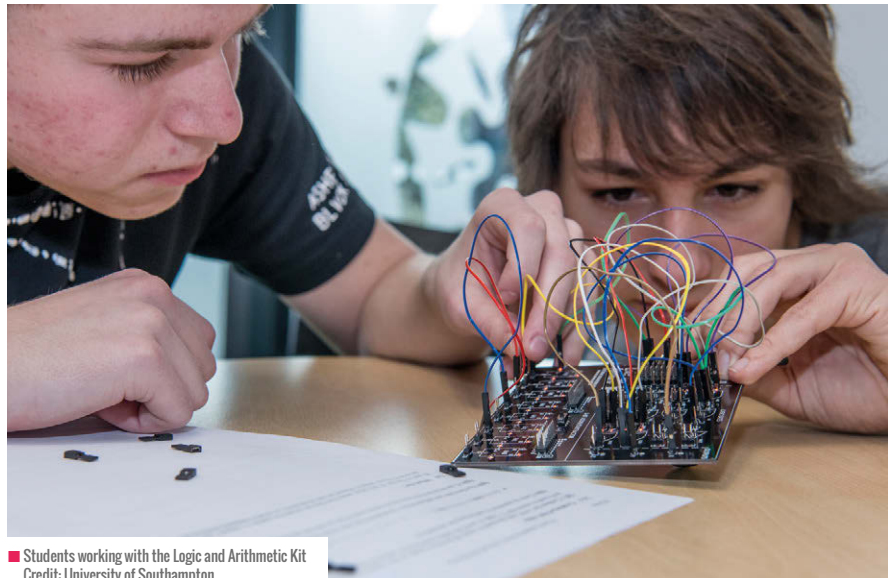
■ Students working with the Logic and Arithmetic Kit
Credit: University of Southampton

increase their knowledge about electronics and so inspire more young people to take up related careers.

Electronics Everywhere has received formal endorsement and support from the Royal Academy of Engineering (RAEng). The UKESF is now working with RAEng to promote the project via their Connecting STEM Teachers programme and using their teacher coordinators to improve the learning resources.

Resources for teachers

The Electronics Everywhere boards have been professionally manufactured and supporting resources, including video tutorials, have been produced. Kits have already been distributed to over 170 schools and one teacher from each school has received face-to-face training at the

STEW EDMONDSON
Stew has been CEO of the UK Electronics Skills Foundation since 2015. He is a Chartered Engineer and a Fellow of the Institute of Engineering and Technology. He has a degree in Electronics (**@theUKESF**).

UNIVERSITY OF SOUTHAMPTON SUPPORTS ELECTRONICS EVERYWHERE

Encouraging A-level students in computing and electronics can be a challenge. When young people visit our university on open days, we really need to pull it out of the bag to ensure they have the best possible experience. Programmes of events can consist of a sample lecture, a practical session, a tour of the campus and a curriculum targeted activity. As an electronics and computer science department we were concerned about the lack of electronics and computing outreach and resources aimed at A-level pupils. This is especially as there is now a shortage of electronics engineers (especially graduates), meaning that there are too few engineers and designers to develop the next generation of products and help produce creative technological solutions needed by society.

The UK Electronics Skills Foundation (UKESF) and the Electronics and Computer Science Outreach Department at the University of Southampton want to encourage more young

people to consider engineering as a career and to be aware of the opportunities available in the electronics sector for the benefit of the UK and wider society.

One fantastic aspect of the project is that it provides outreach in the form of professional development to teachers. It empowers them with the resources to teach aspects of A-level Computing they may have not enjoyed doing previously.

- Reena Pau, University of Southampton
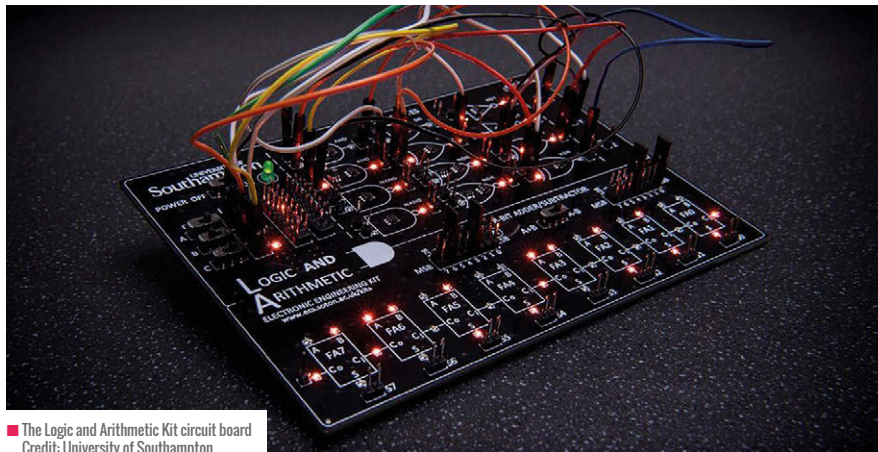
The Logic and Arithmetic Kit circuit board
Credit: University of Southampton

▶ University of Southampton.

The Logic and Arithmetic Kit's bare printed circuit board exposes all the components, creating a link between electronics and applications in many consumer products. This allows students to pursue their curiosity, showing that electronics is not a mysterious black box. The kit is reusable and covers aspects of Boolean operations, logic gates, and base 2 (binary) number systems. These topics are most often taught through simulation or PowerPoint. The board is split into two sections: Logic and Arithmetic.

The Logic section can be used to explore Boolean operations, truth tables, and logic gates, and to implement simple logic functions and circuits. The board has three switchable logic inputs, a range of different logic gates (AND, OR, NAND, NOR, XOR, NOT), and three logic outputs. Orange LEDs indicate the state (logic 0 or 1) of the individual inputs and outputs of every gate.

The Arithmetic section provides an 8-bit two's complement circuit for adding/subtracting, offering the ability to perform A+B or A-B (where A and B are 8-bit binary numbers). This can be used to experiment with unsigned and signed binary arithmetic, as well as offering a different way to observe and understand binary number systems. As an extension, a 9-bit adder circuit can be created, by combining both section of the circuit board.

The kits have been used by over 700 students so far; 70 per cent reacted 'very positively' and were more enthusiastic about electronics as a result. Of the teachers using the resources, 57 per cent said they were an 'excellent' teaching aid. Comments included, 'The kits made a real difference to the engagement of the students and a quite tricky topic was made much more accessible'; 'Students have a much better appreciation of data representation, memory, storage and decision making as a result of using these'; with further comments noting

" ELECTRONICS EVERYWHERE SUPPORTS TEACHERS AND SCHOOLS

that 'They are great bits of kit that bring alive the concepts of logic'.

How to get involved

Electronics Everywhere is a high-impact project. We are investing in and supporting teachers and schools at a local level right across the UK. The project provides reusable classroom resources and trains teachers in their use. This allows teachers to deliver the A-level curriculum for computing in an engaging and interesting way and so improves the learning experience for their pupils. Each school involved in the project:

■ Receives a classroom set of circuit board kits; these resources are fully packaged and are reusable

■ A Computer Science teacher running a session with the Logic and Arithmetic Kit Credit: University of Southampton.

- Has a teacher complete a comprehensive training session to familiarise themselves with all aspects of the teaching resource (a video tutorial is available for teachers unable to travel to Southampton)
- Gains access to online teaching resources, lesson plans, guides, and additional information
- Is covered by a comprehensive support package, including a replacement/repair service.

The ultimate aim of Electronics Everywhere is for every school and college in the UK which offers A-level Physics and/or Computer Science to have a classroom set of the kit, supplied free of charge to those in the state sector. If you would like to get involved, please visit ecs.soton.ac.uk/outreach/kits where you can sign up to the mailing list or email Southampton directly with specific queries about the project.

Electronics Everywhere is also linked to a broader campaign from the UKESF called #TurnOnToElectronics. This multiplatform campaign is fronted by *The Gadget Show* presenter Georgie Barrat and aims to encourage more young people to study and pursue careers in electronics. Georgie said, "I'm delighted to be involved in the UKESF's #TurnOnToElectronics campaign, which is shining a welcome spotlight on careers in Electronic Engineering… studying electronics can lead to an exciting and creative career, which enables students to make a real difference in the world."

To find out more, visit turnontoelectronics.org, where you can watch the campaign video, download the manifesto, and access a wealth of information targeted to young people — from what it is like to be an electronics engineer in the UK to where to go to develop an interest in electronics. Follow @TOTElectronics to keep up to date with the campaign. (HW)

UK ELECTRONICS SKILLS FOUNDATION

The UKESF's mission is to encourage more young people to study electronics and to pursue careers in the sector. The UK electronics sector is big, valuable, and growing; however, the demand for capable, employable graduates is currently outstripping supply. The UKESF is a charity and operates collaboratively with major companies, leading universities, and other organisations to tackle the skills shortage in the sector.

UK Electronics Skills Foundation

CRACKING IDEAS: A COMPETITION FOR YOUNG INVENTORS

Bruce Robinson shares how the Intellectual Property Office and Aardman Productions have collaborated to encourage creative ingenuity
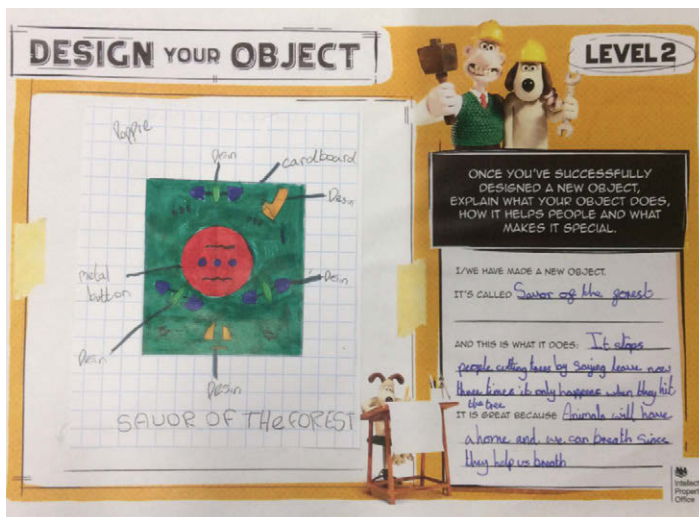
Solutions to the world's trickiest problems sometimes result from years of careful research and product development, sometimes they rise out of looking at a failed project in a new way, and occasionally they just evolve from a germ of an idea following a discussion of a far-reaching idea that's been bounced around between a team.

Young people see the world around them without boundaries. This creativity can be a powerful tool for invention, because young minds are often willing to push the limits, simply because they don't recognise the limits; they haven't learnt them yet.

Creativity and intellectual property

The Intellectual Property Office (IPO) and Aardman – the animation company behind Wallace and Gromit – have launched a new competition that encourages children aged five to eleven to harness their creative ideas. To enter, young inventors need to design something that will make life better. There are resources to support teachers in delivering this competition with their students, as well as activities for students up to the age of 18 and beyond.

A Cracking Ideas project can be big or small and can encompass any area of creative STEM; it is up to the students and their teachers what the project is and where it is going. It could be used to develop a **Big Bang Exhibit** or an entry for a **Crest Award**, or be part of the **Children's University Passport** scheme.

The resources encourage students to learn and research as they go, discovering as they advance their ideas. They learn to appreciate real-world limitations and push boundaries with their imaginations.

Cracking Ideas materials are designed to help learners develop the design process, iteratively improving through prototyping, and then produce their invention, whether as a clay model, an engineering illustration, or a working piece of electronic wizardry. Nothing is out of bounds and nothing is wrong, because what is produced comes straight out of the minds of our young inventors.

In the final stage of the process, the children take ownership of their idea or invention by learning about the process of protecting their new intellectual property: this includes patents, trademarks, and copyright. Children learn the value of their original ideas and the inventions they create, how to protect them from exploitation, and how to eventually convert their creations into a commercial product.

The demand for inspired ideas and inventions isn't going to change anytime soon. The Cracking Ideas competition gives children the opportunity to exercise their creative muscles and forge the future devices they are going to live with one day, and shows how it is possible to make money out of their ideas, too.

The competition is free to enter, and the resources are available for free after registering at **crackingideas.com**. The winner's invention will star in a brand new Wallace and Gromit digital project in 2020. They will also win an Aardman model-making workshop for their school or group. The competition closes on Friday 24 April 2020. **(HW)**

BRUCE ROBINSON
Bruce is a STEM presenter with Wonder Workshops. He is a retired teacher and a professional development provider with STEM Learning and Science Learning Partnership (**@wonderworkshops2**).

■ Many Schoolhouses have seen an increase in girls opting to take computing-related subjects since joining the programme

DIGITAL SCHOOLHOUSE: COMPUTING THROUGH PLAY

Estelle Ashman explains how hosting a programme of workshops and CPD to inspire local primary students in creative tech boosted her own confidence and knowledge

As Digital Schoolhouse's Curriculum Content Developer, Estelle Ashman is responsible for the creation and curation of their innovative teaching resources. With over ten years' teaching experience and a master's degree in Teaching and Learning, Estelle is an enthusiastic practitioner who has the expertise to take difficult computer science concepts and present them in a new and innovative way. Estelle's role with Digital Schoolhouse is hybrid. Her working week is shared between Digital Schoolhouse and classroom teaching at Gildredge House in Eastbourne.

It was almost five years ago when I submitted an application for my school to join Digital Schoolhouse and what a brilliant five years it has been. At the time, I wasn't completely sure what I was joining.

My first experience of a Digital Schoolhouse training event was an innovation – finally, here I was with like-minded individuals from all over the country being trained by some of the best people in computing pedagogy and I was being told that no idea was too crazy! Not only that, but I was being paid to be there – Schoolhouses can apply for a bursary and travel for all training is reimbursed.

With a catalogue of workshops that include everything from lessons on algorithms using Ubisoft's Just Dance to investigating binary numbers with Nintendo's Super Mario Maker 2, Digital Schoolhouse is at the forefront of playful computing.

MACHINE CODE MARIO

Digital Schoolhouse is known for its free, innovative, playful workshops. In this example, students are taught how to create Super Mario levels that require the player to carry out binary conversions and calculations in order to complete the level.

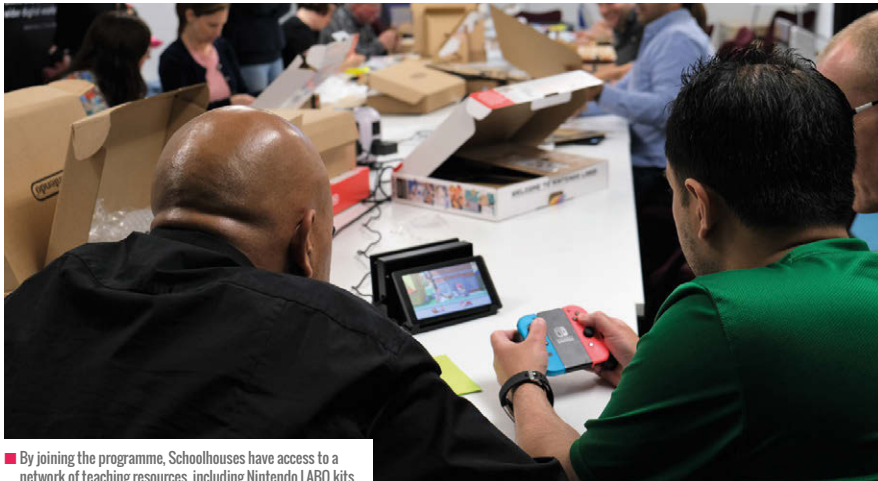Download the workshop resources here: **helloworld.cc/mario**.
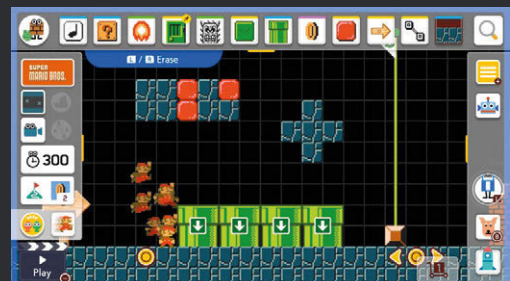
HOW DIGITAL SCHOOLHOUSE WORKS

The not-for-profit Digital Schoolhouse programme, together with Nintendo UK, uses playful learning to engage the next generation of pupils and teachers with the computing curriculum. Digital Schoolhouse is delivered by UK games industry trade body Ukie and supported by the Department of Digital, Culture, Media, and Sport (DCMS).

The process works like this:

■ Education settings (typically secondary schools and FE colleges) are selected to become Digital Schoolhouses.

■ They offer free weekly workshops to visiting primary school pupils and teachers.

■ The aim of each workshop is to teach pupils computing in a way that is creative, innovative, inspirational, engaging, and fun.

■ Computer science concepts are taught using play-based learning. They fuse art and science through dance, games, magic, and storytelling.

■ Each digital schoolhouse lead teacher is given bespoke training to improve their subject knowledge and teaching pedagogy.

■ All visiting teachers are provided with free and personalised support following the visit. This helps them continue to embed computing in their schools.

■ Digital Schoolhouse prides itself on providing high-quality training led by some of the top practitioners in the country

■ By joining the programme, Schoolhouses have access to a network of teaching resources, including Nintendo LABO kits

▶ Rewarding relationships

Working with local primary schools is so rewarding, from encouraging students to work in groups, which I am often told doesn't happen as often as it should, to engaging disengaged students, and every session is different. Each workshop has one thing in common: innovative resources that bring the curriculum to life through playful learning. With our involvement stretching over five years we are now seeing the positive impact that the workshops have on both the students that go on to study with us and the development of our own teaching resources.

I have Lea Gilbert, the previous Head of Gildredge House, to thank for encouraging me to join Digital Schoolhouse. Not only did her suggestion have a positive impact on my sense of worth — it is hugely rewarding to know that someone is noticing your hard work — but it also opened our school to even more opportunities. Over the last few years our students have had the chance to run their own national e-sports tournament, visit top game studios, take part in competitions and I have had the chance to talk about the programme at events and work with partners at trade shows. In the last year, I have had the further opportunity of becoming part of the team and now split my time between working for Digital Schoolhouse and teaching at school.

Join the programme

Each Digital Schoolhouse is based in a school, college, or university environment, and aims to work with a growing network of local primary and secondary teachers to deliver creative and cross-curricular computing lessons using playful learning. Through this model, the programme supports the computing programme of study for the national curriculum in a way that leaves pupils and teachers feeling inspired about, and engaged with, computing and the wider creative digital industries. By joining the programme as a Digital Schoolhouse, it's also possible to improve the transition from primary to secondary school, and secondary teachers can have a better idea of the new intake while raising the profile of computing within their schools and local communities.

Applications are now open for schools wanting to join the programme. For more information visit **digitalschoolhouse.org. uk/join-dsh**. (HW)
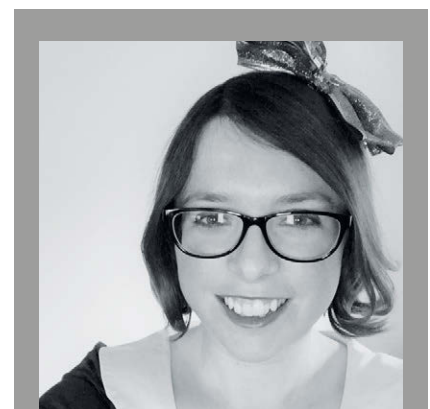
ESTELLE ASHMAN
Estelle Ashman is a teacher at Gildredge House, Eastbourne, and Digital Schoolhouse's Curriculum Content Developer, responsible for the creation of their teaching resources (@CompSci_Geek).

LEARNING LANGUAGES WITH SCRATCH

Scratch and other programming languages are handy educational tools for learning languages

As the Spanish and Computing coordinator at my school, the Premier Academy in Bletchley, I have been trying to use programming activities to improve my students' Spanish skills. Like other teachers, I have been very impressed with the capabilities of Scratch, a block-based programming language and online community. With Scratch, children can program and share interactive media such as stories, games, and animations; it offers students opportunities for deeper learning and understanding of their work in different areas of the curriculum, such as Maths, Art, Science, History, and Modern Foreign Languages (MFL).

Why combine programming with MFL?

By combining programming with Spanish, the children are forced to think creatively and systematically; they have to apply logic to their actions, which helps develop their reasoning skills. They work in pairs and small groups, which encourages collaboration, an essential skill for everyone in today's society. As the learners need to record words and sentences to add to their projects, they improve their speaking and listening skills. Recording themselves and listening back as many times as they like can help to identify errors in pronunciation and boost their confidence. Here are a few of the ways I've incorporated programming into my Spanish lessons.

Animated illustrations

As MFL teachers, we can use Scratch to improve children's vocabulary skills. They can design animated illustrations about the different topics we have previously worked on in our lessons, such as body parts,

JESUALDO MARTÍNEZ
is an Assistant Head Teacher and Computing and Spanish Coordinator at the Premier Academy in Bletchley, home of the codebreakers. He is the author of the blog 'Teaching Languages with Technology' (**helloworld.cc/tech**).

> WITH SCRATCH, CHILDREN CAN PROGRAM STORIES, GAMES, AND ANIMATIONS

classroom objects, or food. Once they have prepared the images, they add text to the different objects so that they can practise their spelling skills; finally, they record their voices, which also helps them to improve their pronunciation skills. In this way, my students have designed cookbooks and interactive body part illustrations.

Interactive games

Another strategy is to ask the children to design interactive games that use the vocabulary they are learning in the lessons, for example making games in which they click on the different words they are listening to, or quiz games in which they have to answer real questions about the topics they have been working on. Something that is a little more difficult, but extremely rewarding, is to create games in which the player clicks on words in the correct order to make a sentence. If you
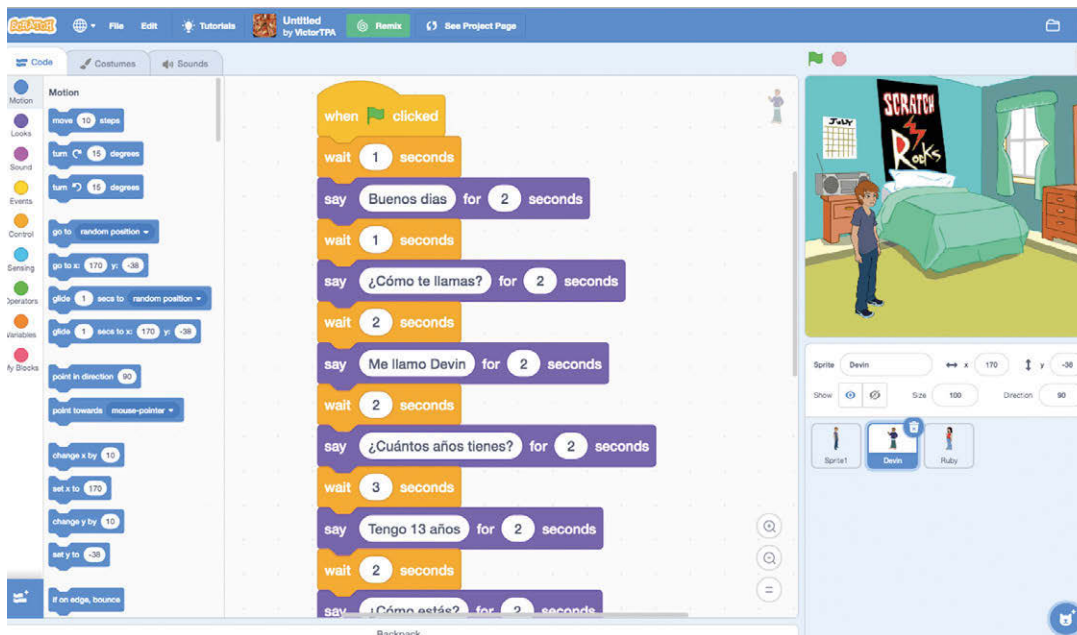
The end result

Incorporating these programming activities into my Spanish lessons has been very motivating and positive, not only in developing digital literacy, but also in improving students' language skills. This has been reinforced by research. Studies have shown that the use of programming activities can be a good way of improving both academic results and student motivation.

