THE MAGAZINE FOR COMPUTING & DIGITAL MAKING EDUCATORS

TOMORROW'S ENGINEERS
The first Schools COP

PHILOSOPHY FOR CHILDREN
A fresh approach to digital literacy

INTERVIEW: CENTRE FOR AI & CLIMATE
AI for mitigating climate change

SUSTAINABILITY & COMPUTING

HOW CAN TECHNOLOGY BE GOOD FOR THE PLANET?

PLUS

CLOZE QUESTIONS • RESEARCH CENTRE LAUNCH • REPAIR CAFÉS • COOLEST PROJECTS
COSPACES • UPGRADE CULTURE • ONLINE SAFETY • COMPUTING QUALITY FRAMEWORK
BAKING IMPOSSIBLE • BROWSER-BASED CODE EXERCISES • AI & ENVIRONMENTAL RISKS

(HW)

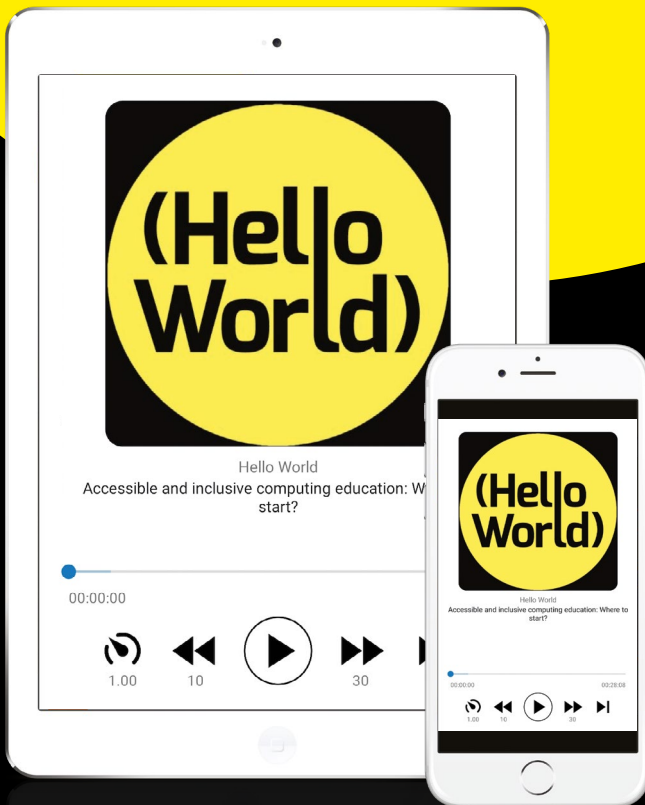# HEAR FROM THE WRITERS!

Listen to Hello World's podcast now:

- Get more great Hello World content between issues

- Hear directly from the educators behind our articles

- Listen on the move — delve a little deeper and have some fun along the way

Subscribe to the podcast!

- Apple Podcasts
  helloworld.cc/applepod

- Spotify
  helloworld.cc/spotifypod

- Google Podcasts
  helloworld.cc/googlepod

- Amazon Music
  helloworld.cc/amazonpod

SCAN ME

(Hello World)

Hello World
Accessible and inclusive computing education: W start?

00:00:00

1.00   10   30

(Hello World)

Hello World
Accessible and inclusive computing education: Where to start?

00:00:00   00:28:08

1.00   10   20

OUR MOST POPULAR EPISODES

How moral is your machine? Ethics in computing education

Accessible and inclusive computing education: where to start?

How can we get everyone excited about code?

HELLO, WORLD!

Over the last three years, we've seen more than 7 million students around the world put down their schoolwork and take to the streets to demand action on climate change. The School Strikes for Climate (or Fridays For Future) has been a platform for young people to express their deep concern for the planet they are inheriting and their growing frustration at political inaction.

Could technology be the answer? This issue explores the interaction between sustainability and computing, from how we can interact with technology responsibly to its potential to mitigate climate change. We delve into the research behind device-repair cafés (page 14) and see repair put into action by a student in Malaysia for his local community (page 24), as well as taking a deep dive into the question of technology obsolescence (page 36). We also share young people's ideas for how technology can be used for good, showcasing sustainability-themed student projects (page 34) and a school-based COP (page 30).

For all our US readers, please come and have a chat if you're going to be at CSTA's annual conference in Chicago next month! The Hello World team will be there to give out print copies of the magazine, and I will be hosting a writing workshop where educators will learn to write an engaging article — see page 11 for more details. As always, please get in touch with your thoughts or ideas about Hello World at **contact@helloworld.cc** and visit **helloworld.cc/ writeforus** if you have an idea for an article — we'd love to hear from you!