As the Computing and Spanish Coordinator at my school, I always try to use technology as an integral part of my language lessons. Scratch is just one of the many tools that educators can use to make lessons more motivating, collaborative, exciting, and engaging for our students and their different learning styles.

Scratch is free, and with a teacher account we can create accounts for groups of students and manage their projects. The Scratch website has many resources, tutorials, and ideas to try in various areas of the curriculum. (HW)

click the wrong word it turns red, while it turns green when it is correct. Another great game is 'Match Pairs', which can be played with pictures, words, or both and involves exactly what the name suggests. This is a great way for children to learn new vocabulary through play.

Animations and interactive stories

My students love to code their own animations and interactive stories, and this contributes greatly to improving their grammar and communication skills. It promotes creativity, collaboration, and engagement. This allows them to practise writing skills as they prepare a storyboard, draft the script, edit, and rewrite it.

Without realising, they are learning by themselves by researching information in their books and dictionaries. They can choose any character they like, switch backdrops, make their characters have a conversation, move the characters, and change an element in the story when they click on it. It's guaranteed to be fun!

Once their projects are finished, the students share them with their partners so that everyone in the class can enjoy and benefit from each other's work.

FURTHER READING

'Computer Programming as an Educational Tool in the English Classroom: A Preliminary Study', 2015, Global Engineering Education Conference, Jesús Moreno-León and Gregorio Robles (**helloworld.cc/madrid**).

'Using Scratch to Teach and Learn English as a Foreign Language in Elementary School', 2016, Sandra Costa, Anabela Gomes, and Teresa Pessoa, International Journal of Education and Learning Systems (**helloworld.cc/scratch**).
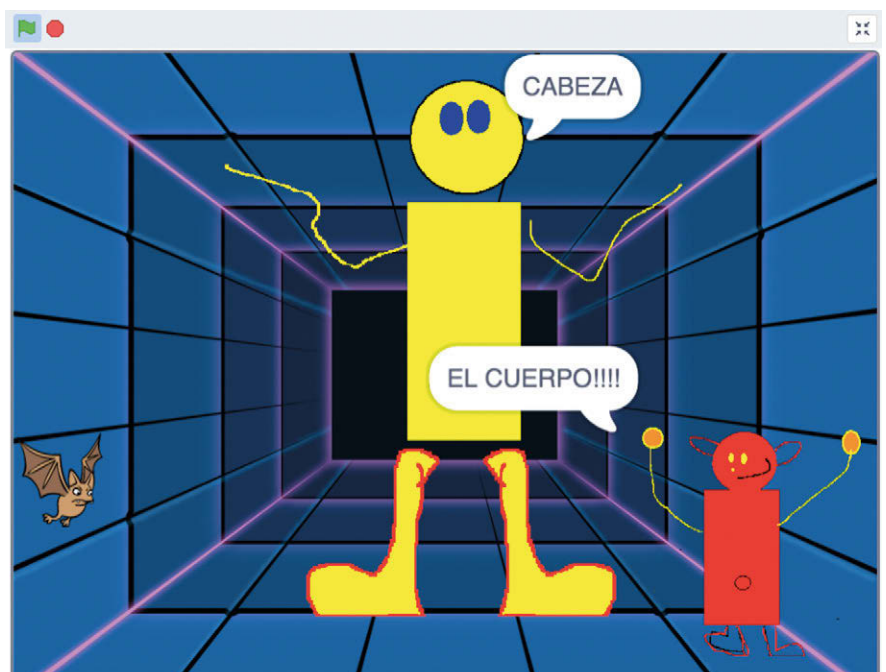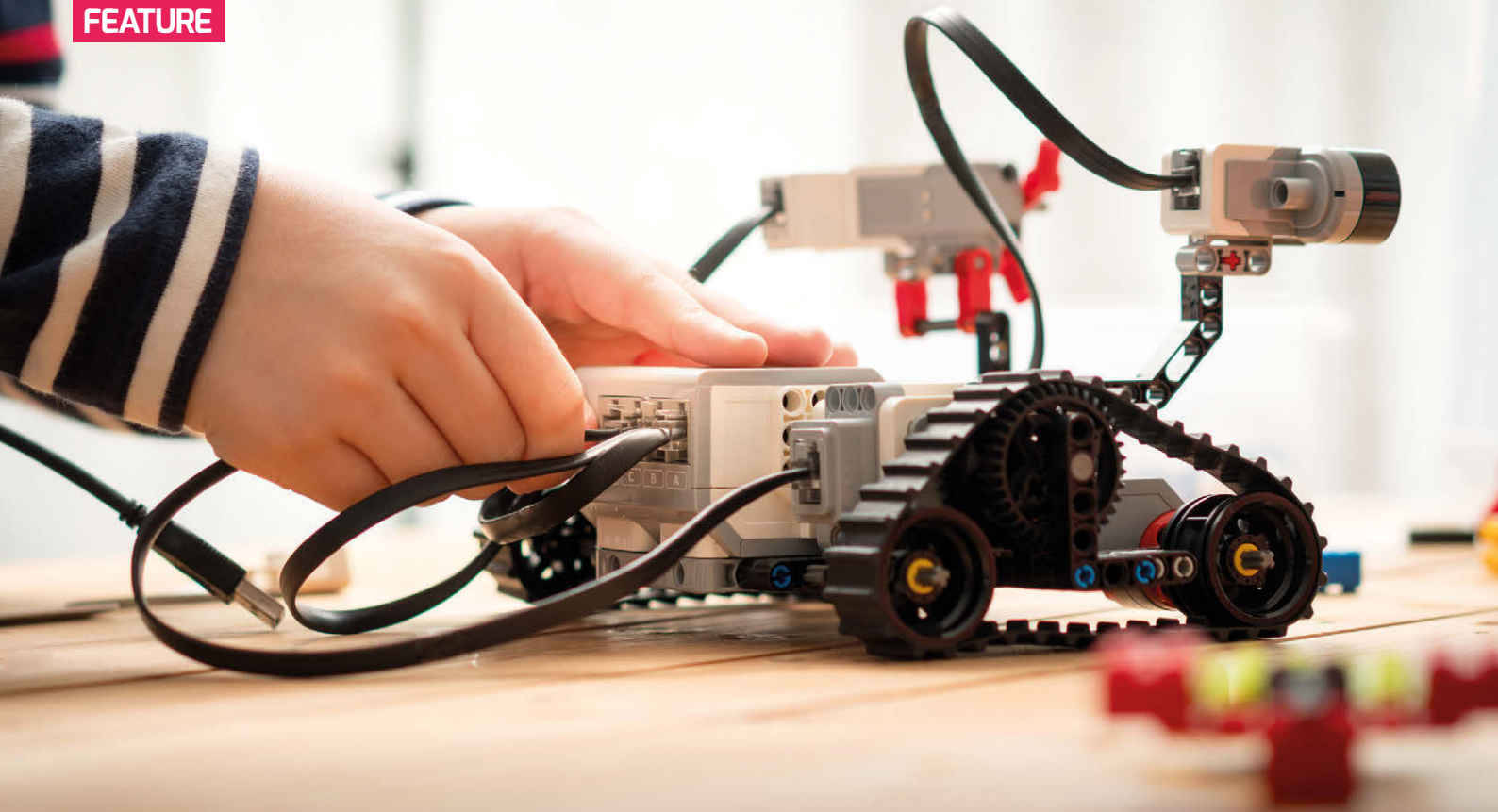
■ Using robots in the classroom helps children engage with coding from a young age

CODING AND 21ST-CENTURY SKILLS

Ursula Martin shows how to include coding within a learning framework that equips students for an uncharted future

How should schools prepare students for lives in a digital society with a dizzying pace of change? Since the 1980s educators, governments, charities, NGOs, and companies have been trying to answer this question. The 21st-century learning framework, conceived in the US but applied in numerous countries, attempts to equip students with the skills they will need. It moves away from rote learning towards an engagement with higher-order skills.

The four Cs of the learning framework are often used by educators in the US to guide their teaching. They are collaboration, communication, creativity, and critical thinking. As coding becomes integral to how students use technology in the classroom, integrating coding into the four Cs of the 21st-century learning framework takes education to the next

level and helps to prepare students for digital careers. This preparation cannot begin too early: it should begin in elementary school and continue to develop as students move on into high school.

Coding and critical thinking

Critical thinking is a skill that students need to develop in the classroom to help them to dispel misconceptions about what they are learning. This also helps students to ask questions that will lead them to a better understanding of what they are learning. Using coding to teach critical thinking skills helps students to solve the problems they identify. For example, there is a problem with lights being left on in the classroom when the room is empty. Students could code an Arduino to turn the lights off if no motion is detected in

the room. This would require students using their critical thinking skills to determine how to solve this problem.

Coding and collaboration

Collaboration is simply working with others to accomplish a specific goal. This may not seem too hard, but it is a skill that many students lack and truly need as they will eventually become a part of a society that functions best through collaborating with each other. Any student can learn to code, but having students work together when coding opens a door to developing better collaboration skills as well as enhancing their skills in the core subject areas. For example, students could increase their maths skills by learning to code Dash and Dot robots to travel on a specific path. The path could be designed
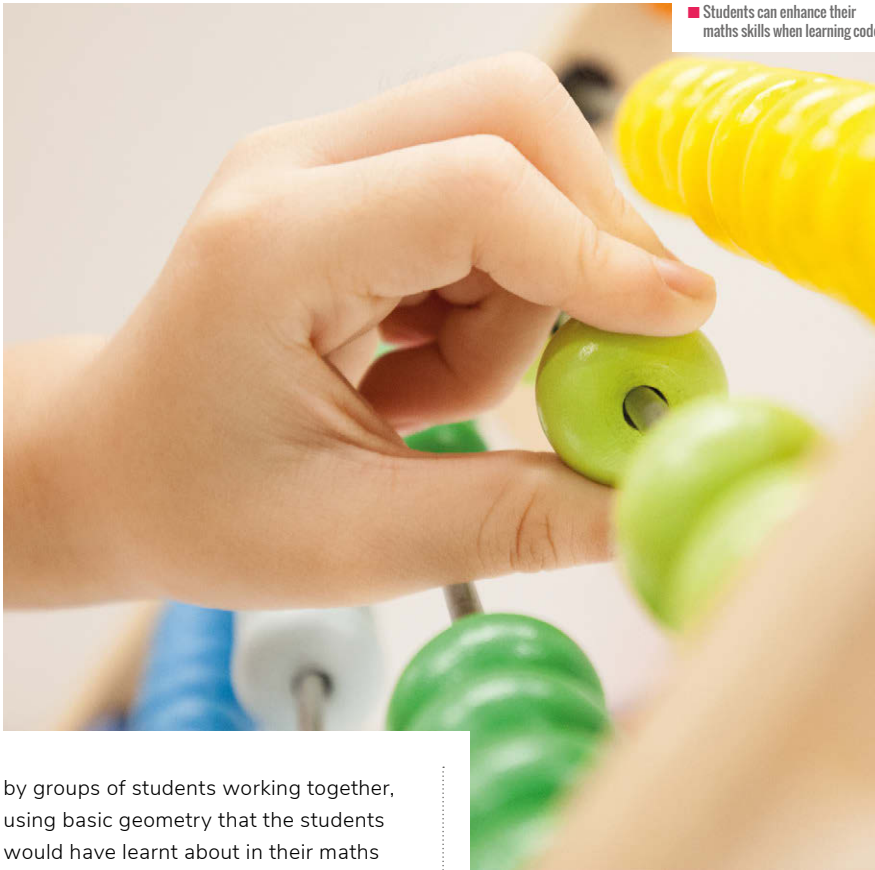
by groups of students working together, using basic geometry that the students would have learnt about in their maths classes. Having students collaborate to accomplish this task helps them to understand that everyone's thought process is not the same and that different opinions can help to create a great project and provide a richer outcome than one person working alone.

learnt. For example, if students are tasked with coding a drone to fly, they must be able to work with others to communicate exactly what they desire their drone to do (fly, flip, and so on) and how long they want that action to take

Coding and creativity

Creativity, as part of the learning process, allows students to learn by thinking outside the box. This helps students to take ownership of their learning in a non-traditional way. When coding is added to this part of the learning process, it introduces an opportunity for students to use different types of technology to 'create' what they are learning. For example, as students study biogeochemical cycles, they learn about the carbon cycle and how it interchanges with oxygen. A project on this could include building a small greenhouse, growing a plant in it and coding a Raspberry Pi to open and close the windows of the greenhouse as the oxygen and carbon dioxide levels reach certain levels. This level of creativity would help students to understand just how these biogeochemical cycles work.

A different way to learn

Framing educational activities using the four Cs of 21st-century learning challenges students to learn in a non-traditional way. Adding coding to this process changes the dynamics of learning altogether. Giving students the opportunity to take ownership of their learning will help teachers to become facilitators and ensure that students better understand how learning works. (HW)

> **" INTEGRATING CODING INTO THE LEARNING FRAMEWORK TAKES EDUCATION TO THE NEXT LEVEL AND HELPS TO PREPARE STUDENTS FOR DIGITAL CAREERS**

Coding and communication

Communication is key in education and in the workforce. Students must learn good communication skills to be effective in every aspect of their daily routine, whether that be in the classroom or in their working lives. Coding introduces a different type of communication ability into the learning process. As students learn to code, they learn to communicate by speaking and writing what they want to create, or what they have previously

place. This means using apps such as the educational programming platform Tynker, to figure out how to accomplish this task. In this way, students not only learn a new 'language' but they also learn how to communicate what they have learned by having the drone perform the tasks they coded it to do. Communication is such an important skill for students to develop and effectively use, and acquiring coding knowledge can play a major part in learning this skill.

URSULA MARTIN
Ursula Martin has been an educator for 17 years. During that time, she has taught Biology, Anatomy and Physiology, and Environmental Science. She is a Raspberry Pi Certified Educator and District Level Technology Resource Teacher in Mobile, Alabama

EIRINI KOLAITI LEARNING MANAGER AT THE RASPBERRY PI FOUNDATION

······································

IN DEFENCE OF PSEUDOCODE

Eirini Kolaiti explores whether describing code in simple English can support the teaching and assessment of programming skills

When I used to teach programming it didn't take me long to realise how easily my students could confuse or even forget basic structural elements of a programming language. They would be tripped up by misusing capitalisation, white space, or parentheses. Basic syntax errors like these would sidetrack students from focusing on developing a solution. Instead, they would resort to trying out every possible instruction they could find online in the hope that it would solve their errors. It turns out that my experiences weren't unique, and are also supported by research in computing education.

The case for simple languages

In 2006, Linda Mannila and colleagues from the Turku Centre for Computer Science in Finland compared 60 programs written by students aged 16–19 after their first programming course, in either Python (a simple language, developed for its readability) or Java (a more advanced language). They analysed the assignments in terms of both syntax and logic errors, and overall functionality.

The results were remarkable. Programming in Python not only helped students avoid making syntax errors, but it also allowed them to solve the given problem. The percentage of Python programs that ran correctly and fulfilled the intended purpose was more than double the Java programs.

One potential drawback of teaching programming in Python rather than a more complex language, such as Java, is that it does the students a disservice – that they would need to relearn aspects of programming when moving to a less intuitive language, like Java. However, the team did not find this to be the case: students who had first learned with Python were at no disadvantage when switching to Java.

So, if choosing a simple language can play such a positive role in student progression, why not opt for that option?

From simple to pseudocode

Allison Elliot Tew and Mark Guzdial took the matter of programming language choice even further. In 2011, they developed a way of comparing the knowledge of university students who took introductory programming
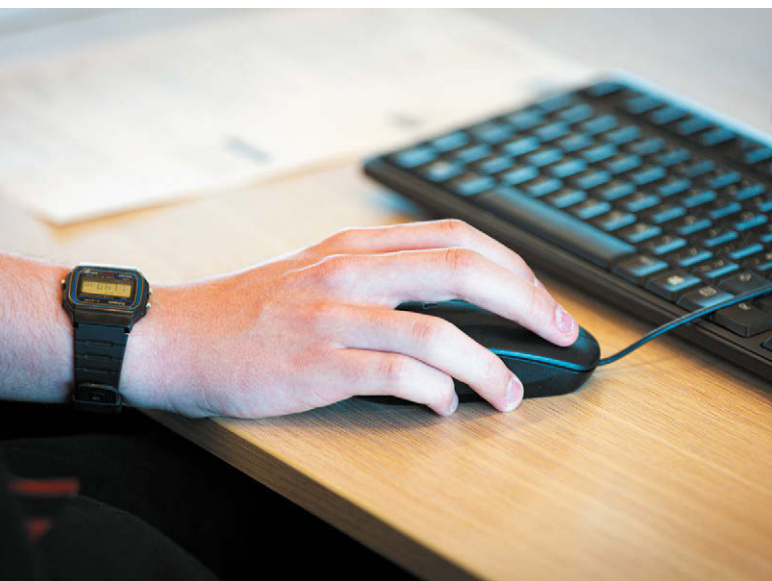
courses in Java, Matlab, and Python. They wanted to make the assessment language-independent, so they used pseudocode instead of any of the taught programming languages. Pseudocode makes use of simple English to describe what a program does. It is laid out in a similar manner to a programming language, but removes some of

> ❝ A PSEUDOCODE-BASED ASSESSMENT CAN ACCURATELY DETERMINE STUDENTS' COMPETENCY

the clutter that is needed for the machine to understand the code, which increases the complexity to a human reader.

The results demonstrated that a pseudocode-based assessment can accurately determine the students' competency regardless of their programming background. This means that students could transfer their comprehension of fundamental programming concepts to pseudocode notation. Reversing this logic, surely we can use pseudocode to scaffold the learning of programming concepts.

Walk before you run (a program)

In 2004, an international group of researchers, led by Raymond Lister at the University of Technology in Sydney, conducted a study regarding programming competency across seven countries. Instead of asking students to produce their own programs, the researchers examined

PSEUDOCODE AT GCSE

REBECCA FRANKS
Learning Manager at the Raspberry Pi Foundation

I recently asked GCSE Computer Science teachers on Twitter which type of pseudocode they used for designing programs. 38% followed the exam board specific pseudocode and 53% did not specify a syntax. Several commented that it was much more important that learners could effectively design their own programs and the syntax used was less relevant.

> Rebecca Franks
> @FranksberryPi
>
> In GCSE Computer Science lessons do you get your learners to design programs using #caschat
>
> | Exam board pseudocode | 37.8% |
> | Any pseudocode | 53.3% |
> | Something else (comment) | 8.9% |
>
> 45 votes · Final results
>
> 11:29 AM · Feb 13, 2020 · Twitter for iPhone
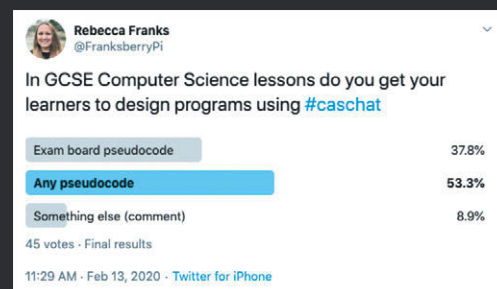
There is currently no requirement for students to write using a specific pseudocode syntax. However, most exam boards have their own unique approach to pseudocode that they use for presenting questions. Students will need to be familiar with these to answer the questions successfully. Exam questions will indicate the form or response required, whether it's a specific programming language, natural English, pseudocode, or flowchart.

Linear search in pseudocode:

```
function linear_search(list, element)
    for i = 0 to len(list) - 1
        x = list[i]
        if x == element then
            return TRUE
        endif
    next i
    return FALSE
endfunction

v = ['Bob', 'Doug', 'Alice']
value = input("Enter search string or q to quit: ")
WHILE value != 'q' AND value != 'Q'
    print( linear_search(v, value) )
    value = input("Enter search string or q to quit: ")
ENDWHILE
```

Linear search in Python:

```
def linearSearch(list, element):
    for x in list:
        if x == element:
            return True
    return False
v = ['Bob', 'Doug', 'Alice']
while True:
    value =input("Enter search string or q to quit: ")
    if value.lower( ) == 'q':
        break
    else:
        print linearSearch(v, value)
```

> whether students could understand existing code (written in Java or C++), for example, by predicting the output of a given program. The results suggest that what stops many students from performing well in programming tasks is not a lack of ability to problem-solve, but a fragile knowledge of fundamental concepts. Students were unable to hand-trace code (where the values of variables are calculated by hand) because of an inefficient command of basic programming tasks, such as iterating over an array or the use of recursion. These areas are mostly related to an ability to read, rather than write, code.

I would therefore argue that as educators we should use programs in pseudocode to foster these preliminary skills, so that students benefit from a reduction in the cognitive load caused by language-specific syntax. After all, most programming languages are not designed with the aim of teaching programming in mind, whereas pseudocode can be adapted to meet the needs of the students.

Pseudocode activities can be used in lessons in order to practise reading and tracing code, and as an opportunity to discuss basic programming concepts. Students could

PSEUDOCODE ACTIVITIES

Here are some ideas for ways to incorporate pseudocode into your teaching:

- Start activities with pseudocode to discuss concepts and then convert into code to deal with syntax errors, and to test/debug the algorithm and check the structure of the algorithm. It should be easier to write an algorithm after students have worked on a pseudocode version, rather starting with a programming language
- Ask students to write a program using pseudocode and then swap with a partner for writing it in a programming language
- Give snippets of pseudocode to test basic misconceptions, for example in the use of recursion
- Give small programs that students can hand-trace, writing out the values of variables as the program progresses, to check their skills at understanding code

Linear search in Java:

```java
import java.util.Vector;
    public class JavaEx {
        public static boolean linearSearch(Vector v, Object o) {
            for (int i=0; i 5 v.size( ); i++) {
                if (v.elementAt(i).equals(o)) {
                    return true;
                }
            }
            return false;
    }

    public static void main(String args[ ]){
        Vector v = new Vector( );
        BufferedReader in = new BufferedReader(
            new InputStreamReader (Systems.in)
            );
        String[ ] values = {"Bob", "Doug", "Alice"};
        for (int i = 0; i 5 values.length; i++) {
            v.addElement(values[i]);
      }
        String value;
        while (true) {
            System.out.println("Enter search string or" + " q to quit: ");
        value = in.readLine( );
        if (value.toLowerCase( ).equals("q")) {
            break;
        }
        else {
            System.out.println(linearSearch(v, value));
        }
        }
    }
}
```

then convert the pseudocode programs (their own or each other's) into compilable code. This gives them the opportunity to experiment with how to implement algorithmic constructs using the specific features of a programming language, and how to deal with syntax errors. Having the stepping stone of pseudocode therefore helps with the skills needed to test and debug algorithms,

Pseudocode can also be used for formative assessment. Low-stakes tests and starter activities that use pseudocode snippets to test specific misconceptions can help unpack the underlying processes of program execution. I have found that challenging students with small, targeted pseudocode programs helps with engagement and information retention. In this way, pseudocode could pave the way towards gaining fundamental knowledge and skills through reading and tracing code before moving on to writing actual programs and dealing with the inevitable corollary of syntax errors. (HW)

EIRINI KOLAITI
Eirini is a Learning Manager at the Raspberry Pi Foundation. She is an RPi Certified Educator and used to be a Computer Science teacher. She misses her Girls' Computing Club (**@eirinikolaiti**).

FURTHER READING

Mannila et al. (2006): helloworld.cc/simple

Tew, A. E. and Guzdial, M. (2011): helloworld.cc/independent

Lister, R et al. (2004): helloworld.cc/study

CESI•CS: RECENT DEVELOPMENTS IN IRISH SCHOOLS

In October 2019, CESI (the Computers in Education Society of Ireland) reviewed their Community of Practice (CoP) for computing educators

S ince its foundation in 1973, the Computers in Education Society of Ireland (CESI), has flourished as a dynamic community of educators striving to promote the use of computers in Irish education at all levels. It is a voluntary grassroots organisation that draws its members from primary, secondary, and third level educators, as well as other relevant organisations.

CESI•CS and CoP

The CESI•CS CoP focuses on computing education and is a collaboration with education centres within Ireland through the Education Support Centres of Ireland and Trinity College Dublin. CESI•CS began in September 2017 and was financially supported by Google for the first two years.

The aims of the CESI•CS project are to make:

- A sustainable CoP for computer science in Ireland
- Improvements in computer science education
- Bottom-up collaboration at local levels and across sectors
- Teacher readiness

On Saturday 12 October, CESI•CS hosted its second National Symposium to bring together computing educators from across Ireland to discuss the development of computing so far. A total of 50 computing
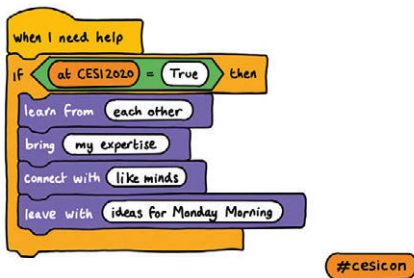
#cesicon

RELEVANT LINKS

- CESI - Computers in Education Society of Ireland - **cesi.ie**
- CESI•CS - CESI community of practice for computing - **cesi.ie/cesi-cs/**
- NCCA - Coding in Primary Schools Initiative - **helloworld.cc/primary**
- JCTCIA - Junior Cycle Short Course Coding in Action - **helloworld.cc/aims**

Driven by Volunteers

■ CESI Logo @visualthinkery, licenced under CC-BY-SA

■ Educators meet to discuss computing in Ireland thanks to CESI's grassroots group

educators took part, representing all sectors, levels and government agencies involved.

The context of the CESI•CS symposium is that in the past few years, significant curriculum development work has occurred for computing in Irish schools:

are keen to respond to the societal and parental pressure to introduce computing, making a rationale for establishing the CoP to prepare the ground.

Since September 2017, CESI•CS has organised 57 regional and five national

> " TEACHERS ARE KEEN TO RESPOND TO THE SOCIETAL AND PARENTAL PRESSURE TO INTRODUCE COMPUTING

- For ages 12–15, there is now a computer science short course module, Junior Cycle Coding in Action.
- At senior cycle level, for ages 16–18, a Leaving Certificate Computer Science qualification equivalent to UK A-level was introduced in 40 phase-1 schools from September 2018. This will open up to a wider number in September 2020.
- From September 2017, computational thinking has been explored at primary level through the Coding in Primary Schools Initiative at the NCCA (National Council for Curriculum and Assessment) which has involved some 40 schools over two years.

But not all schools and teachers can be part of these pilot programmes. Teachers

face-to-face meetings all over Ireland in order to enhance the grassroots organisation of CESI based on trust, respect, and collaboration. In addition, educators could connect nationally by signing up to the CESI mailing list. A facilitator's guide was distributed at the Symposium as a means to encourage and support volunteer educators to take over the organisation and facilitation of meetings.

A major focus of the discussion in Dublin was the need for preparation of the second cohort of computer science teachers, who will be teaching the Leaving Certificate at senior cycle level from September 2020. CESI plans to continue support for them and others at a grassroots, regional, and national level, both face-to-face and online.

CESI 2020 conference

A more recent highlight for Irish computing educators was the annual CESI conference, which attracted interest both nationally and internationally. Impassioned computing educators met on the day to share best practices, innovations and research on all topics related to the use of computers in education. The CESI 2020 conference kicked off with a TeachMeet on Friday 28 February at 7pm in the Athlone Sheraton Hotel and continued all day in Athlone Institute of Technology on Saturday 29 February. The theme of the conference was 'Our evolving learning landscape'. (HW)

FURTHER INFORMATION

For more information about CESI, contact Adrienne Webb, CESI chair - chair@cesi.ie

Written by CESI members
HELEN O'KELLY
DR RICHARD MILLWOOD
ADRIENNE WEBB
KATIE MOLLOY
PAT SEAVER

Learners with more agency will be more engaged in their learning

PROJECT-BASED LEARNING: A PATH TO AGENCY

Mark Calleja investigates project-based learning, a teaching method that gives your students the tools to take charge of their own learning

P roject-based learning is about asking your students to solve a real-world problem by designing and creating a project over a specific period of time. Getting started with project-based learning in your classroom requires a bit of a shift in thinking: instead of providing an activity that demonstrates prior knowledge or acts as a unit assessment, students' learning is directly embedded into and emerges from the investigative and design processes they engage in while solving the problem you've posed. That is to say, their project is the unit: it's the vehicle for teaching the knowledge and skills that students need to learn, as well as the assessment process that demonstrates their learning. I'm a huge proponent of teaching this way, for a whole host of reasons.

Creating solutions from the ground up

Agency: For your learners, there's a massive difference between being asked whether they know the answer to a problem, and being asked to find a solution to one. The first question assumes knowledge (and thereby frames a lack of knowledge as failure) and has a narrow focus, while the second gives learners the room to be wrong or not know yet, and to develop real understanding and practical skills in a self-directed way.

Engagement: Learners with more agency are likely to be more engaged in their learning, too. A huge benefit of project-based learning is the scope it gives you to set relevant, real-world problems for your students. Being able to relate their learning to their own lives motivates students: when they see that they can apply new skills and

knowledge to other situations in their lives, they understand the true purpose of the work they're doing.

Universal skills: Project-based learning doesn't give students ready answers to a specific problem; it asks them to build a mental toolkit for understanding any problem, so that they can create solutions from the ground up. By enabling this in-depth learning, you equip your students for real life, letting them practise skills required in most industries today: taking initiative, working responsibly, decomposing and solving problems, collaborating in teams, and communicating their ideas clearly.

Giving the power back to your learners

When your learners are interested and engaged with their own learning, your job changes from passing on knowledge and

Learners need only teach themselves using the sheets and videos provided to make their ideas real

MARK CALLEJA
Resident trickster at the Raspberry Pi Foundation and amateur gentleman adventurer looking to go pro, Mr C is a teacher, maker, hacker, MacGyver, Raspberry Pi Certified Educator (**@M1st3r_C**).

> ## PROJECT-BASED LEARNING ASKS STUDENTS TO BUILD A TOOLKIT OF SOLUTIONS

RESOURCES TO SUPPORT PROJECT-BASED LEARNING

Guides for how to run hackathons with your students are available at **rpf.io/hackathon-guides** and **rpf.io/component-sheets**.

managing motivation to facilitation and inspiration. In project-based learning, you direct learners towards information instead of handing them answers, and you support them in creating something they didn't know they were capable of.

One way to get started with project-based learning is to use a bank of Python resources that embody the idea, developed with the National Citizen Service. There's a bank of 14 different helpsheets (with accompanying YouTube animations) that provides images of each basic electronic component, a simple wiring diagram with numbered GPIO pins, and the simplest of gpiozero code snippets to execute its basic functions, all on one handy page.

The helpsheets cover the most common simple components used in digital making, from LEDs to infrared motion sensors, cameras, Bluetooth remote controls, and beyond. There are also sheets that explain the Sense HAT's on-board sensors, joystick and LED array, with accompanying code examples. We also made some that cover some commonly used processes in digital making, such as playing sounds with Python, making a remote control with the Blue Dot Bluetooth app, and setting up a Raspberry Pi-based gadget to function automatically as your students intend it to as soon as they power it up (known as 'running headless').

The intention behind the sheets is that you will first support your students through the design discussions they'll need to have before they start making things and show them the library of components they have available. When they know what functionality they want from their invention (and what's possible given their time and hardware constraints), learners need only teach themselves using the sheets and videos to make their ideas real. All of the code is broken into three sections to make each Python script modular; students can simply combine the code snippets on the sheets to make larger scripts that create more complex functionality.

If you'd like to replicate our hackathon model with your own students, we have released the facilitator's guide to running the full two-day experience, complete with session timings, delivery notes, workshop slides, and a student support document called the 'Developer's Guide' in which participants can make notes and get discussion prompts and tips throughout their build process. **(HW)**

Why not run a hackathon with your own students?

USING DEEPER LEARNING AS A FRAMEWORK FOR TRANSFORMATIVE EDUCATIONAL EXPERIENCES

Brent Richardson outlines the case for a pedagogical approach that challenges students

A benefit of completing school and entering the workforce is being able to kiss standardised tests goodbye. That is if you don't count those occasional 'prove you watched the webinar' quizzes some supervisors require.

In the real world, assessments often happen on the fly and are based on each employee's ability to successfully complete tasks and solve problems. It is often obvious to an employer when their staff members are unprepared.

Formal education continues to focus on accountability tools that measure base level proficiencies instead of more complex skills like problem solving and communication.

One of the main reasons the U.S. education system is criticised for its reliance on standardised tests is how this method of assessing a student's comprehension of a subject can hinder their ability to transfer knowledge from an existing situation to a new situation. The effect leaves students ill-prepared for higher education and the workforce.

A study conducted by the National Association of Colleges and Employers found a significant gap between how students felt about their abilities and their employer's observations. In seven out of eight categories, students rated their skills much higher than their prospective employers had.

Some people believe that this gap continues to widen because teaching within the confines of a standardised test encourages teachers to narrow their instruction. The focus becomes preparing students with a limited scope of learning that is beneficial for testing.

With this approach to learning, it is possible that students can excel at test taking and still struggle with applying knowledge in new ways. Educators need to have the support to not only prepare students for tests but also develop ways that will help their students connect to the material in a meaningful manner.

BRENT RICHARDSON
Brent is the Fab Lab Director for the community development organisation BakerRipley in Houston, Texas. He studied Art and Technology at the University of Oklahoma where he used digital fabrication equipment to create interactive gallery exhibitions.

■ Outside of the classroom, assessments are often based more on problem solving than standardised tests

■ The project focused on capturing the building of a new Fab Lab site in Houston

In an effort to boost the U.S. education system's ability to increase the knowledge and skills of students, many private corporations and nonprofits directly support public education. In 2010, the Hewlett Foundation went so far as to develop a framework called 'deeper learning' to help guide its education partners in preparing learners for success.

The principles of deeper learning

Deeper learning focuses on six key competencies:

1. Master core academic content
2. Think critically and solve complex problems
3. Work collaboratively
4. Communicate effectively
5. Learn how to learn
6. Develop academic mindsets

This framework ensures that learners are active participants in their education. Students are immersed in a challenging curriculum that requires them to seek out and acquire new information, apply what

" DEEPER LEARNING ENSURES THAT LEARNERS ARE ACTIVE PARTICIPANTS IN THEIR EDUCATION

they have learned and build upon that to create new knowledge.

While deeper learning experiences are important for all students, research shows that schools which engage students from low-income families and students of colour in deeper learning have stronger academic outcomes, better attendance and behaviour and lower dropout rates. This results in higher graduation rates, and higher rates of college attendance and perseverance than comparison schools serving similar students. This pedagogical approach is one we strive to embed in all our work at Fab Lab Houston.

A deeper learning timelapse project

The importance of deeper learning was undeniable when a group of students I worked with in Houston built a solar

powered time-lapse camera. Through this collaborative project we quickly found ourselves moving beyond classroom pedagogy to a 'hero's journey' — where students' learning paths echo a centuries-old narrative arc in which a protagonist goes on an adventure, makes new friends, encounters roadblocks, overcomes adversity and returns home a changed person.

In this spirit, we challenged the students with a simple objective, 'Make a device to document the construction of Fab Lab Houston'. In just one sentence, participants understood enough to know where the finish line was without being told exactly how to get there. This shift in approach pushed students to ask questions as they attempted to understand constraints and potential approaches.

The project helped to develop the students' project management and engineering skills

> Students shared ideas ranging from drone video to photography robots. Together everyone began to break down these big ideas into smaller parts and better define the project we would tackle together. To my surprise even the students that typically refused to do most things were excited to poke holes in unrealistic ideas. It was decided, among other things, that drones would be too expensive, robots might not be waterproof and time was always a concern.

The decision was made to move forward with the stationary time-lapse camera because although the students didn't know how to accomplish all the aspects of the project, they could at least understand the project enough to break it down into doable parts and develop a ballpark budget. Students formed teams and picked one aspect of the project to tackle. The three subgroups focused on taking photos and converting them to video, developing

STUDENTS COULD UNDERSTAND THE PROJECT ENOUGH TO BREAK IT DOWN INTO DOABLE PARTS AND DEVELOP A BUDGET

a remote power solution and building weatherproof housing.

A group of students found sample code for the Raspberry Pi that could be repurposed to take photos and store them sequentially on a USB drive. After quick success, a few ambitious learners started working to automate the image post processing into video. Eventually, after attempting multiple ways to program the

computer to dynamically turn images into video, one team member discovered a new approach. Since the photos were stored with a sequential numbering system, thousands of photos could be loaded into Adobe Premiere Pro straight off the USB with the 'Automate to Sequence' tool in Premiere.

A great deal of time was spent measuring power consumption and calculating solar panel and battery size. Since the project would be placed on a pole in the middle of a construction site for six months, the students were challenged with making solar-powered time-lapse camera as efficient as possible.

Waking the device after it was put to sleep mode proved to be more difficult than anticipated, so a hardware solution was tested. The Raspberry Pi was programmed to boot up after receiving power, take a picture and shut itself down. With the Raspberry Pi safely shut down, a timer relay cut power for ten minutes before returning power and starting the cycle again.

Finally, a waterproof container had to be built to house the electronics and battery. To avoid overcomplicating the process, the group sourced a plastic

Students had to calculate the likely power consumption and which battery size would be required

A special box was created to house the camera and Raspberry Pi, helping it to withstand stormy weather

weatherproof ammunition storage box to modify. Students operated a 3D printer to create custom parts for the box.

After cutting a hole for the camera, a small piece of glass was attached to a 3D printed hood ensuring no water entered the box. On the rear of the box, they printed a part to hold and seal the cable from the solar panel, where it entered the box. It only took a few sessions before the group produced a functioning prototype. The project was then placed outside for a day to test the capability of the device.

The test appeared successful when the students checked the USB drive. The drive was full of high quality images captured every ten minutes. When the drive was placed back in the Raspberry Pi, a student noticed all the parts inside the case moved.

The high temperature that day of the test melted the glue used to attach everything. This unexpected problem challenged students to research a better alternative and reattach the pieces.

Once the students felt confident in their device's functionality, it was handed over to the construction crew to install the camera on a twenty foot pole. The installation went smoothly and the students anxiously awaited to see the results.

Less than a week after the camera went up, Houston was hit hard with the rains brought on by Hurricane Harvey. The group was nervous to see if the project they had constructed would survive. However, when they saw that their camera had survived and was working, they felt a great sense of pride.

They recognised that it was a collaborative effort of the group to problem solve possible challenges, that allowed their camera to not only survive but to capture a spectacular series of photos showing the impact of the hurricane in the location it was placed. The footage can be seen at: **vimeo.com/238079409**
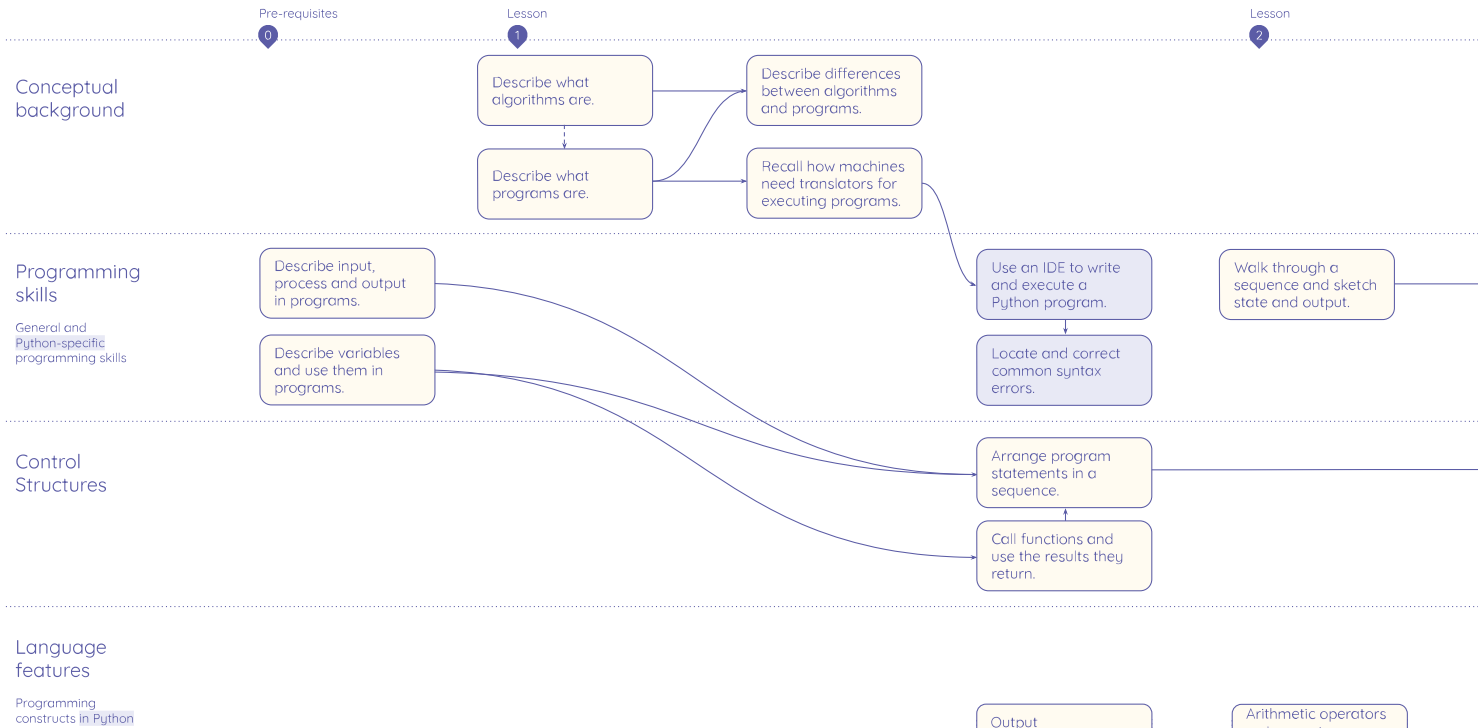
A worthwhile risk

Overcoming many hiccups throughout the project was a great illustration of how the students learned how to learn and to develop an academic mindset. A set back that at the beginning of the project might have seemed insurmountable was laughable in the end.

Throughout my experience as a classroom teacher, a museum educator and now a director of a digital makerspace, I've seen countless students struggle to understand the relevance of learning and this has led me to develop a strong desire to expand the use of deeper learning.

Sometimes it feels like a risk to facilitate learning rather than impart knowledge, but after seeing a student's development into a changed person, ready to help someone else learn, it makes it worth the effort. Let's challenge ourselves as educators to help students acquire knowledge and use it. (HW)

Work on the construction site was documented in images, which were used to create a short video

	Pre-requisites 0	Lesson 1		Lesson 2
Conceptual background		Describe what algorithms are. → Describe what programs are.	Describe differences between algorithms and programs.	
			Recall how machines need translators for executing programs.	
Programming skills General and Python-specific programming skills	Describe input, process and output in programs.		Use an IDE to write and execute a Python program.	Walk through a sequence and sketch state and output.
	Describe variables and use them in programs.		Locate and correct common syntax errors.	
Control Structures			Arrange program statements in a sequence.	
			Call functions and use the results they return.	

Language features
Programming constructs in Python

Output	Arithmetic operators and expressions
`print`	`+, -, *, /, //`
Assignment (literals)	Assignment
`=`	`=`
Input (strings)	Input (numerical)
`input`	`int, float`

LEARNING GRAPHS: TOOLS TO PLAN FOR PROGRESSION

Carrie Anne Philbin introduces an approach to map the computing curriculum

CARRIE ANNE PHILBIN
Carrie Anne is Director of Educator Support, Raspberry Pi Foundation, leading the development of resources to teach computing.

A progression framework is the backbone of any subject curriculum. It illustrates the sequence in which children learn, noting where core understanding of a topic is established in order to progress. As part of the National Centre for Computing Education (NCCE), my team is developing a bank of teaching resources that deliver the computing curriculum in England. In order to successfully do this, we've studied a lot of progression frameworks, examination specifications, and even some research papers. We found that there are two quite different ways of presenting progression that show what should be taught and when it should be taught, as well as information on how or why concepts should be taught.

Firstly, there is the approach of creating a categorisation of skills and concepts into a list or table. Sequencing is shown by having objectives listed by key stage, year group or even by learners' age. Examples of this approach include the CAS computing progression pathways (**helloworld.cc/ path**), and the Massachusetts Digital Literacy and Computer Science Curriculum Framework (**helloworld.cc/standards**). They are essentially a list of required knowledge bundled by theme.