Gemma Coleman
Editor

FEATURED THIS ISSUE

HARRY & ANNA WAKE

Coding cousins Harry and Anna are students in the UK. They introduce their website teaching other young people to code on **page 72**.

MONIR EL MOUDDEN

Monir is a computer science teacher in London, UK. On **page 88**, he discusses integrating competitions into the curriculum.

SHANNAN PALMA

On **page 52**, Shannan, who works at HER Academy, shares how she created a television-inspired physical computing challenge.

(HW) CONTENTS

20-43

COVER FEATURES

SUSTAINABILITY
A series of articles exploring computing and sustainability

© Blue Planet Studio/stock.adobe.com

NEWS, FEATURES, AND OPINION

52

BAKING IMPOSSIBLE!
A physical computing and baking challenge

60

COSPACES
Introducing algorithms in VR

68

MENTORING
Developing a computing curriculum with a school in The Gambia

LEARNING

RESOURCES & LESSON PLANS

CONVERSATION

REVIEWS

SIGN LANGUAGE FOR COMPUTING TERMS

A team in Scotland, UK, has developed a British Sign Language glossary for computing terms that aims to widen tech access

Ronnie Kerr

Experts at The University of Edinburgh have helped produce a British Sign Language (BSL) glossary for deaf people engaged in digital technology. They are part of a team that has created more than 500 signs covering computer science, cybersecurity, data science, and software development. The glossary aims to help the deaf community in schools, colleges, universities, and workplaces access qualifications and careers in tech.

Teamwork

A UK-wide team of eight deaf people who are also tech experts spent eight months with sign linguists developing and testing the new signs for the glossary. Team member Ben Fletcher, principal engineer with the Financial Times, says it is important to create a common language for deaf people in tech: "I have studied and worked in computing throughout my whole life, but tech and BSL have often been a difficult combination. There's a huge list of computing terms, very few of which have dedicated and widely recognised signs, and others I just had to make up. It was very frustrating."

Before the glossary was launched, deaf people often had to spell out each individual letter of the specialised terms they were using. Popular tech words and phrases now covered include ethical hacking, firewall, data breach, machine learning, and phishing. Secondary pupil Billy-Jack Gerrard, who attends St Augustine's R.C. High School in Edinburgh, is deaf and wants to study AI and computer science at university: "The terms will make life so much easier and, in turn, be far more inclusive for deaf people like me who are wanting to pursue a digital career."

Skills gap

Head of Digital Technologies and Financial Services at Skills Development Scotland, Phil Ford, adds, "This will help deaf people get jobs in tech while also enhancing inclusivity — all with the ultimate aim of plugging the skills gap in a sector that is vital for Scotland's economy."

You can find the complete list of signs on the Scottish Sensory Centre website (**helloworld.cc/ssc**). Kate Farrell of the Data Education in Schools initiative is keen to keep adding to the list, and told us, "Like the technology itself, which is constantly changing, the accompanying language has to evolve — we therefore welcome continued input from technologists."

The new BSL lexicon is backed by government agency Skills Development Scotland and the Scottish Sensory Centre, based in Edinburgh. **(HW)**

> "THIS WILL HELP DEAF PEOPLE GET JOBS IN TECH, ENHANCING INCLUSIVITY AND HELPING TO PLUG THE SKILLS GAP

■ One of the newly created signs

Data Ethics
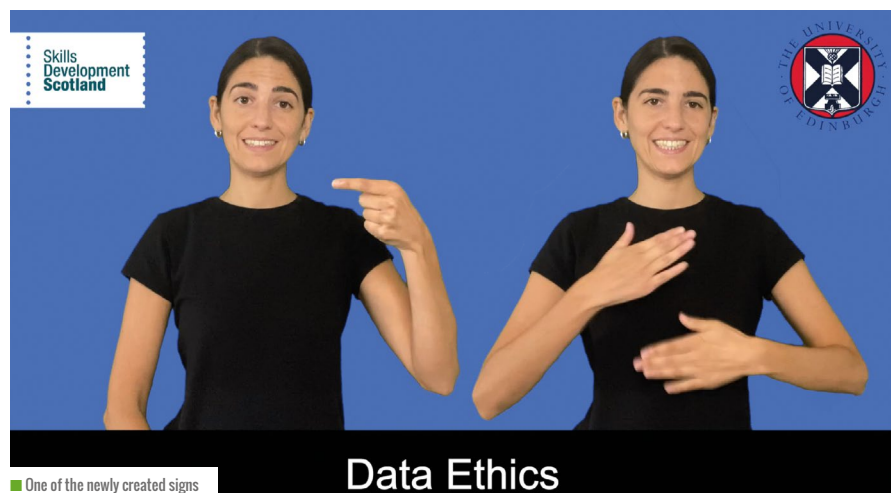
LAUNCHING THE RASPBERRY PI COMPUTING EDUCATION RESEARCH CENTRE