Another approach is to use a map of possible trajectories through learning waypoints—key building blocks of learning—and how they connect to each other. This approach highlights where prerequisite knowledge needs to be

Walk through branches and sketch state and output.

Walk through loops and sketch state and output.

Use variables as counters in iterative programs.

Use boolean variables as flags.

Use binary selection to control the flow of program execution.

Use selection to control the flow of program execution.

Use iteration to control the flow of program execution.

Combine iteration and selection.

Selection (binary)
`if, else`

Selection
`if, elif, else`

Iteration
`while`

Relational operators in logical expressions
`==, !=, <, <=, >, >=`

Logical operators and expressions
`and, or, not`

Randomness and using modules
`randint`

This resource is available online at ncce.io/prg4-lg. Resources are updated regularly — please check that you are using the latest version. This resource is licensed under the Open Government Licence, version 3. For more information on this licence, see ncce.io/ogl.

■ The learning graph for the Year 8 topic of Python programming. Solid arrows show prerequisites while dotted arrows show a suggested order of learning

mastered before students can move on, as well as the dependent knowledge contained in other 'nodes', each containing one part of the computing curriculum that needs to be mastered in order to progress. Cambridge Mathematics (**cambridgemaths.org**) are leading the way in 'developing a flexible and interconnected digital framework to help reimagine mathematics education 3–19'. We've been lucky enough to learn from their work, which has helped us to create learning graphs.

A tool for teachers

The learning graphs organise computing content—concepts, knowledge, skills, and objectives—into interconnected networks. We found that nodes often form clusters corresponding to specific themes and we could connect them if they represent two adjacent waypoints in the learning process. Depending on the context, the nodes in a learning graph could contain anything ranging from the contents of a curriculum

> ## LEARNING GRAPHS DIRECTLY INFORM LESSON PLANNING AND ALSO ADD VALUE BY SHOWING OTHER OPPORTUNITIES

strand across an entire key stage, to the learning objectives of a 6-lesson unit. When our team starts working on a unit, the learning graphs are in a fluid state: they uncover the structure of the content and the possible journeys through it, without being bound to a specific teaching pathway. The graphs eventually reach a fixed state, where the nodes are further structured and arranged to reflect our suggestions on the order that the content could actually be delivered.

We believe that learning graphs could be useful to teachers on a whole new level. They directly inform lesson planning, but also add value by showing opportunities to assess understanding at landmark points in

a lesson or unit. By checking that students are grasping the concepts, teachers are able to think more about how they are teaching. They can revisit knowledge that perhaps didn't land with learners the first time.

All progression frameworks are subjective, and with little research into computing education, we rely on teachers' experience of combining the 'what' we teach and 'how' to teach it to help inform this work. If you've not taken a look at our learning graphs, you can access them via **teachcomputing.org/resources**. Do let us know your thoughts by emailing us at **resourcesfeedback@raspberrypi.org** (HW)

BEHIND THE SCENES AT AN ONLINE COURSE

Michael Conterio reveals some of the work that goes into creating materials for online training courses

It's been a busy year for my team at the Raspberry Pi Foundation—we've launched 14 online courses as part of the National Centre for Computing Education (NCCE). Here's how we've been going about making sure that all of them are high-quality learning experiences.
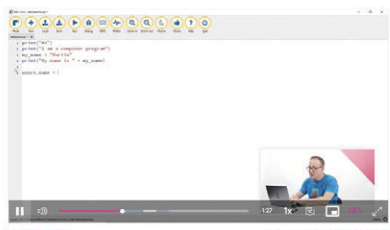
First, make a plan

Although we've launched an average of more than one course a month, the story behind each course starts five or six months before with the planning stage. The most important part of creating a

Courses use screencasts so that the instructors can talk you through writing code.

course comes first: working out what it's about. It sounds silly—surely when you start writing a course you know what it's about—but even when you've got an overarching topic you still need to work out the specifics to include. Sometimes, it's a case of working out the programming concepts you want to cover, the most sensible order to cover them

> ## IT'S A CASE OF WORKING OUT WHAT PROGRAMMING CONCEPTS TO COVER AND THE SENSIBLE ORDER TO COVER THEM IN

in, and some programs that the learners could work towards building. In other cases, the structure has been a lot harder to pin down, for example in our 'Impact of Technology: How to Lead Classroom Discussions' course. In that case, we had to work out how to make sure we interleaved useful advice for running classroom discussions and help for long-answer questions with information to start off discussions on a range of different topics.

Once the overall content has been decided on, we decompose the course down into smaller segments. One person then takes control to plan each course step by step. Dropping down from a group to a

single person here helps cut down the time spent going back and forth on particular elements and can keep the course feeling more coherent. While writing the course step by step, the lead educator tries to make sure that we're not only sharing useful pedagogical approaches, but also implementing them in the courses. We know that teachers' time is limited, so each

step needs to have a good reason to exist. The flow of the course between steps is important too. We need to help keep learners feeling that the material makes

Each course has one person overviewing each step

■ Collaboration is an important part of course design

sense and motivates them to continue. Not getting this right at this stage can lead to trouble. Although we may make minor changes to the plan as we write the course, the few times we've made larger changes, we've had to put in a lot of extra work to make sure the learner's journey through the course still makes sense.

Before the writing can start, the step-by-step plan will be checked by others in the team, particularly those who helped with some of the earlier planning. This can help spot where some elements from the initial planning haven't been followed through in the step by step, as well as point out a few points where the plan can be polished up slightly.

The write stuff

Once the plan is approved, it's time to get down to the writing. This can take

■ Real-world examples, like creating a train departure board, help teachers make programming relevant to students

several months, thanks to a combination of factors, such as the amount of material needed (courses can have over 25,000 words), the fact that our writers are often juggling multiple responsibilities, and also because I can get quite picky. In my role as production manager, I see the initial drafts of each step and generally provide feedback. At this point I'm pretty much acting as an advocate for the course participants, trying to make sure that everything is as clear as it can be for them, whether that's a definition of a term, instructions for activities, or in making sure that learners not only understand what they are doing but also why we are asking them to do it. Sadly, I am fallible, but other people will also get a chance to check the course later and we can also respond to feedback from our learners.

Writing for online courses is pretty hard on our creators, as it's quite different to the writing (and teaching!) they usually do. Although learners will be able to comment and get feedback, real-time communication isn't possible, so our course creators must navigate the tricky task of conveying tone in solely text-based communication, as well

as trying more generally to minimise confusion. They've also got to try and make sure that there's enough of their own passion and voice present to make a connection with each of the potentially thousands of learners per run. It's not an easy job, but they all put in a lot of hard work, and take a lot of feedback with good grace, to make the courses the best that they can be.

▶

MICHAEL CONTERIO

Michael works for the Raspberry Pi Foundation, producing online courses. Previously a physicist, he moved into education and helped develop Isaac Physics. In his spare time he enjoys walking, board gaming and performing stand-up comedy.
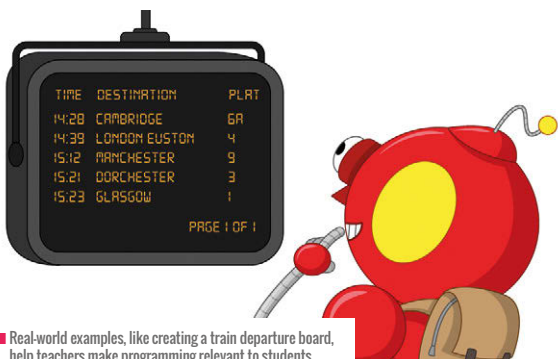
All our teams work together to ensure good high-quality learning materials

ONLINE COURSES FOR COMPUTING EDUCATION

There are over 20 free online courses from the Raspberry Pi Foundation, available to access from anywhere in the world. Whether you're simply curious about computing, a coding club volunteer, a lifelong learner, or an school teacher, we've got all your learning needs and interests covered.
And if you're a teacher in England, you can use a subset of the Raspberry Pi Foundation's online courses as credit towards your NCCE Computer Science Accelerator Programme certificate - a development programme with a mixture of online and face-to-face modules to help you confidently teach GCSE computer science. Visit the NCCE website to find out which courses are part of this programme, and book your place today at **teachcomputing.org.**

▶ Illuminating computing

One part where we're very lucky is with our illustrators, animators and videographers. They all work together with the course writers to not only make the course look good but to fill it with memorable content that aids the learners comprehension of the material covered. While writing the course, the author adds notes on content they would like added to their text which they later discuss with the other content creators. The production of these assets is a big job and can take up to two months for each course, despite having different people working on each aspect. As the people producing these are not subject specialists, it's easy for a key part of an illustration to be misinterpreted or part of an animation to accidentally pave the way for a misconception. To avoid this, all of these people have to continue to work closely with the course author and the production

managers to ensure that all of these pieces of media come out looking as close as possible to how they were imagined.

Our lead educators often have to step out of their comfort zone at this point, as they find themselves in front of a camera. Presenting like this is very different to standing in front of a class – looking directly into a camera can feel very strange, especially when there's an autocue to read. The scripts loaded onto the autocue have been adapted from the course text by one of our video production managers to cut them down for time, as well as to adapt anything that sounds more awkward when spoken rather than written. In any case, our videographers guide our educators through the whole process and by the end some find that they even enjoy being in front of a camera.

Greater than the sum of its parts

As the course launch date nears, we have to start putting all of this material together. Our team must adjust animations to match the voiceover timing and edit them into the videos. Other members of the team must send the videos off for transcribing and then check the transcriptions are accurate. We also need to upload the images and write suitable alt text for them, for accessibility. Our wonderful copy editing team check through the text

> ❝ PRESENTING IS VERY DIFFERENT FROM BEING IN THE CLASSROOM

for the inevitable typos and unfortunate phrasings, and fix them to make sure that the meaning isn't lost. And finally we've got to upload all of these things onto the online platform, using a seemingly never-ending array of text boxes and buttons. At this point both us and the web team make a final check, and the course is done!

Now all we've got to do is facilitate the course, responding to discussions, queries and cries for help, until the course run comes to an end. Well, until next time. (HW)
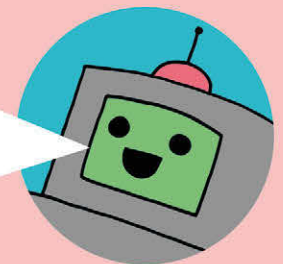
{code club}

START A CODE CLUB IN YOUR SCHOOL!

It's easy to get your school coding! Code Club supports educators to run coding clubs for 9- to 13-year-olds with free, step-by-step project guides for learning Scratch, Python, and HTML/CSS.

"I run a Code Club in our school for children in Year 5 and 6, which is always popular. Code Club activities encourage learning independence, and the children love to share the games they've coded with each other."
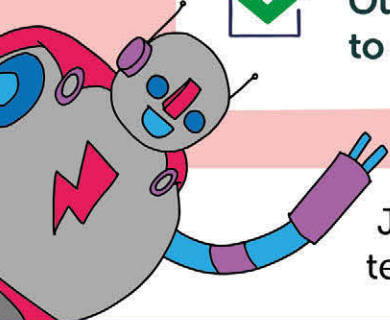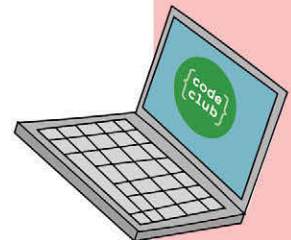
Jill, Teacher

✓ **Code Club is free**

✓ **Code Club offers engaging, hands-on activities that have been designed to spark curiosity and inspire creative thinking**

✓ **Our free online training is perfect for teachers who are new to coding or are starting a Code Club for the first time**

Join the network of over 7000 clubs across the UK teaching more than 100,000 young people to code!

Get involved at codeclub.org

PRACTISE WITH POETRY

Adrienne Tough shares an approach to support students in their use of computing vocabulary

AGE RANGE

14-16 years

YEAR GROUP

UK years 10 and 11

LESSON TYPE

Unplugged (pen and paper)

REQUIREMENTS

- Poem to annotate
- Pen and paper

S tudents can find some computing concepts quite intimidating – particularly the extended vocabulary. So I've been keen to find creative ways to make them more accessible. One method I've developed is writing poetry for the classes – and eventually, I'd love them to write their own, too!

The students are presented with poems from a range of topics. They can choose which one to focus on, or the teacher can decide. The poems can be effective in a revision lesson before an assessment. The students can recall the key concepts taught in the module and annotate the poem to check how much they have learnt.

and helps develop their reading and comprehension. The activity develops key literacy skills, and seems to be well received. This method not only helps develop oracy – an invaluable skill – it also keeps the students engaged as they wait to see who will be selected, and the rhyming helps make the content memorable.

DIFFERENTIATION

Students with SEND can be supported in this activity by grouping students together and giving them more support with annotations. A keyword bank can also be provided so that they can identify the key concepts without having to read the whole poem, if that is easier.

To challenge high attainers, you can encourage them to write their own poem to contribute, or you can set high expectations of their annotations.

For students with English as an additional language the poems can easily be translated online, and images can be paired to help explain the concepts.

> " POETRY CAN BE A CREATIVE WAY OF MAKING COMPUTING CONCEPTS MORE ACCESSIBLE TO STUDENTS

So far, the poems have been used as a classroom-based activity, as well as being given to students as optional revision. If I know the answer is in a poem, I will direct them to read the relevant poem and then turn the question back on the students. This makes them more responsible for their own learning

Students have even asked for extra copies to annotate and some have quoted lines back as an answer to a question posed in a later lesson.

When introducing poems, I have found it has been helpful to model expectations and demonstrate how to annotate a poem before setting students the task. **(HW)**

ADRIENNE TOUGH

Adrienne is a Computing teacher in West London. She recently completed the Teach First graduate scheme. She is keen to investigate and deliver creative measures to help students understand key computing concepts (**@tough_miss**).

OBJECTIVES

✔ To recap different computing concepts (depending on the poem), particularly targeting common misconceptions

✔ To recap and define key words associated with the unit that has recently been covered

STARTER ACTIVITY: INTRODUCTIONS 5 MINUTES

Students will read the poems independently and become familiar with the content. I am keen to save resources, so rather than printing them out, I email them to the students and present them on the board. I also like to print out some spare copies, as some students will want to draw on them.

An optional activity for higher attainers is to write down the additional information that the poem has helped to generate.

ACTIVITY 1: READ ALOUD 5 MINUTES

Display the poem on the board and have the students each read a couplet (that's a pair of lines), then choose someone else to continue reading from there. Otherwise, you can pick students to read. If the students choose to add a rhythm, encourage it! The more fun they have, the more they are likely to remember the concepts that are introduced in the poem.

Circulate around the room to keep the students on task and to highlight key concepts or help with things the learners stumble on. Support students to help them use the correct pronunciation, and read together as a class at least a few times.
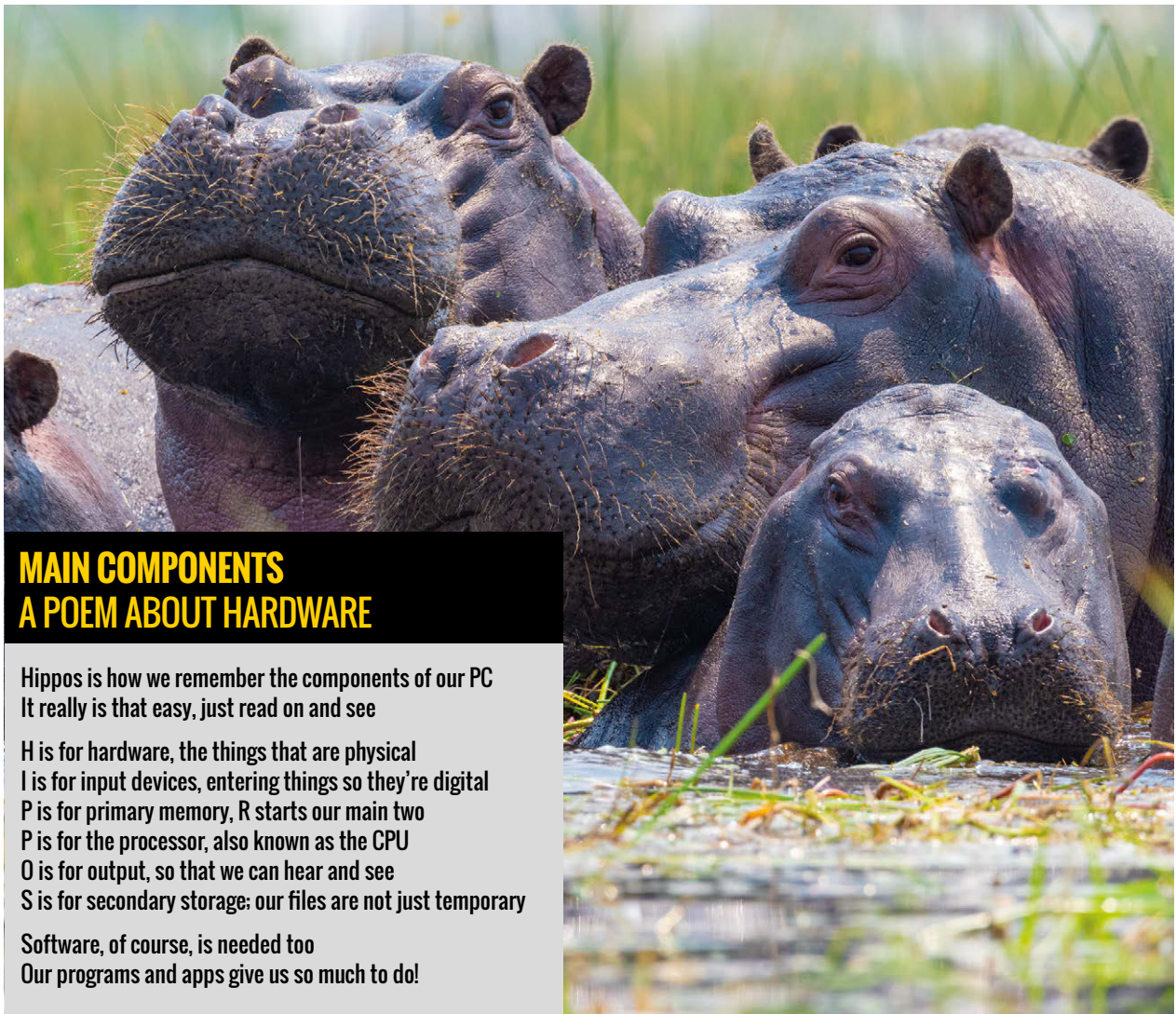
MAIN COMPONENTS
A POEM ABOUT HARDWARE

Hippos is how we remember the components of our PC
It really is that easy, just read on and see

H is for hardware, the things that are physical
I is for input devices, entering things so they're digital
P is for primary memory, R starts our main two
P is for the processor, also known as the CPU
O is for output, so that we can hear and see
S is for secondary storage; our files are not just temporary

Software, of course, is needed too
Our programs and apps give us so much to do!

PRIMARY MEMORY: A POEM ABOUT ROM AND RAM

Directly accessed by the CPU,
That's our primary memory, remember the main two:

First there's RAM, temporarily storing what we're on
But forget to click save and then it's gone!
Being volatile means to save it needs electricity,
It also allows you to access and edit easily.

Next we have ROM, this enables our computer to start
But to access and edit we need to be very, very smart
Is it volatile? No, it's not,
Our contents will not be forgot!

There are others of course, like cache and virtual,
So keep reading on to become even more intellectual!
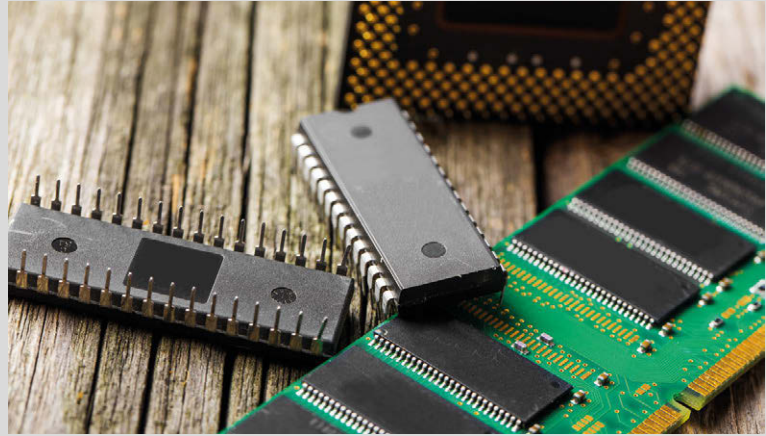
> ### THE ACTIVITY HELPS TO DEVELOP KEY LITERACY SKILLS

RELEVANT LINKS

Adrienne's poems are available at **computingpoetry.weebly.com**.

'Computational Fairytales' by Jeremy Kubica helped inspire Adrienne's idea: **computationaltales.blogspot.com.**

ASSESSMENT

This can work as an assessment or as a starter for the next activity: present the poem again, but with gaps in, then ask the students to rewrite and complete the missed content

This lesson assists revision for end-of-topic assessments. Short quizzes or games such as Kahoot! can be used to ask questions about what the poem is covering.

ACTIVITY 2: MAIN ACTIVITY – RIP IT APART!

Students will now annotate what you have written, so encourage and embrace their critical eye. You could look at what has been highlighted if this was completed in previous step, or highlight the key concepts now. Perhaps ask a student what they would highlight. Students will then be questioned on the content of the poem.

For example, in one poem on computing components, I have written:
"P is for primary memory, R starts our main two."
I then ask what R stands for. If anyone doesn't know, they have to try to find out. I then ask for further elaboration – for example, "What other characteristics do you know of ROM and RAM? What other examples of primary memory could you give?" Pose a few questions and then leave the students to continue to analyse independently, with partners, or in groups. Circulate and assist where necessary.

Students can analyse the poems on printouts or write in their books. They could create mind maps or flash cards based on the reading, so they are also creating their own revision material. Students often do not know how to revise, so going through these methods can benefit them, not only in computing, but in their other subjects too! If working in groups, work or groupings can be swapped and other poems can also be distributed.

ACTIVITY 3: PLENARY ACTIVITY – WHAT'S BEEN READ SHOULD BE SHARED 5 MINUTES

Ensure students have had the opportunity to work with others, and that they have shared their ideas and looked at any other creative projects that have been inspired to get started with.

Towards the end of the lesson, choose students to recall what they learnt from the poem – without looking, if they can. Have the other students elaborate or challenge them as necessary.

CREATE A HOLOGRAPHIC ILLUSION

Nazia Fakhruddin introduces a project in which students explore how 3D programming and optics can be used together to create awe-inspiring results

A hologram is an image that appears to be three-dimensional and can be seen by the naked eye from all sides. A real hologram can be seen without the use of any special lenses or screens; images that appear to be holograms, but are actually produced by placing lenses in a certain way to create the illusion of a three-dimensional ghostly image, are not true holograms; these are called holographic illusions. They were used for the first time to create a theatrical illusion, known as Pepper's Ghost, by an inventor called John Henry Pepper in 1860.

When I first came across this illusion, I thought it was both really interesting and easily programmable using p5.js. I introduced 3D programming concepts to my class using this project and the students enjoyed every step; they also went on to explore and experiment with ideas of their own.
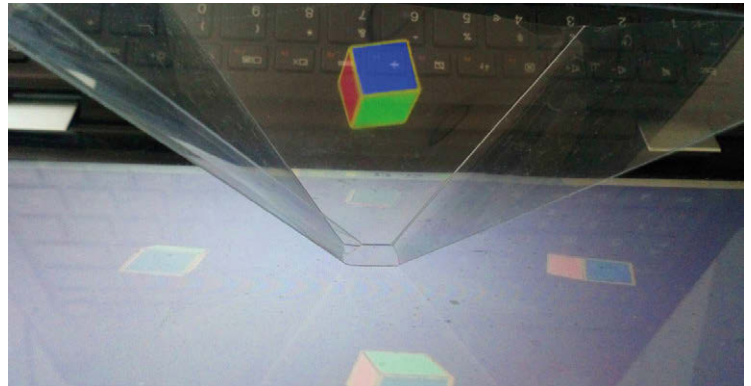
For this project, we will be creating 3D objects using p5.js WEBGL library and a 3D holographic pyramid, which will act as a lens, to observe our holographic illusion. Inexpensive 3D holographic projection pyramids can be bought online, or you can use the top of a clear water bottle. When an image from a screen is reflected at an angle of 45 degrees onto a transparent screen, a reflected virtual 3D image with depth is created. **(HW)**

OBJECTIVES

✔ **3D shapes**

✔ **Pepper's ghost pyramid principle**

WHAT IS P5.JS?

p5.js is a JavaScript library for creative coding, with a focus on making coding accessible and inclusive for artists, designers, educators, beginners, and everyone else. It is free and open-source.

For more information on p5.js, see Saber Khan's feature on pages 44-45 of Hello World issue 8.

DIFFERENTIATION

Support:
Get the students to create the different 3D shapes and rotate them. Ask them to give 3D colour to the shapes.

Stretch and challenge:
Make two shapes and try rotating along different axes by using the rotateX, rotateY, and rotateZ functions. Can your students create a rotating Saturn together with rings?

ACTIVITY 1:

Let's start by creating a new project using the p5.js web editor. You can find it at **rpf.io/p5js.**

It has the following two lifecycle functions: the setup() function, which is called only once at the start of the program execution, and the draw() function, which is called repeatedly, clearing the canvas every time.

Edit the setup() function, by changing the canvas dimensions and adding the parameter of WEBGL in createCanvas().

In the draw() function, change the background() to 0, as we need a black background so we can view our illusion clearly. **1**

```
function setup(){
    createCanvas(600,600,WEBGL);
}
function draw(){
    background (0);
} 1
```


Reflected Light Rays · **Virtual Image** · **Projection Pyramid** · 45 · **Image on screen**

In the draw() function, we need to make four boxes in four positions: up, down, left, and right.

Start by creating the first box between push() and pop() functions. The box will appear in the centre of the stage when you press the play button. **2**

To change the placement of the box, it needs to be translated. To do this, use the translate() function, keeping the x position as 0 and changing the y position to -200. The box should appear at the top of the canvas now. **3**

```
function draw(){
    background(0);
    push();
    box(50);
    pop();
} 2
```

Try challenging your students by then asking them to add and translate three more boxes in a similar way. Each new box should be between its own push() and pop() function. Here's what they might come up with. **4**

```
function draw(){
    background(0);
    push();
    translate(0,-200);
    box(50);
    pop();
} 3
```

```
function draw(){
    //top
    push();
    translate(0,-200);
    box(50);
    pop();

    //bottom
    push();
    translate(0,200);
    box(50);
    pop();

    //right
    push();
    translate(200,0);
    box(50);
    pop();

    //left
    push();
    translate(-200,0);
    box(50);
    pop();
} 4
```

(-ve y-axis)

(-ve x-axis)   0,0   (+ve x-axis)

(+ve y-axis)
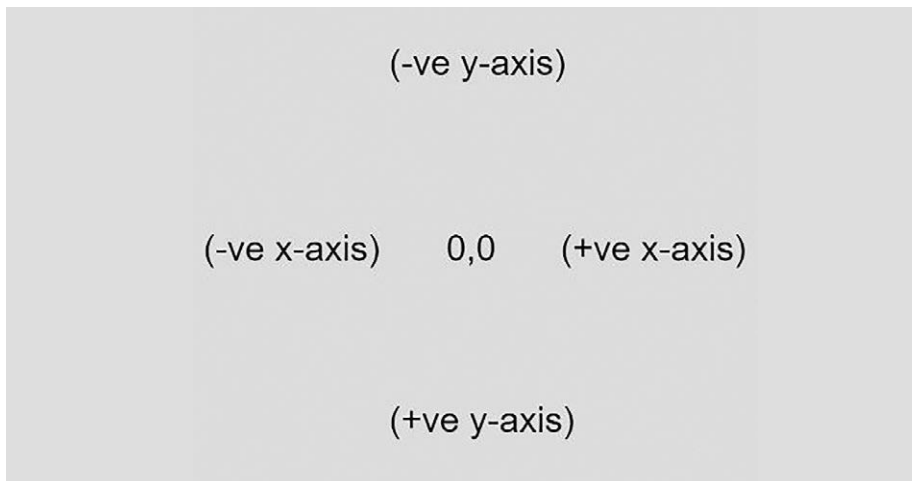
ACTIVITY 2:

Start by making a global variable called angle and initiate it with 0 at the top of the script.

```
angle=0;
```

Then, in the function draw(), increment the angle by 0.003, so the angle changes slowly.

```
function draw(){
    background(0);
    angle+=0.003;
    push();
    ...
```

Now, in the function draw(), on the line below the first translate(), add a rotateX(angle). When you press play, the box should now be rotating on the x axis. Try adding rotateY(angle) and rotateZ(angle) functions as well. Have your students try and make all the boxes rotate in the same way.

```
    push();
    translate(0,-200);
    rotateX(angle)
    rotateY(angle)
    rotateZ(angle)
    box(50);
    pop()
```
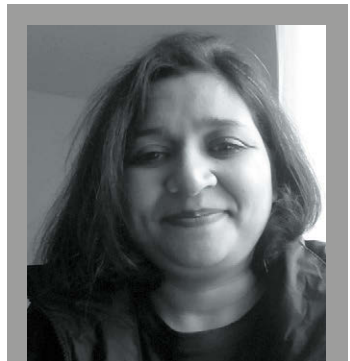

NAZIA FAKHRUDDIN
Nazia is a Royal Institution Computer Science Masterclass Speaker and has been a Code Club UK Volunteer for the past six years in the local library in the London Borough of Hounslow; she is the author of the book *Superfun p5.js Projects* (**@NZAmbience**) .

ACTIVITY 3:

To give 3D colour, use the function normalMaterial() at the top of the draw() function. Also use stroke('yellow') and strokeWeight(3), to highlight the box boundaries.

```
function draw(){
    background(0);
    angle+=0.003;
    normalMaterial();
    stroke('yellow');
    strokeWeight(3);
```

Place the 3D holographic pyramid or water bottle in the middle space between the boxes. Now observe from the side the holographic illusion of a rotating cube.

The complete code for this project can be found at **rpf.io/p5js-hologram.**
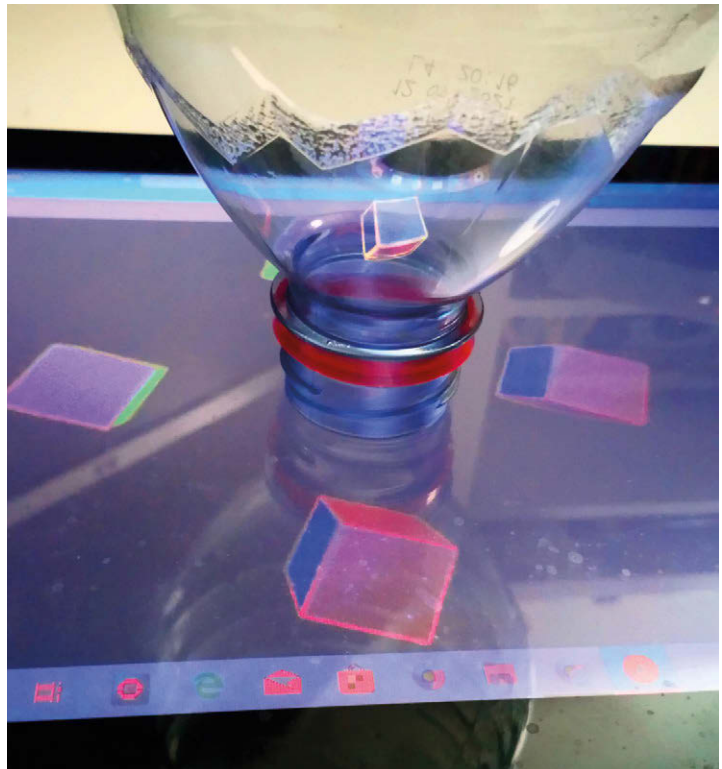
■ If you don't have a 3D holographic pyramid you can also use a clear water bottle to see the holographic image, which will be small but visible

GETTING STARTED WITH PYTHON

Tim Bateup shares his own experiences of introducing students to Python

After being told that my department had no room in the budget for flashy classroom resources, I needed to come up with a way to teach coding in an engaging manner that didn't cost a cent. As a result, I designed a Python workbook, specifically designed for teachers who are looking for a completely free way to teach coding to beginners.

This workbook starts out with a basic introduction to what computer programs are and how they work. Students briefly look at algorithms and flow charts before jumping straight into their first experience of coding in Python. As the term progresses, students watch just 21 short video tutorials that teach them basic coding concepts and develop their understanding. Once they have watched a few tutorials, they then try their hand at completing some practical activities to test their new-found knowledge. In total, the workbook contains 24 activities for students to complete.

Though the video tutorials and associated activities are not covered in this article, I focus on how algorithms can be introduced to the students early in the term in a hands-on and practical way. Students usually start off quiet in this lesson, but once they start acting out the algorithms, the whole classroom comes to life with lots of laughter and plenty of learning and engagement. **(HW)**

OBJECTIVES

✔ To understand how computer programs work

✔ To be able to write algorithms and represent these diagrammatically as flowcharts

✔ To be able to write simple programs using Python code

ACTIVITY 1: THE FIVE-MINUTE CHALLENGE 5 MINUTES

At the start of every lesson, as students come into the room and get themselves sorted, display a question on the projector related to some coding completed in previous lessons or something that will be covered in that particular lesson.

The question could be getting students to draw a flow chart, adding comments to some prewritten code, or even finding errors in code and fixing them. The first two students to solve the problem each receive an award, for example, I brought in Mario and Luigi hats that the successful students could wear for a day. The kids love to wear them, and they make the classroom environment a little more of a positive place to work in.

In today's lesson, the five-minute challenge is for students to write out the step-by-step instructions for making a glass of Milo—a chocolate and malt drink popular in Australia.

Students need to imagine that they are living in the near future and hanging out at home with their pet robot. They start to feel a bit thirsty so they tell their pet robot to make a glass of Milo for them. For the robot to complete this task, the student must give the robot clear and easy-to-follow instructions (an algorithm).

The students are given five minutes to write down the step-by-step procedure for the robot to make a glass of Milo. Students will need the instructions they have written for the next part of the lesson.

> **STUDENTS MUST GIVE CLEAR INSTRUCTIONS**

ACTIVITY 2: ACTING OUT THE ALGORITHM 40 MINUTES

When the students have their algorithms for making a glass of Milo, they will complete each other's instructions. It's a good idea to have some towels on the ground or to move the class outside at this point to avoid making a mess in the classroom.

Ask for volunteers to act the role of a robot. Have your robot stand out in front with the milk, Milo, cup, and spoon on a table in front of them. Ask for another volunteer to read out their algorithm, one step at a time. The robot must act out exactly what they hear for each step and not let any human instinct take over. This usually results in some hilarious outcomes (and a bit of a mess).

After the first algorithm is complete, conduct a quick debrief with the class. Ask questions like:
- What steps worked well?
- What steps didn't work so well?

- How could the algorithm be improved?
- Could certain steps have been completed in a more efficient manner?

Repeat the acting and debriefing for two more algorithms with different volunteers. Once finished, bring the class back together and explain to students how today's lesson is an analogy of how computers read instructions and process code. Students will then begin to understand the importance of breaking down problems into small chunks and creating specific instructions for the computer.

Moving on from here, students continue on through the workbook to look at flow charts and learn how to represent their algorithms diagrammatically. They then dive right into coding where they can see first-hand how a computer processes instructions.

- Mu is a simple Python code editor, useful for students starting to code.

DIFFERENTIATION

The use of video tutorials throughout this unit of work allows students to work at their own pace, in a 'flipped classroom' setting. Students are encouraged to complete the video tutorials in their own time, either at home or during lunch breaks. They then come to class with any questions they might have and will generally spend the lesson completing activities in the Python workbook. While this approach does require time management on the part of the student, they generally do a good job of this once they are shown how.

With video tutorials, students put headphones on and seem to enter their own world - there are very few distractions in the classroom and most students work quietly for the entire lesson. It is common to see students refer back to the video tutorials throughout the course as they complete the activities. This helps them refresh their memory on how to complete certain tasks and saves the teacher from having to repeat themselves.

ASSESSMENT

To assess this unit of work, I like to run two assessment tasks:
1. An exam that is made up of a theory and a practical component.
2. A small project, where students work in pairs to solve a number of real-world problems using Python code.

These assessment tasks will give you a clear idea of the capability of each student, while also lending themselves to a range of 21st-century skills such as critical and creative thinking, communication, collaboration and teamwork, and ICT skills.

TIM BATEUP
Tim Bateup is a Digital Technologies teacher at Noosa District State High School in Queensland, Australia. He has been teaching for 13 years across a wide range of subject areas. He has a YouTube channel: www.youtube.com/tbate54.

RELEVANT LINKS

Workbook Download
helloworld.cc/workbook

Teaching Python Podcast
helloworld.cc/podcast

MAX: A MACHINE LEARNING ADVENTURE

Machine Learning core ideas and concepts can be presented to children like a game

AGE RANGE

7-10 years

YEAR GROUP

Greek Year 9

LESSON TYPE

Unplugged (pen and paper) and presentation

REQUIREMENTS

- Laptop, HDMI cable, large screen
- Whiteboard and markers
- Pen and paper
- Combination lock, chest and chocolate coins (bonus step)

'**I**f you can't explain it simply, you don't understand it well enough.' In this famous quote, Albert Einstein suggest that there is no such thing as a subject so difficult or complex that we cannot explain it to children.

This is the challenge we undertook last May when we set out to explain core machine learning concepts to children, in the form of a game. For the most part of the session, at our CoderDojo in Votanikos, a neighbourhood in Athens, Greece, we used presentation slides. At the end of each task, the learners had to find the same answer to that of our robot, Max, in order to succeed. We did not use any coding or engineering parts, since our 'robot' was an avatar. We had eight children aged 8–10 attending the club at the time.

Before the black box

A common perception of machine learning is that it works like a black box, or like magic: that it receives data from one point and produces an answer at the other, and it is up to the algorithm to interpret the data in an appropriate manner in order for the model and reach a decision. While this is partly true, there is a very important stage, before the use of each model, which we call data preprocessing and feature extraction.

In data preprocessing, we use various ETL (Extract, Transform and Load) methods to process the data and clean it as much as possible. For example, if we provide images that are dark and blurry, then the model would not learn from them.

In feature extraction, we need to detect the useful information that is hidden inside the data, which will be used later in machine learning algorithms. A better way to summarise that would be 'find those characteristics which best define your data'. Feature extraction is performed by the user manually and is not part of a machine learning algorithm.

The next important issue is the selection and training of the model. This is again strongly related to the nature of both the problem to address and the data. For example, if you want to predict a yes/no task, your problem is defined as classification. If you know your labels beforehand, this is supervised classification.

A lasting impression

Six months after running this activity, we asked the children a number of questions about this session.

The story and its critical details were mostly remembered. Core machine learning concepts like the features and the decision trees were recalled, whereas there were some difficulties with the 'k-Nearest Neighbours' example. The children enjoyed their reward at the end.

This gives us confidence that we can organise more lessons on difficult subjects in the future. We have learned to focus more on

OBJECTIVES

✔ Discover machine learning

✔ Learn some uses of machine learning

✔ Understand and apply basic algorithms of machine learning (no coding)

stories that involve children as an essential part of that story, and be more careful with concepts where higher levels of mathematical ability are required, where our the approach needs to be adapted. But we are on the right path.

If you find this adventure as interesting as we did, then you can try making your own sessions based on this material. Children will have fun experiencing their first contact with machine learning concepts, getting prepared for a digital world where these concepts will most likely have an important role. You can contact us via email at **alzeakis@gmail.com** and **ezeakis@gmail.com**, with any questions. We will be more than happy to help you where we can! **(HW)**

ACTIVITY 1: SETTING THE SCENE:
MAX THE PIRATE-ROBOT   5 MINUTES

We started the series of activities by introducing Max:

'Max is a robot that has a dream of becoming a pirate and finding treasure. Since he doesn't know a lot about the world, we need to help him. We will help him find a ship, a crew, a map, the treasure island on the map and then to find the treasure on the island.'

ACTIVITY 2: MAX FINDS A SHIP:
THINK LIKE A COMPUTER   5 MINUTES

'Max needs to start his journey by finding a ship. But he doesn't know what a ship is or what one looks like. We need to find a way for him to understand what a ship is.'

Our learners need to understand that there are no obvious answers when it comes to computers. We need to provide data to computers in order for them to understand. Max has no idea of the concept 'ship' except that he needs one. We asked our learners what would be the easiest way for Max to learn about ships and they all agreed to show him images. Therefore, to start with, we provided him with images of ships, so he could understand what a ship looks like.

ACTIVITY 3: MAX FINDS
A CREW WITH FEATURES   15 MINUTES

'Max has to find a crew for his new ship. There are many people he can choose from, but he has to be smart. If he chooses two similar people, then they may team up in the future and cause a mutiny onboard. How can he learn things about them, that will help him see if they are similar or dissimilar?'

In this step, the mission was for Max to find a crew for his new ship. This discussion introduces what features are and why they are important for the solution of each problem. We showed our learners specific images of people and asked them questions like: What makes each picture different? Why is Pirate 1 different to Pirate 2? And finally, is there a global good feature that can always be used to resolve any given problem?

The answer to the last question is no and the children saw this. For example, if you are looking for a picture of a child, then you would discount any bearded people, but not all non-bearded people are children. So, we asked them to find the two most dissimilar people in the photos. We used eight predefined features (e.g. hair length, beard, eye colour, etc.) and then found the two images which were least similar. We identified that two dissimilar people would probably not team up and cause a mutiny onboard. A symmetrical task would be to find the two most similar people with the notion that they would collaborate more easily. Therefore, each problem requires a different approach in the features identified.

ALEXANDROS ZEAKIS AND MANOS ZEAKIS
Alexandros Zeakis is a Data Analyst in a research centre. He is interested in inspiring young people within Data Science.
Manos Zeakis is a DevOps Automation Engineer. He volunteers as a Dojo Champion and Raspberry Pi Foundation Translator.

ACTIVITY 4: MAX FINDS THE GUIDE: DECISION TREES — 15 MINUTES

'Max has to go to the market to find a specific person that will give him the map. There are other pirates in the market as well, that will try and trick him, so he has to learn beforehand how to identify the correct person. He has already learnt about features, so is there a way for him to utilise that knowledge and find his guide with the correct questions?'

Here, we introduce our first machine learning model — a Decision Tree. It's a relatively easy model to explain as it uses words rather than maths, and can be drawn. The general idea is that you start from the first question (root) and each answer leads to a different question (branch). If the tree is built correctly, the last question should lead to your answer (leaf). The game *Guess Who?* works this way — ask the right questions and based on the answers, you will reach the right person. Again, by using specific images of people, we designed our own Decision Trees based on features that our learners detected in the images. Please note, there is no one correct answer!
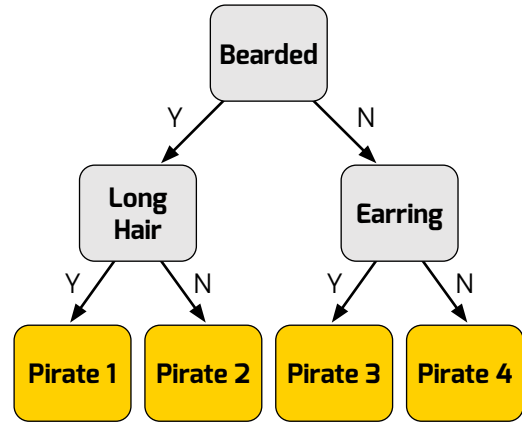
■ Decision Trees resemble the game *Guess Who?*. Max uses this approach to find the person with the map.

ACTIVITY 5: MAX FINDS A MAP

15 MINUTES
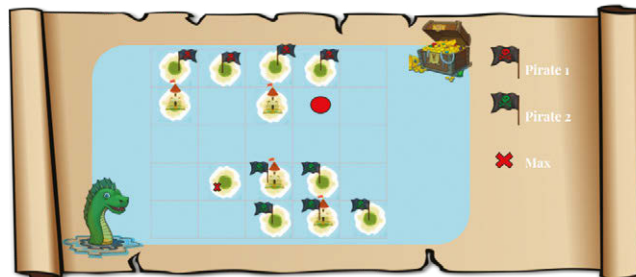
'Max visits the area on the map, only to find that a war rages between two major pirates that want to conquer the same islands Max wants to. He could wait for the war to be over and then find any unoccupied island, but he is impatient and wants to start anyway. Based on the ancient code of pirates regarding naval battles and his wit, Max will try to work out the outcome of the battle using these rules."

Another machine learning model that can be easily explained in natural language is 'k-Nearest Neighbours'. When a predefined distance exists on labeled data, a new unforeseen piece of data can be labelled in relation to this previous data (hence the 'Nearest Neighbours' part). The main difference with Decision Trees, that we used above, is that Nearest Neighbours requires maths. This is the main reason that the exact form of this model was not used and we adapted the model to a game so that the learners could understand the main idea. The game had three simple rules (see image) and then the learners predicted each pirate's move in the map, based on the given rules and the benefits for them. The final state of the map after the battles exposed a single island that Max could safely visit.

RULE 1
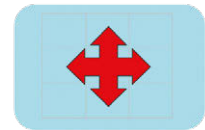
A ship can move up, down, left or right

RULE 2
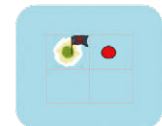
A ship can occupy an island if it is adjacent to it and there is no castle

RULE 3

A ship can occupy an island with a castle if there are two occupied islands adjacent to it
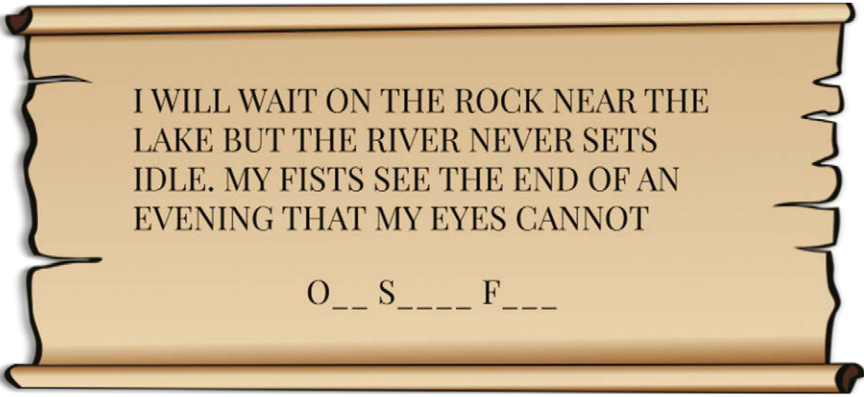
ASSESSMENT

There are two ways to assess the learners throughout the lesson.

The competitive way would be to split the learners into groups and have them solve the puzzles in each task as a group. Each group that solves the puzzle first takes the most points, the second group one less, etc.

The collaborative way would be to have the children solve each puzzle as one group.

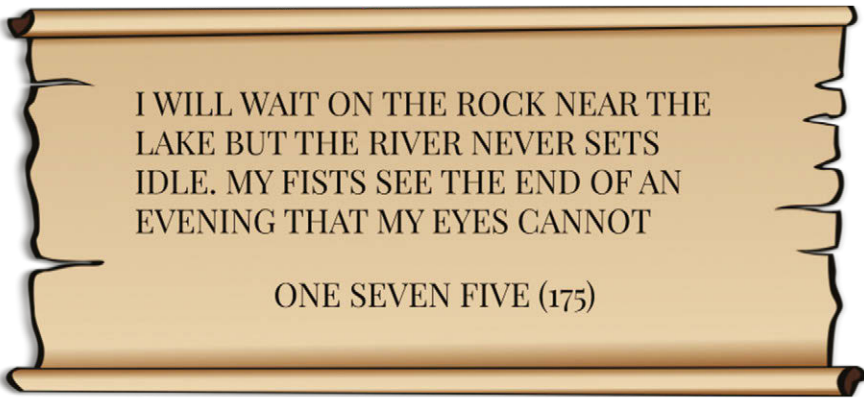We preferred the collaborative way.

■ Initial unsolved puzzle

ACTIVITY 6: MAX FINDS THE TREASURE: LANGUAGE MODELS   15 MINUTES

'Max enters the cave on his island and finds a chest that is locked. This particular lock requires no key, but a numerical combination. His only hint is a piece of paper next to his chest, which includes a small phrase with a riddle at the end. How can he use this phrase to find the answer?'

Next, we used language models from text analytics. The main idea is that when you write a new text, you don't always produce new context but rewrite something you have written previously. If someone has an old written text of yours and you begin a new phrase, then they will most likely be able to predict what you will say. This is a theory based on probabilities, but we simplified this by looking at the frequency of words used when the only information was one letter. For instance, if you have the phrase 'This is an item' and someone gives you 'i_', you will think that 's' will most likely be the next letter, since 'is' occurs twice and 'it' only once. Following these rules, Max used the given text (see image) and found the secret combination to unlock the chest, claim his treasure and achieve his dream of becoming a pirate.

   As a bonus, Max offered our learners a second chest — physical this time — with a second phrase, where each team of two learners would have to find the correct digit. When all teams found the correct digits and unlocked the chest, they found a treasure of chocolate coins inside.

■ Final solved puzzle

MAKE ME HAPPY: MACHINE LEARNING FOR KIDS

Dale Lane shares a machine learning resource for primary-aged students

O n my website **machinelearningforkids. co.uk, I've built a tool that introduces machine learning by providing hands-on experience for children.** It provides an easy-to-use, guided environment for training machine learning models to recognise text, numbers, images, or sounds.

I believe it's important that we introduce this to every kid in the classroom—not just the ones who enjoy coding. We need to give every child enough basic literacy to be able to appreciate how the world around them works.

If they are presented in the right way, I'm convinced that kids — even at primary level — are able to understand the basic principles and can build fun projects.

In this activity, students create a character in Scratch that smiles if you say nice things to it and cries if you say mean things to it. **(HW)**

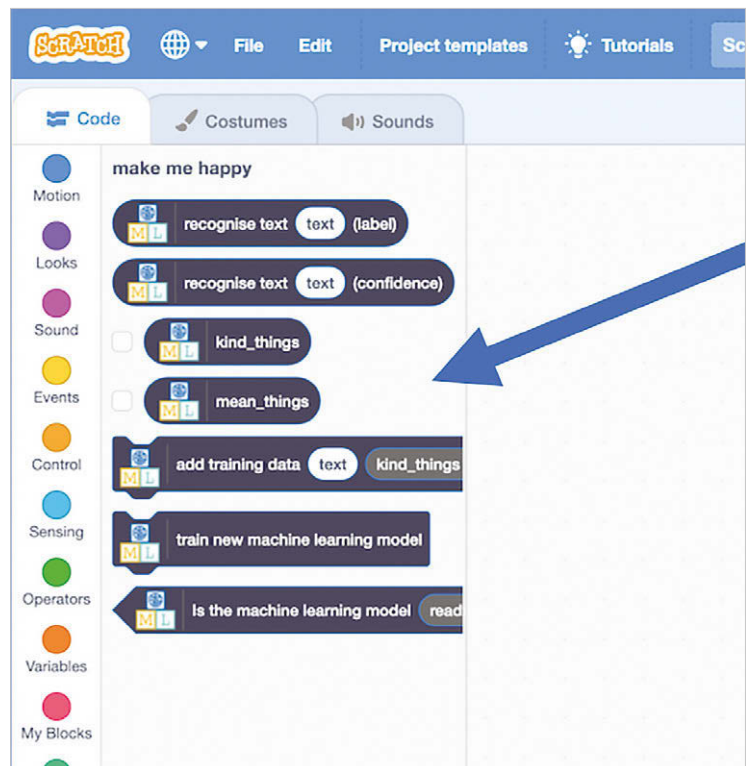## ❝ KIDS ARE ABLE TO UNDERSTAND THE BASIC PRINCIPLES OF ML

STARTER: CREATE A CHARACTER

15 MINUTES

Firstly, students need to get set up on **machinelearning forkids.co.uk**. They can use the 'try it now' function for free — this lets students use the site for up to four hours without registering, which is long enough for most lessons.

■ Click the 'Open in Scrach' button. You should see new blocks from your project at the bottom of the list.

HEALTH AND SAFETY REQUIREMENTS

Younger students may get carried away when writing insults to train the machine learning model. It may be helpful to set boundaries for what language is appropriate.

ACTIVITY 1: A FACE THAT REACTS 20 MINUTES

Once set up on Scratch within the machinelearningforkids.co.uk environment, students will need to create faces to respond to statements. Time management is important here: students often lose track of time drawing their face and don't leave enough time for training or coding.

Students will need to create a new sprite with three costumes — faces that reflect 'sad', 'not sure' and 'happy'.

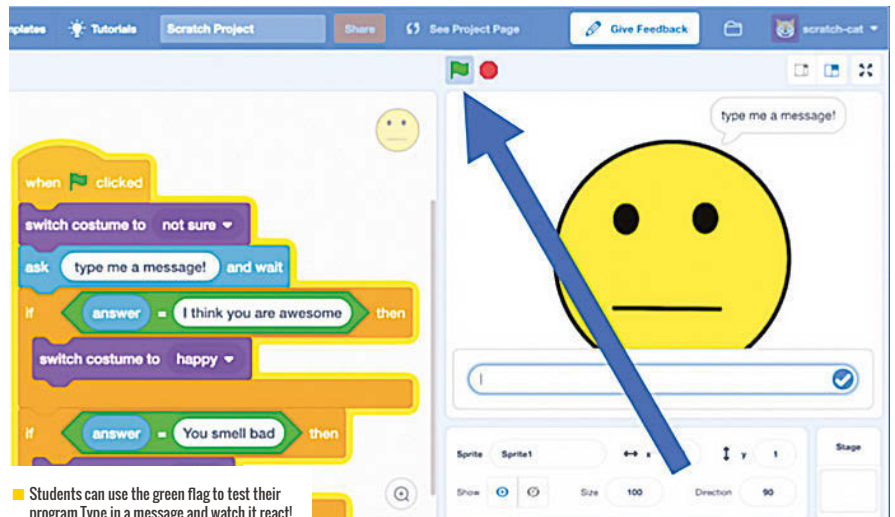■ Students can type in a message and watch the character react
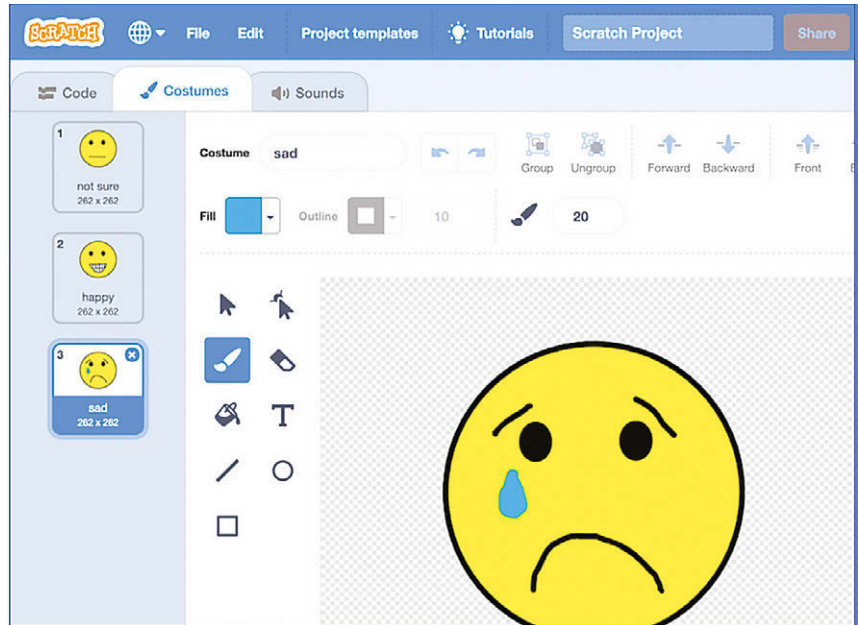
LEARNING OBJECTIVES

✔ Sentiment analysis

✔ Supervised learning

DALE LANE

Dale is a Developer for IBM with a background in artificial intelligence and machine learning. He has worked on solutions for IBM clients using machine learning, as well as being a developer for many years on IBM's AI platform, Watson (**@dalelane**).

■ Students can use the green flag to test their program. Type in a message and watch it react!

DIFFERENTIATION

■ **Write a reply:** Instead of just changing the way they look, make your character reply, based on what it recognises in the message.

■ **Try a different character:** Instead of a person's face, why not try something different, like an animal? It could react in different ways, instead of smiling. For example, you could make a dog that wags their tail if you say something kind to it.

■ **Different emotions:** Instead of kind and mean, could you train the character to recognise other types of message?

■ **Real world sentiment analysis:** Can you think of examples where it's useful to be able to train a computer to recognise the emotion in writing?

ACTIVITY 2: SUPERVISED LEARNING 20 MINUTES

So far, students have created a character that should react to what people type and programmed it using a simple rules-based approach.

However, if you want the character to react to other messages, you need to add more 'if' blocks. The problem with this is that you need to predict exactly what messages the character will receive. Making a list of every possible message would take forever!

Next, we try a better approach — teaching the computer to recognise messages for itself. Students will train a computer to recognise text as being kind or mean. Instead of trying to write rules to do this, they will collect examples. These examples are being used to train a machine learning model.

This is called supervised learning. The computer will learn from patterns in the examples you've given it, such as the choice of words and the way sentences are structured. These will provide the basis for the computer to recognise and identify new messages.

The examples your students give will be used to train a machine learning model. Back in Scratch, they will then be able to use a new custom 'recognise text … (label)' block which will return either 'kind things' or 'mean things' based on the training you've given to the computer.

They can use this to choose the costume to switch to. Students can then test their projects: type a kind message and press enter. The character should smile. Click the green flag again. Type a mean message and press enter. The character should look sad. This should work for messages that you didn't include in your training.

Recognising **text** as **kind_things or mean_things**
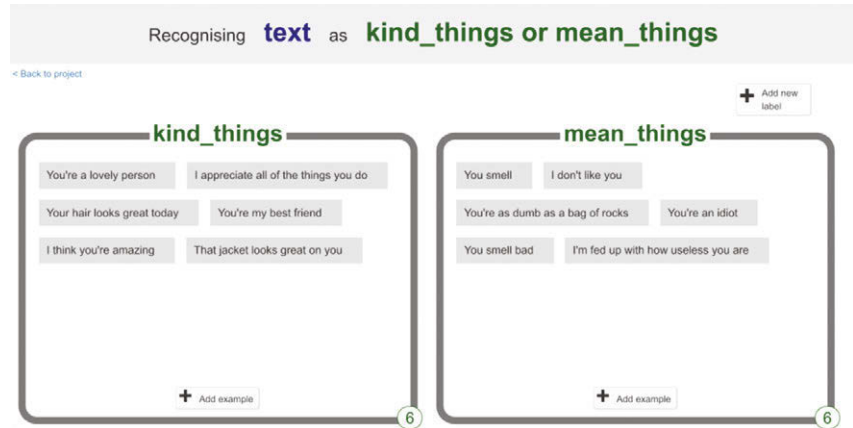
SOME TIPS FOR TRAINING A MODEL

More examples
The more examples you give it, the better the computer should become at recognising whether a message is kind or mean.

Try and be even
Try and come up with roughly the same number of examples for kind and mean.

If you have a lot of examples for one type, and not the other, the computer might learn one type is more likely, which will affect the way that it learns to recognise messages.

Mix things up with your examples
Try to come up with lots of different types of examples, such as, make sure that you include some long examples and some very short ones.

RELEVANT LINKS

Full student worksheets and teacher information sheets are available for free at **machinelearningforkids.co.uk**

ASKING AI: WHAT IS A BUTTERFLY?

How can we introduce the concepts of neural networks to 11-year-olds?
A resource developed for the Royal Society encourages the study of AI

Machine learning is based on a rich tradition drawn from taxonomy of species. How can the classification of butterflies and birds help us to understand machine learning and AI?

In this activity, students will learn how to create a machine learning model, train it and investigate their findings.

Most students have daily contact with AI in the form of mobile phone assistants, smart speakers and newsfeeds. Few have been provided with the opportunity to consider how machine learning works.

In the lesson, students will isolate the characteristics of a butterfly, use these to check whether images of butterflies meet the requirements of the model. They will investigate how using multiple examples builds confidence in models and extends knowledge of taxonomy.

When creating a neural network, there are a number of different models that characterise neural nets which can become complicated. This should not be off-putting, as the basis remains the same. **(HW)**

OBJECTIVES

✔ **What is machine learning?**

✔ **How does machine learning work?**

> " STUDENTS ARE PROVIDED WITH THE OPPORTUNITY TO CONSIDER HOW MACHINE LEARNING WORKS

ACTIVITY 1: CLASSIFYING A BUTTERFLY   45 MINUTES

Ask your students to search 'dead leaf butterfly' or 'orchid mantis' – images which may confuse an automated classifier of butterflies. It would be useful here to get your students to consider how they would construct a taxonomy for a butterfly. Suggest that ten characteristics are agreed on. This is a useful paired / small group activity. Features that your students should look for are characteristics such as four wings, six legs and so on.

STUDENTS MODEL THE TRAINING OF A MACHINE LEARNING ALGORITHM

Time and prior learning may mean that not all of your students are able to arrive at ten characteristics. That's fine, what is important here is that a set of features is agreed.

Using the 'Butterfly Identification Machine' students should write the characteristics in the feature circles. It may be easier to code the characteristics, e.g. 'Four wings = A, Six legs = B'. Step one of building the neural network is complete.

The next step is to get students to view five images of butterflies. You could save time by providing these yourself. Armed with the images, ask students to write 0 in the second column of circles if that feature cannot be seen. Similarly, students should write 1 in the circle if they can see the feature, e.g. if two antennae can be seen then the value is 1, if less than four wings are seen then the value is 0.

The final stage in building the network is to find the average of values. For a student identifying seven characteristics out of ten, there is 0.7 probability that what they see is a butterfly. Take this further by finding the average of the averages. The process should be applied to five or more butterfly images.

At this stage, it is a good idea to take time to consider what has happened. Students have modelled the training that a machine learning algorithm undertakes. There are a range of neural net models used in machine learning. The maths behind them gets complex quite quickly but all have their basis in arriving at a

Feature   **Value**
              0 if not in list, 1 if in list
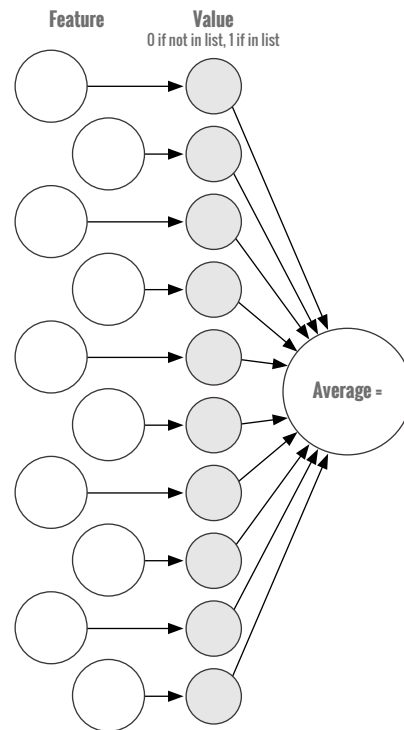
Average =

■ Students write butterfly characteristics in the feature circles, and then mark a 1 or 0 in the second column depending on whether these can be seen in an image. The average of the numbers in the second column then gives a probability of whether the image is a butterfly.

confidence value. In our example, each image produces a value that exists somewhere between 0 and 1.

Now it is a useful exercise to ask students to consider a number of questions:

1. What value would be considered as acceptable for claiming that the image is that of a butterfly?
2. At which value would it be agreed that the image is not that of a butterfly?
3. How many images might we need to classify to feel secure in our model? Can agreeing on characteristics allow us as a class to share the training model and save time?

The last question is useful in illustrating one of the benefits of using computers to classify images. They are fast and able to engage in monotonous tasks.

■ Students use different images to model the training that a machine learning algorithm undertakes.

ACTIVITY 2: EXTENDING THE MODEL   5 MINUTES TO AN HOUR

Once the training model has been agreed, students could repeat the butterfly identification process using images of birds. Ask: could any of the birds be classified as a butterfly? This leads students to reconsider how robust their model is and to a deeper understanding of taxonomy.

Do we need to reconsider which are valid characteristics? This starts to introduce the concept of a multi-layer neural network where the presence, or absence, of key characteristics immediately renders the image as not a butterfly. For example, a score of 0 for four wings, two antennae, thorax and abdomen is rejected as a butterfly. Likewise a score above an agreed value for these items passes the image classification to the next level of the network.

To embed the learning of this exercise, encourage your students to explain what they have accomplished and learned. Understanding the algorithm process is important. Time allowing, the butterfly model could be recreated using flowchart symbols. This could be further extended by creating a programmed version of the classification process.

In working through this topic students get to the heart of what machine learning is. A good deal of how machines learn extends from asking the correct questions. The next step is to consider what actions need to be taken once the model has been trained. For example, once a self-driving car is capable of recognising a pedestrian with an agreed level of confidence the more important part of the process is what does the car do when faced with the pedestrian. Simply knowing that something is the case does not dictate action. This area is rich with philosophical questions to discuss with your students.

AI AND ETHICS

Marc Scott shares some thought experiments your students can discuss to explore the moral dilemmas of AI

In this lesson, students will explore and discuss some of the ethical issues that are considered by programmers as they develop more advanced artificially intelligent software.

I first taught this lesson to a group of 12- and 13-year-old students, and not only were the discussions varied, energetic and fun, but it was really interesting to see the students' thinking about ethical problems in computer science, often defending some very unusual positions.

This is an unplugged lesson. It is a great way to give your students a break from their usual computer based lessons and get involved in a lively discussion. **(HW)**

ACTIVITY 1: PAPER CLIPS 10 MINUTES

Start by sharing the hypothetical thought experiment of the paper clip maximiser with your students. First devised by the philosopher Nick Bostrom in 2003, the story illustrates the need to include ethics into artificial intelligence design.

The story imagines you give a superhuman artificial intelligence system the simple task to create as many paper clips as possible with the resources available. The machine then sets to work towards a goal of turning all matter in the universe, including human beings, into either paper clips or machines that make paper clips.

You can ask your students for additional rules you should give the AI, other than making as many paper clips as possible.

OBJECTIVES

✔ **Evaluate ethical dilemmas that AI can pose to consumers and programmers**

■ One application of AI where ethics is debated is self-driving cars

MARC SCOTT
Marc was a teacher of Science, Computer Science and Systems and Control for 15 years, before joining the Raspberry Pi Foundation, where he now works as a Senior Learning Manager.

ACTIVITY 2: LAWS OF AI 30 MINUTES

Hopefully, in the starter activity, your students will have figured out that the single instruction, 'create as many paper clips as possible with the resources available', was the source of a problem. The machine also needed to value human life and have limits set on its resources and time for making paper clips.

Now ask your students to come up with rules that all AI machines should have programmed into them to ensure catastrophic events do not happen. How can they keep humankind safe from future AI machines? Ask students to share and discuss these rules.

You can introduce the three fictional laws of robotics, invented by the science fiction writer Isaac Asimov in 1942:

- **First Law:**
  A robot may not injure a human being or, through inaction, allow a human being to come to harm.
- **Second Law:**
  A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.
- **Third Law:**
  A robot must protect its own existence

as long as such protection does not conflict with the First or Second Laws. You can discuss whether these are sufficient rules or whether there might be scenarios that might still cause problems.

After discussing these rules, you can introduce the trolley problem: a runaway train is careering towards five unsuspecting workers. Do you pull a switch to divert the trolley onto another track, where only one man works alone, or do you do nothing?

This thought experiment was devised by philosopher Philippa Foot in 1967. It pits two schools of thought against each other: that the morally right thing to do is to save as many lives as possible so you should pull the lever, or the morally right thing is to never kill a person, so you shouldn't touch the lever.

You could introduce this by giving the students each a card with the problem, ask them to read it silently and come to their decision before discussing it in groups. These cards could contain variations of the problem: What if there were 1000 people on the track? What if the one person was a child and the five were terminally ill? What if instead of a lever you have the option of pushing someone in front of the train to stop it running into five people?

Since its inception, the trolley problem has been widely used in psychological experiments about ethics and morals. In 2007, a study showed that 89 percent of people would push the lever, but only about 11 percent would push someone in front of the train. Do your students' choices follow a similar pattern?
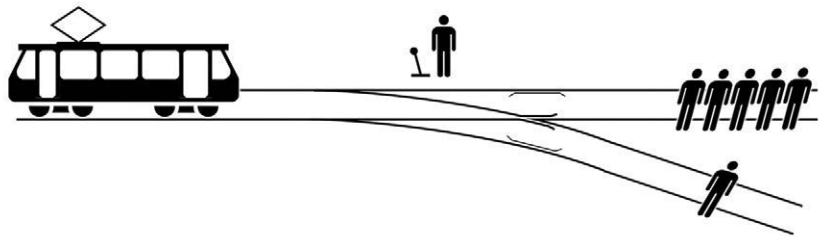
ACTIVITY 3: SELF-DRIVING CARS 10 MINUTES

Explain to your students that there are now an increasingly large number of self-driving cars being produced and running on our roads. Have them imagine a scenario where a self-driving car is driving at 70mph down a road, when suddenly a person steps into the car's path. Then present some or all of the following questions for them to answer.

Should the car hit the pedestrian or should it swerve into the wall alongside the road, where it might kill the passenger?	If they were to program the car's AI, whose life is more important, the pedestrian or the driver?	What happens if there are three passengers in the car, along with the driver?
Should the driver be able to see the code the car is running and alter the code to make different decisions?	Who should decide how the cars behave? Is it the responsibility of governments, the manufacturers, the programmers or the buyers?	Would they be happy to be the owner of a car that has been programmed in such a way?

Recent research has shown that most people agree that self-driving cars should be programmed to save as many lives as possible. However, they would not buy a car that they knew was programmed to sacrifice its driver. This is a problem that car manufacturers are currently grappling with: the Mercedes self-driving car due for release this year will be programmed to prioritise its passengers.

ASSESSMENT IN
SECONDARY COMPUTING

Alan O'Donohoe and **John Woollard** offer practical, common-sense advice on how to assess Computing for ages 11 to 16

I t's important for teachers to be clear from the start about their reasons for assessment, so they can be certain that the assessment strategies they use are fit for purpose. Although the motivation for assessment can be intrinsic, it's more often driven by a strong outsider influence, such as the expectations of school leaders and parents or the need to meet performance targets. We'll look at three purposes for assessment in this article.

■ **Assessment of learning:** The purpose is to assess what the learners can now do, what they now know, their improved understanding and, importantly, their attitude to learning, with the intention of recording, tracking, and reporting this to others. This type of assessment may not be highly valued either by the teacher or the learner, but it is necessarily fed into the accountability agenda in schools, from school to school, across the school, across the subject, and across the teaching team. These assessment results are typically used to inform pupils and parents about the progress that has been made over a set time period. Therefore, it is vital that the

■ Regular assessment can lead to more engaged learners

measurements are viewed with the beginning of the time period in mind, as well as the end.

■ **Assessment for the learner:** The purpose of this is fundamentally to support each individual learner. When an individual learner can objectively observe their own progress, this will help them to understand where they are in terms of their own individual learning journey. Through regular analysis, learners can more accurately identify any gaps in their knowledge and understanding and take appropriate action to address these gaps. This can lead to much more engaged learners in class, who are better informed about their own abilities and progress.

■ **Assessment for teaching:** This is intended to inform the teacher in order to help shape their teaching plans. Assessments focus on the success criteria for teaching and might include measures of learner engagement, attainment, and progress. However, they may also be based on a professional judgement of what constitutes a 'good lesson' and the reasons for that judgement.

It is likely that some assessment is taking place to fulfil each of the reasons listed; it could be your own professional judgement about how well the learners are doing when you teach in a particular way, informal monitoring of outputs, or casual feedback to students about their attainment in particular tasks.

Three dangers to watch out for

Danger 1: Don't seek the holy grail of assessment. Be wary of the dangers of being oversold apparently attractive assessment solutions by enthusiastic advocates. Some solutions that work extremely well, perhaps for a teacher in one particular context with boundless energy and motivation, are probably unsustainable in the long term. In reality these solutions may have:

■ High start-up costs in terms of investing time to learn to use the system and adapt the resources
■ High ongoing maintenance costs in capturing and inputting data
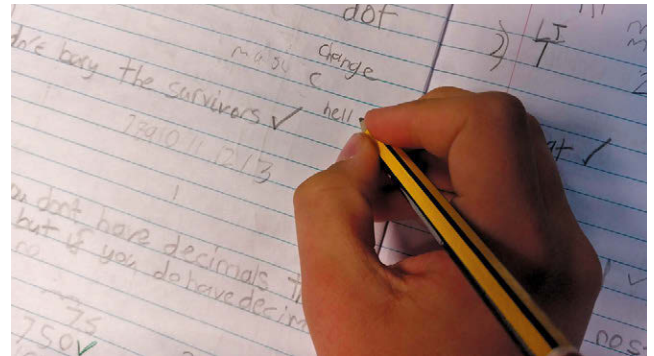■ Low value in terms of informing stakeholders
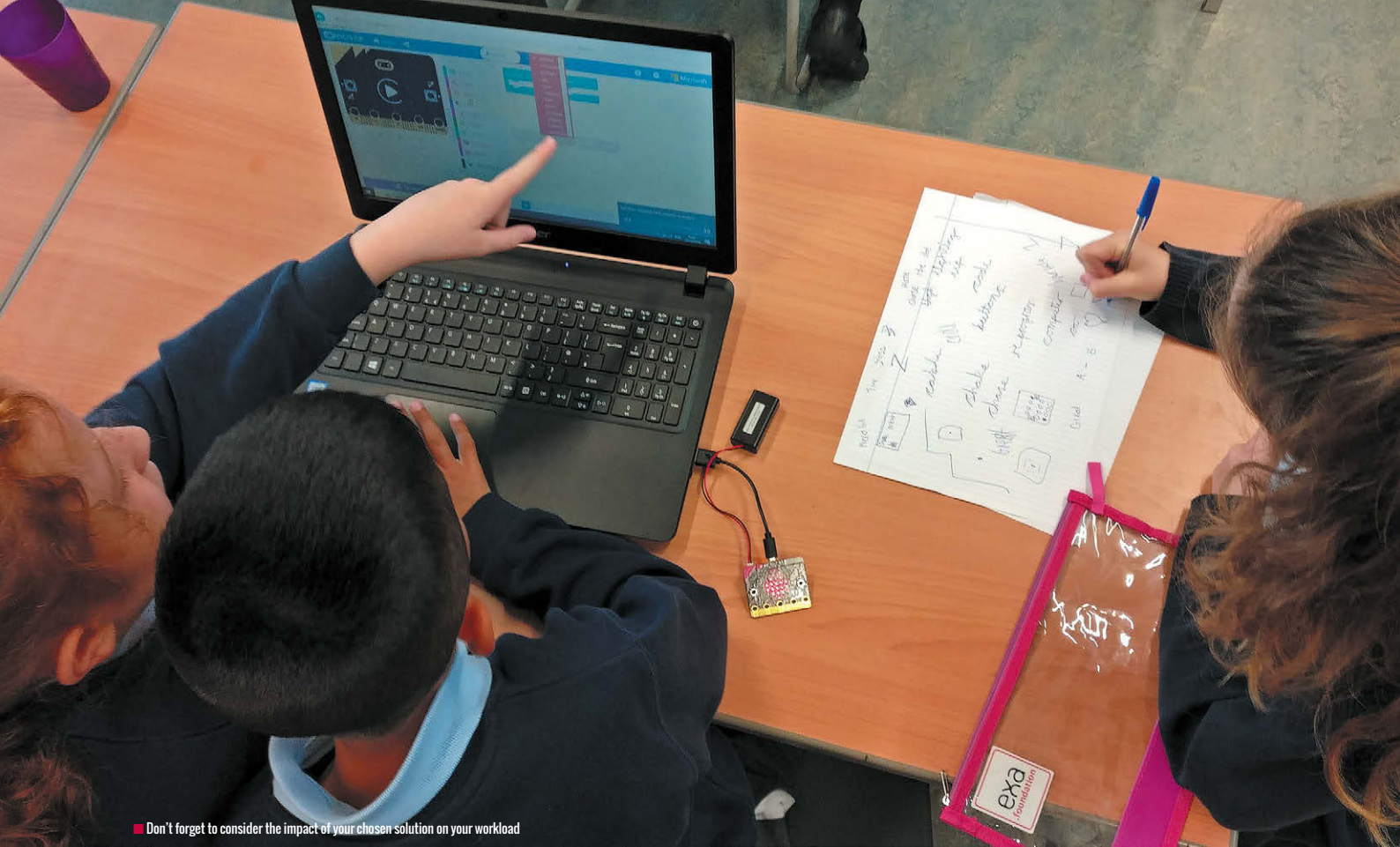
■ Don't forget to consider the impact of your chosen solution on your workload

DON'T DISMISS SIMPLE SOLUTIONS, AS THESE MAY BE THE ONES THAT WORK BEST FOR YOU

■ We have suggested four tried and tested assessment strategies

It's really about finding an approach that you think will work for you and your learners in your setting, and then refining and developing it to get the most from it. If this is the way you approach assessment, then the holy grail will eventually come and find you!

Danger 2: Consider the workload impact. Your well-being is incredibly important, and there is little value in creating a wonderful-sounding assessment strategy if it is impractical to use on a regular basis. It would be wise to calculate how much time would be needed for you to keep on top of a solution, and then consider this in terms of all the classes you teach. 'Little and often' is the key with assessment. Do not dismiss simple solutions, as these are often the ones that will work best for you. Solutions that are 'just good enough' are the most likely to serve you well in the longer term. You can always tweak and improve a system that is just good enough once it becomes embedded into your regular routine.

Danger 3: Beware false success. Some commercially available assessment solutions do not limit or control students' attempts at multiple choice quizzes, meaning that students can choose to repeat the tests until they score 100%. These can be popular with

ALAN O'DONOHOE
Alan has more than 20 years' experience teaching and leading technology, ICT, and computing in schools in England. He runs
exa.foundation, delivering professional development to engage digital makers, supporting the teaching of computing, and promoting the appropriate use of technology (**@teknoteacher**).

JOHN WOOLLARD
John Woollard is a teacher trainer, author, and university lecturer working in both computer science and special educational needs. He is a member of the BCS School Curriculum and Assessment Committee, and chair of CAS Assessment working group.

▶
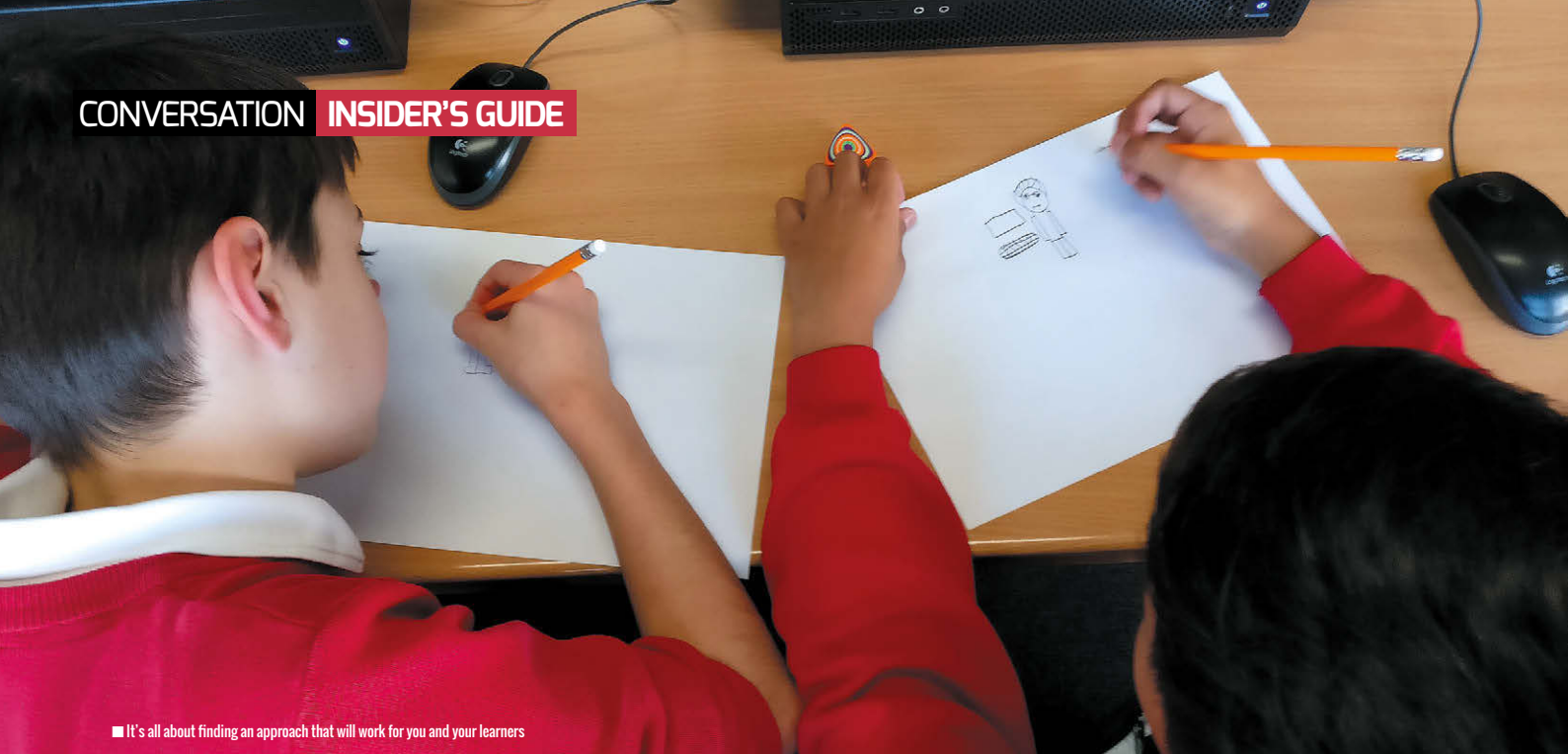
■ It's all about finding an approach that will work for you and your learners

▶ students, who love the rewards of achieving high scores in tests. However, this creates a dangerous, false sense of success in which students have only learnt the answers to the tests by rote, and this learning does not convert to the same success in an exam that is designed to assess their understanding. If you are using one of these systems, take steps to prevent your students from repeating the quizzes too many times. It's best not to use a particular question or quiz more than three times a year.

Recommended strategies

In this article, we are recommending four strategies to consider.

Multiple choice quizzes (MCQs): Instant quantitative results

Multiple choice quizzes should be a prominent feature of assessment in all Computing classrooms. Computer-marked assessments provide teachers and learners with a rapid means of achieving instant quantitative results that can be used to inform learners of their progress and provide data to help teachers diagnose gaps in understanding. It's important that the same questions are used before a period of study as well as after. There is an initial time investment in importing or creating the quizzes, but once they are set up, the software automates the assessment. If these are used regularly, over time they will help paint a picture of the learning landscape in your classroom.

Exam-style assessments

These are short, one-page exams lasting just ten minutes, presented in the style of an exam paper every four lessons, with a variety of questions worth one, two, four, and nine marks, to be completed either in exercise books or on paper. After the time has elapsed, students swap papers and mark them in class under the direction and supervision of the teacher, who then collects the papers for moderation. These give the students a more objective form of assessment in an exam-style experience; while these are not as easy to administer or assess as MCQs, they do provide valuable data that feeds into the assessment of learning.

> " MULTIPLE CHOICE QUIZZES SHOULD BE A PROMINENT FEATURE OF ASSESSMENT IN ALL COMPUTING CLASSROOMS

Attitude assessments

Every week, students are prescribed a set topic to study out of class. This is described in depth in the 'No Headaches Homework' article in Hello World issue 8. Students are expected to provide evidence of their studies in an agreed format every week, and this is graded out of five. If the teacher collects the books for grading every two weeks, it's possible to assess both pieces of work from a single class in less than 30 minutes. Over time, this generates another set of numeric data that teachers can use to inform their assessment judgements. As well as providing a rich programme of learning out of class, this strategy provides teachers with some valuable data about students' attitudes to learning.

Markbook

This is a subjective assessment that has the benefit of providing long-term data. At a convenient point during each lesson, take two minutes to scan the class and put a mark against each pupil's name:

- 0 means absent
- 1 means present but with little effort or achievement
- + means meeting expectations (whether it be for the pupil, class, or year group)
- * means exceeding expectations

Handwritten entries are easy to change: in this system, a '1' becomes a '+' by adding an extra line; a '+' becomes a '*' by adding an extra two lines.

In this strategy, we make professional judgements based on what we see, but importantly, these judgements are tempered with what

we know about the individual learners, the task being undertaken, and the context of the lesson. This is a classic 'assessment without levels' approach. All judgements are made relative to that pupil, in that class, at that point in time.

The record reveals trends over time, both for the class and individuals, reveals topics that are not doing well, and, importantly, provides a good aide-memoire or baseline when the next stage of reporting takes place.

Assessment records in the form of spreadsheets or forms can be used for professional-looking communication with the subject lead or senior management.

Make assessment a habit

We have suggested four tried and tested strategies that you could use in your own teaching. Each of them can work very well in isolation, so you could start by introducing one of them into your teaching and concentrate on making it work for you before you begin trialling any of the others.

Whatever approaches you choose to use for assessment, aim to arrive at a position where the assessments become a habit, so that you rarely need to...

- Give the processes further thought;
- Provide additional time and/or effort; or
- Reintroduce the system to the learners

For more free, friendly advice about assessing Computing, contact the authors of this guide, Alan O'Donohoe, alan@exa.foundation, and John Woollard, John.Woollard@computingatschool.org.uk. (HW)

CS Teacher	04-Sep Intro to Python	11-Sep loops(1)	18-Sep hardware (inputs)	25-Sep etc.
Abigail	❘			
Addison	○			
Alyssa	❘			
Ava	+			
Benjamin	✳			
Chloe	○			
David	❘			
Ella	✳			
Emily	✕			
Emma	+			
Ethan				

■ A markbook is a classic 'assessment without levels' approach

YOUR QUESTIONS

Q DO YOU HAVE ANY SUGGESTIONS FOR UNPLUGGED LESSONS TO USE WITH A YEAR 8 CLASS WHO ARE ABOUT TO START PROGRAMMING?

A **BEN GARSIDE:** Getting learners to understand that computers need precise instructions is a key objective before they start programming. There are some really interesting offline activities you can do around this, such as getting learners to direct a human robot around the classroom using instructions, whilst you watch out with a keen eye for the learners making human assumptions. Another activity that is a favourite of ours at Raspberry Pi is for learners to form a 'human algorithm' where they have to perform sounds in a sequence. Both of these activities feature in lesson 1 of our Year 7 programming unit on the resource repository (**helloworld.cc/evidence**) and are described in detail.
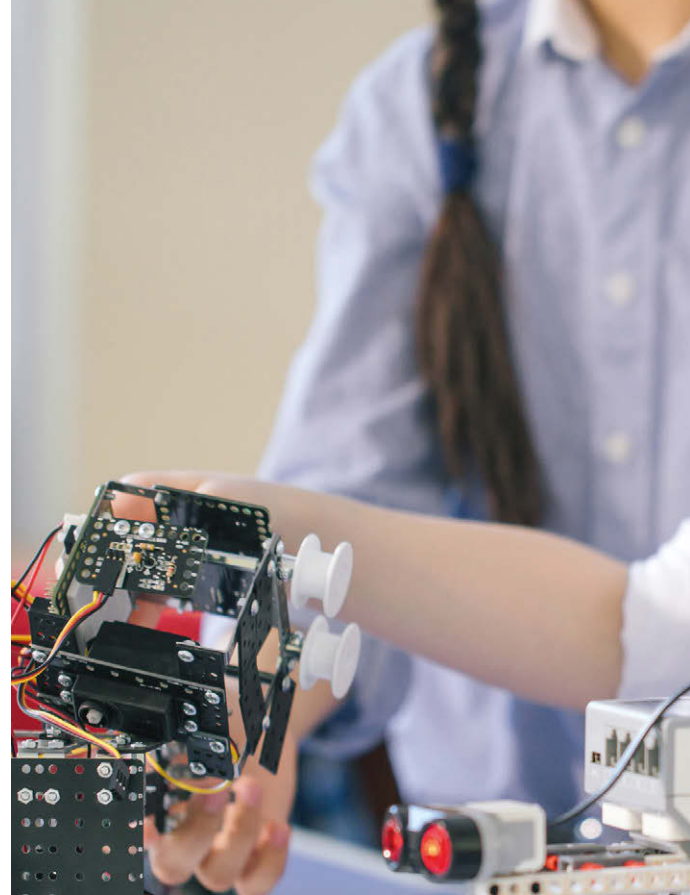
As a shorter unplugged activity later in the unit, you might consider using Parsons problems (where learners have to reorder jumbled up lines of code to form a working segment of code). They are a great way to recap the learning from a previous lesson and could be done independently or in small groups, before going to the computer and testing out their solution (see page 87 of Hello World issue 8).

I'd also recommend checking out the following two websites that have some great ideas for unplugged lessons: **cs4fn.org** ('Searching To Speak' is a personal favourite of mine) and **csunplugged.org**.

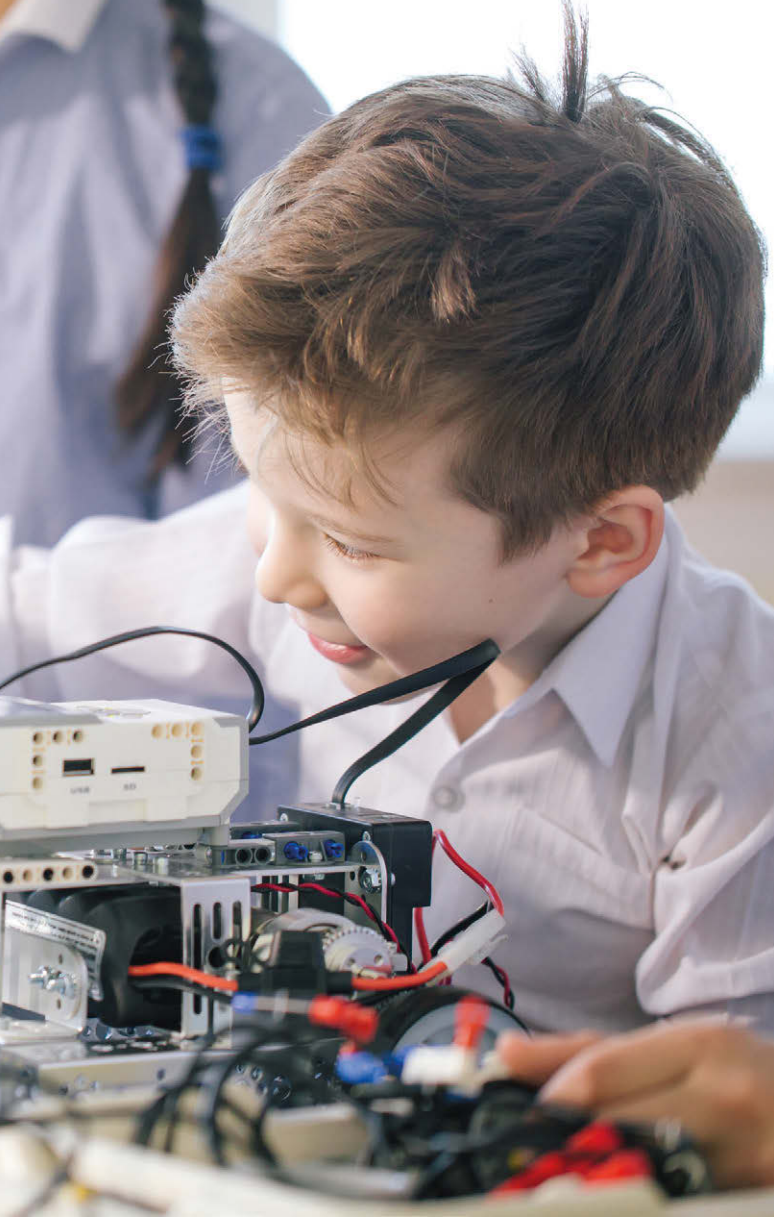**Q** WHAT ARE YOUR VIEWS ON COMPUTING EDUCATION AND SCREEN TIME FOR STUDENTS IN PRIMARY SCHOOLS?

A **SWAY GRANTHAM:** Both adults and children spend a lot of time using screens in their daily lives. Most of the recommendations around a limited time using devices, such as the World Health Organisation's advice, is actually around the idea that screen time is sedentary time, which other activities, such as reading, also involve. However, research such as the LSE Media Policy Project, which reviewed current advice and upcoming research on screen time and related topics, shows the concept is totally misleading.

They suggest instead that it's about:
- screen context (where, when and how digital media are accessed)
- content (what is being watched or used)
- connections (whether and how relationships are facilitated or impeded)

In the modern world, limiting screen time could mean limiting the time someone has to talk to a family member or practising their times tables! It's really important that we all participate in a variety of activities, both online and offline. In our curriculum resources for year 2, we introduce children to the Children's Commissioner's Digital 5 a day which suggests five activities you should do each day, to ensure you have a healthy balance and are not spending too much time doing one of those activities, whatever that activity is.

Q I'VE GOT A SPARE BLOCK OF TIME WITH MY YEAR 8 CLASS AFTER THEY LEARN PYTHON. WHAT SHOULD I TEACH?

A **BEN GARSIDE:** If you're interested in app design, I recommend that you take a look at App Lab from code.org. It is a free-to-use, web-based coding environment where learners can use drag and drop tools to create the user interface elements to the app, but use code to power the app. The code interface is block based by default but can be switched to text based where the learners can code in JavaScript. Code.org have an 'hour of code' activity to allow learners to become familiar with the interface, but keep an eye on the NCCE Resource Repository where we will soon release a full unit of work on app design using App Lab.

MARTIN O'HANLON: If you wanted to take their Python knowledge further, you could consider using Python to create graphical user interfaces (GUIs) or introduce physical computing. There are two online courses available via teachcomputing.org, Programming with GUIs and Teaching Physical Computing with Raspberry Pi and Python, which will give you ideas and materials to get started.

> If you have a question you'd like the Learning Team at the Raspberry Pi Foundation to answer contact us on Twitter via **@HelloWorld_Edu.** Alternatively, email us with 'Question' in the subject line at **contact@helloworld.edu**.

Q MY SCHOOL IS KEEN TO FIT IN COMPUTING INTO WHOLE-DAY SESSIONS. DO YOU HAVE ANY RECOMMENDATIONS ABOUT WHETHER COMPUTING SHOULD BE TAUGHT IN DAY-LONG SESSIONS OR BLOCKED OVER A WEEK?

A **SWAY GRANTHAM:** There are a couple of things to consider when blocking the curriculum. Firstly, are you confident that the curriculum coverage will be the same. For example, if it's usually a six week half term and you've only got five hours in your computing day that half term, how will teachers manage that time limit whilst ensuring there are no parts of the curriculum left out and all are taught sufficiently?

Secondly, by teaching something in one solid day you have fewer Assessment for Learning opportunities. In a weekly lesson, a teacher would reflect on their teaching and what the children learned and make amends to next week's plans accordingly. However, when the next lesson is in 10 minutes, it's unlikely the teacher has the time or resources to change the plan for the lesson if something isn't going well. I also wonder, if the learners will find a day of building on the same concepts challenging for their cognitive load, which may mean they don't retain what they learned as successfully. However, I have not tested this theory.

YOUR LETTERS

A feeling of home

Hello, HelloWorld!
I'd like to just send a quick email to say, "Thank you!"

In the world of Computer Science education, there aren't many avenues for connecting with peers, especially in the U.S., where it's not a mandatory subject area. Often, you're the only CS teacher/advocate at your school, so opportunities to connect are very limited. The internet can be a big and overwhelming place to find comradery and resources that are a perfect fit. Online searches can steer you in directions far from what you're truly searching for. With all of this, it's so easy to feel like you're all alone, with sporadic places to turn to piece together resources, best practices, model lessons and advice from peers.

When I read HelloWorld, I get a sense of feeling 'at home'; I've found my people. There's something in each issue for everyone related to CS education. From teachers to coaches, to policymakers, to advocates, to researchers, and more; everyone can get something from each issue. In my role, I support CS teachers through professional development and I find myself constantly using articles to encourage, challenge and motivate my participants.

Thank you again for all the efforts that go into bringing this to life. It's truly an invaluable resource for continuing to grow.
Rosemary Bianchi

This email made our day, thank you, Rosemary! You've summed up exactly what we try to do here at Hello World.
Sian Williams Page, Editor

HELLO WORLD WRAPPING: FROM PLASTIC TO PAPER

I'm really happy to say that from this issue onwards, Hello World will be shipped to our hardcopy subscribers in paper envelopes.

As I wrote in response to Paul Sutton's letter in issue 9 (page 95), the move away from plastic packaging has been something we've been trying to implement for a while now.

A recent UN report highlighted that plastic packaging accounts for about half of the plastic waste in the world. When discarded in landfills or in the environment, this can take up to a thousand years to decompose, causing numerous problems as it does so. High concentrations of plastic bags have been found blocking the airways and stomachs of hundreds of animal species. The toxic chemicals in plastic packaging transfer to animal tissue, eventually entering the human food chain. And, if current patterns continue, by 2050 the plastics industry will account for 20 per cent of the world's total oil consumption, exacerbating the climate crisis.

Some readers suggested we use decompostible potato starch wrapping to combat this, used by several other publications. However, not everybody has access to a compost bin. Paper recycling is widespread: hopefully it's clear to know what to do with this envelope.

It can often feel like individual actions are futile in battling the environmental crisis but it's thanks to the readers' letters that we were able to justify this switch in packaging. I know I will bear this in mind when considering how I interact with other businesses and brands in the future. Thank you to everyone who has emailed, or left a comment on our survey, to ask for this change.
- Sian Williams Page, Editor

OUR SURVEY

RESPONSES FROM OUR 2019 READER SURVEY

Thank you to everyone who took the time to answer our reader survey. The results are really important to us: we are constantly trying to make Hello World as useful as possible for computing educators and your feedback guides us on how to do this.

This year, we are trialling using shorter surveys at more frequent intervals.

These will only be sent to a subset of our subscribers — the research team at the Raspberry Pi Foundation assure us that random sampling leads to more robust results — so if you receive an email from us asking for feedback, please do take a few minutes to tell us how we can improve the magazine for you!

Excellent magazine to learn what is happening in the computing world for educational purposes. I may not get lessons directly from the magazine, but it helps to generate other ideas that I may not have got without reading the magazine.

I appreciate that the magazine is less frequently published and has more interesting content. It's not just trying to sell something. I wait for the publication but I take a good month to work through it.

I don't often get the time to read the magazine. A digital version would be great as a PDF seems not very accessible: I can't easily start reading one or two articles. I guess I am more of a snacking person when it comes to reading stuff online, rather than a magazine person.

We are starting to share articles online from this issue onwards. Take a look at helloworld.cc and let us know what you think.

I find Hello World is sometimes too focused on one subject rather than a mix of topics. Some of the editions I haven't really read much because of this, particularly those with long articles which I don't feel are that relevant. I prefer it when there is something for everyone. I do enjoy learning about what teachers are doing in other schools but prefer the shorter articles.

Thanks for this feedback, we try to strike the right balance between features that can delve into the details of an idea and easier quick-reads. It's something we'll carry on thinking about.

I find the magazine a great source of ideas and resources that complement my teaching. Could you have a SEN bit about teaching kids with different abilities?

I'm really happy to say Catherine Elliott has started her regular column on SEN. See page 14 of this issue.
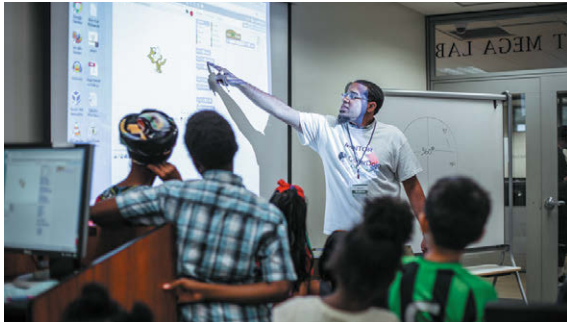
I find the magazine a great source of ideas and resources that complement my teaching.

I have found it very useful as a catalyst to investigate new things or further my knowledge on familiar things.

I live in the US and would like a reasonable cost option to subscribe to a print version.

Great to hear you enjoy the magazine. A print version in the US isn't on the cards in the immediate future, but you never know!
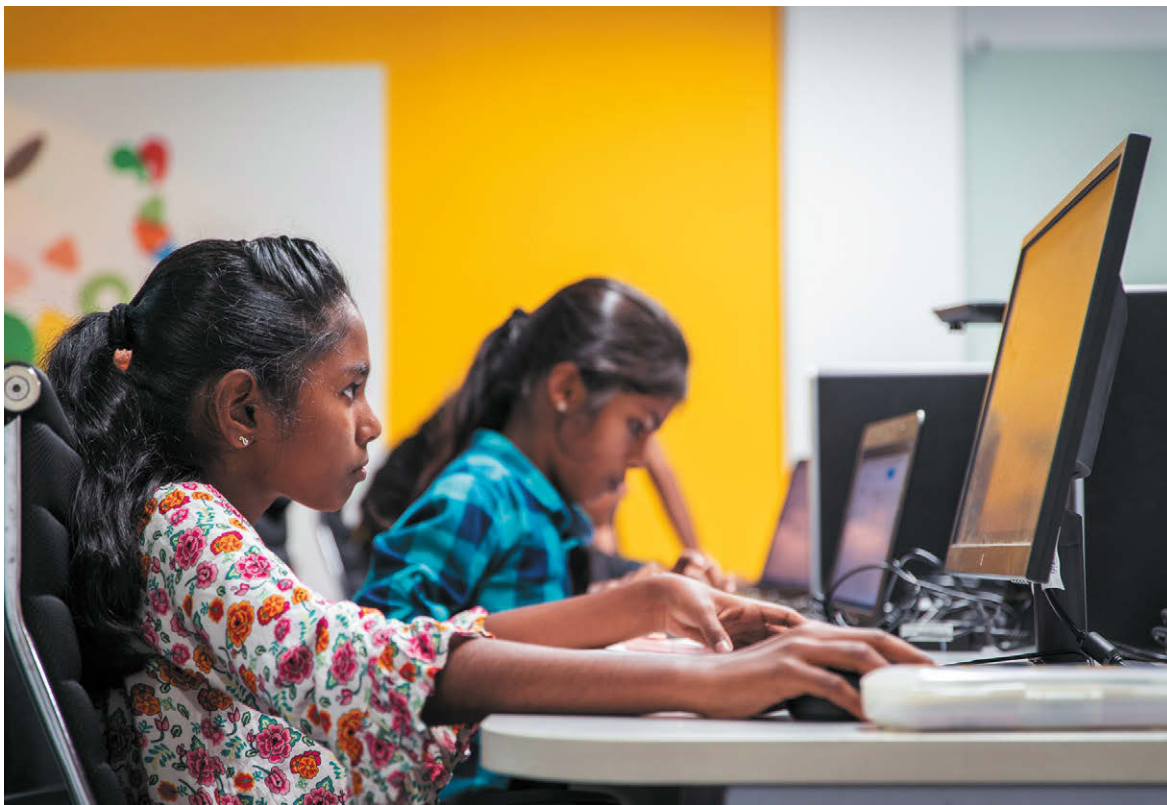
I Love Hello World! Such a fantastic resource for anyone teaching CS! And thanks for making digital editions available for those of us teaching outside the UK.

I love the magazine, and I especially like when there are articles showing specific projects that I can bring to the classroom, in particular Scratch programs.

I think it's an absolutely brilliant resource. I think the thing it does for me is remind me that I'm not alone and that it's ok to try and fail and learn as there is a community of support. It's a rare publication, something that makes me feel part of a community of practice and that's why it's important to me. I may not read all the articles but when I see it come through the post box it reminds me that there are good educators out there doing great things and that I too can contribute to this in some small way

I think it's an excellent resource. Please keep them coming. There's something about holding a copy of the magazine in your hands that's, uh, intangible online.

It looks like the targeted audience for this magazine is teaching age 6-14. I miss what should be taught after this age group (the age of my students are 17-18 yo)

Resources for and features around teaching A Level students is something we can definitely make more space for in the magazine. In the meantime, if you're looking for ideas for how to support these students check out **isaaccomputerscience.org**

BODY IMAGE IN THE PRIMARY SCHOOL

Ben Hall discusses the second edition of a book dealing with one of the most pressing issues for children in the modern world

INFO | **BY** Nicky Hutchinson and Chris Calland | **PUBLISHER** Routledge | **PRICE** £23.99
ISBN 978-0-18842-581-9 | **URL** helloworld.cc/image

When Nicky Hutchinson and Chris Calland published the first edition of this book in 2011, they could not have foreseen how much more pressing its subject matter would be in the nine years that followed. In 2011, the smartphone was in its infancy, iPads was less than a year old, and Instagram celebrated its first birthday with fewer than 10 million users. Fast-forward to 2020 and we have the latest Ofcom statistics showing that 45 per cent of 5–15 year olds own their own smartphone, and double that number have online access at home. With these devices enabling unfiltered access to images of 'perfect' bodies, perpetuated by the media, the rationale for this publication couldn't be any clearer.

The book itself is split into two main sections — the introductory chapters offer a clear explanation of body image, how it can affect children, and how it can be incorporated into the curriculum for children between 4–7 and 7–13. There is then a fully resourced scheme of work, including lesson plans with extension activities. The plans are refreshingly simple and easy to follow, a welcome contrast to other, more wordy PSCHE schemes I have come across. The age split of the lesson plans may be an issues for some teachers: 7 to 13 is a broad range, and in the UK this splits across two Key Stages.

A sensitive approach

Central to the approach included in this scheme of work is an avoidance of overt references to diets, junk food, or body appearance. Instead, it focuses on promoting emotional well-being and resilience. These are exactly the qualities children will need to deal with and process the stream of artificial perfection presented to them on their social media feeds. This is a sensitive and well-thought-through strategy — there is a substantial amount of research suggesting that tackling such issues head-on can make the problem worse.

While it may have been difficult to anticipate how much changes in technology and social habits would bring body image issues to the forefront, there is no doubt that this problem is now here to stay. **(HW)**
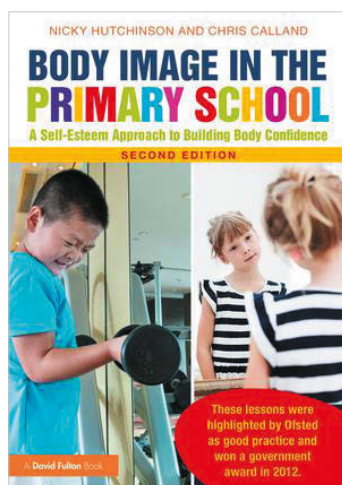
ESSENTIAL READING

Three books to help you explore machine learning further

DATA SCIENCE FROM SCRATCH

BY Joel Grus
PUBLISHER O'Reilly
PRICE £29.85
**ISBN** 978-1492041139
URL helloworld.cc/datascience

Learn how many of the most fundamental data science tools and algorithms work by implementing them from first principles. Learn the basics of linear algebra, statistics, and probability and understand how and when they're used in data science.
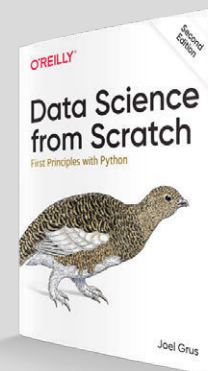
AI NARRATIVES

BY Stephen Cave (editor), Kanta Dihal (editor), Sarah Dillon (editor)
PUBLISHER Oxford University Press
PRICE £55
ISBN 0198846665
URL helloworld.cc/narratives

The potential social, ethical, political, and philosophical consequences of AI are explored in detail, with contributions from leading humanities and social science scholars.
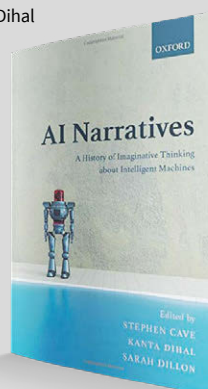
THE BOOK OF WHY

BY Judea Pearl and Dana Mackenzie
PUBLISHER Penguin
PRICE £10.99
ISBN 0141982411
URL helloworld.cc/why

Judea Pearl's accomplishments over the last 30 years have provided the theoretical basis for progress in artificial intelligence. Here, Pearl and science writer, Dana Mackenzie explore the saying 'Correlation does not imply causation.'

"HELLO, WORLD!"

Everything you need to know about our computing and digital making magazine for educators

Q WHAT IS HELLO WORLD?

A Hello World is a magazine for computing and digital making educators. Written by educators, for educators, the magazine is designed as a platform to help you find inspiration, share experiences, and learn from each other.

Q WHO MAKES HELLO WORLD?

A The magazine is a joint collaboration between its publisher, Raspberry Pi, and Computing at School (part of BCS, the Chartered Institute for IT).

Q WHY DID WE MAKE IT?

A There's growing momentum behind the idea of putting computing and digital making at the heart of modern education, and we feel there's a need to do more to connect with and support educators, both inside and outside the classroom.

Q WHEN IS IT AVAILABLE?

A Your 100-page magazine is available four times per year.
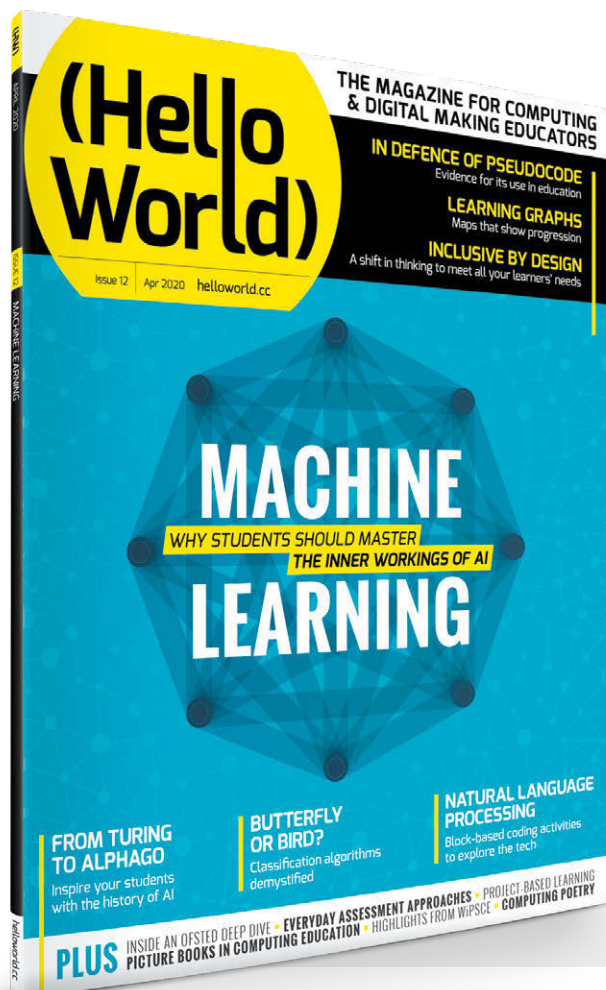
IT'S FREE!

Hello World is free now and forever as a Creative Commons PDF download. You can download every issue from **helloworld.cc**. Visit the site to see if you're entitled to a free print edition, too.

(Hello World)

THE MAGAZINE FOR COMPUTING & DIGITAL MAKING EDUCATORS

IN DEFENCE OF PSEUDOCODE
Evidence for its use in education

LEARNING GRAPHS
Maps that show progression

INCLUSIVE BY DESIGN
A shift in thinking to meet all your learners' needs

Issue 12   Apr 2020   helloworld.cc

MACHINE
WHY STUDENTS SHOULD MASTER THE INNER WORKINGS OF AI
LEARNING

FROM TURING TO ALPHAGO
Inspire your students with the history of AI

BUTTERFLY OR BIRD?
Classification algorithms demystified

NATURAL LANGUAGE PROCESSING
Block-based coding activities to explore the tech

PLUS INSIDE AN OFSTED DEEP DIVE • EVERYDAY ASSESSMENT APPROACHES • PROJECT-BASED LEARNING
PICTURE BOOKS IN COMPUTING EDUCATION • HIGHLIGHTS FROM WiPSCE • COMPUTING POETRY

WANT TO GET INVOLVED?

There are numerous ways for you to get involved with the magazine.
Here are just a handful of ideas to get you started

- **Give us feedback**
  Help us make your magazine better – your feedback is greatly appreciated.

- **Ask us a question**
  Do you have a question or a bugbear you'd like to share? We'll feature your thoughts and ideas.

- **Tell us your story**
  Have you had a recent success (or failure) you think the wider community would benefit from hearing about? We'd like to share it.

- **Write for the magazine**
  Do you have an interesting article idea or lesson plan? We'd love to hear from you.

GET IN TOUCH

Want to talk? You can reach us at:
contact@helloworld.cc

FIND US ONLINE

www.helloworld.cc

🐦 @HelloWorld_Edu

f fb.com/HelloWorldEduMag

SUBSCRIBE IN PRINT TODAY!
PAGES 34–35

(Hello World)

helloworld.cc