The creation of a new research centre represents a unique partnership between the University of Cambridge and the Raspberry Pi Foundation

Sue Sentance

Last July, the University of Cambridge and the Raspberry Pi Foundation announced the creation of a new research centre focusing on computing education research for young people in both formal and non-formal education (helloworld.cc/centreblog). This is an exciting venture and has the potential to represent an important step change for the field. Computing education research that focuses specifically on young people is relatively new, particularly in contrast to established disciplines such as mathematics and science education research. However, computing is now a mandatory subject in some countries, and is growing globally, so we need to investigate the learning and teaching of this subject rigorously, in conjunction with schools and teachers.

Developing the centre

We have been making progress towards establishing the centre over the last nine months or so. In October I was appointed director, and in December we were awarded funding by Google for a one-year research project on culturally relevant computing teaching, following on from a project at the Raspberry Pi Foundation. We are uniquely positioned in that the research team for the centre straddles both organisations. They complement each other well: the University of Cambridge is one of the highest-ranking universities in the world, and the Raspberry Pi Foundation has a compelling mission around empowering all young people with the computing skills and knowledge they need in today's world.

We are officially launching!

On 20 July 2022 we will be holding an in-person afternoon event in Cambridge, UK, to celebrate the establishment of this research centre, and to support our long-term goal of creating a community with fellow researchers, teachers, and other education practitioners. This is open to all, and will include opportunities to meet members of the research team and listen to a series of short talks to launch the centre formally — register at **helloworld.cc/centrelaunch**. We are thrilled that Professor Mark Guzdial, who many readers will be familiar with, will be cutting the ribbon!

Research and you

In the busy lives that teachers live, the word 'research' may conjure up an image of out-of-touch academics buried in books in stuffy libraries. Teachers will probably not have time to read long research articles or engage with new theories, which is one reason this magazine always includes some accessible insights into emerging research. In our research in the centre, we want to make sure that firstly, we work with teachers and schools directly when implementing and evaluating research projects, but also that we publish results in a number of different formats as promptly as we can, without a paywall. We look forward to working with a large community of teachers; even if you can't attend the launch, do sign up to be part of our mailing list at **computingeducationresearch.org**. (HW)

■ The new research centre will sit in the Department of Computer Science and Technology at the University of Cambridge

INSPIRING YOUNG GIRLS

A new initiative from BootUp PD aims to inspire all girls to see themselves
in STEM roles through a monthly video series, For Girls By Girls

Karen Melara

What does it take to get young girls into tech? This was the burning question that BootUp PD (**bootuppd.org**) and the Color of Our Future committee members had when discussing gender and racial inequities in computer science education. Research shows that disparities in how young girls view STEM appear as young as six. Girls are more inclined to pick up on societal stereotypes than boys and become convinced that certain activities are not suitable for them (**helloworld.cc/ bian2017**). Spotlighting young girls doing remarkable things in computer science and discussing the intersectionality of computer science with design, fashion, music, and art, seemed like the way to show girls what's possible with a computing education. And just like that, BootUp PD launched the monthly video series *For Girls By Girls* (4GXG) in January 2022 (**helloworld.cc/4GXG**).

4GXG aims to lift young girls' voices to demonstrate that computer science and tech are accessible to them, no matter who they are, where they come from, or what their passions are. Our young hosts, 14-year-old Nyla and 11-year-old Tera, hope to inspire young girls to explore the possibilities of tech and ways to get started with code. "I feel like being a host has been a great experience so far! In the beginning, I was so nervous. Coding has taught me how to solve difficult problems and break them down. My sister and I have grown tremendously. Meeting all these new girls who are doing amazing things to make a change is awesome!" says Tera.

Inspiring conversations

Since the series launched, Nyla and Tera have hosted incredible girls and young women, connecting over topics such as the importance of code, cybersecurity, and inspirational women, and building a sense of community and safety online. The conversations so far have ranged from three young siblings being exposed to computing through coding camps and their mum (**helloworld.cc/siblings**) to one young woman's experience of starting a non-profit to inspire and empower girls just like them in STEM fields (**helloworld.cc/varija**).

Whether a student becomes a data scientist or a designer, the skills gained from a computer science education, like computational thinking and problem-solving, transcend subject areas and passions — and that's something that co-host Nyla has taken away from her conversations so far: "Being a host for 4GXG has really been a fun experience! The beginning was a little rocky. I learnt to get comfortable the more I interviewed girls. One way that coding helped me in my studies is with music. I learnt to code different sounds with notes and see how they fit together. But I continue to learn more about code after meeting these amazing girls around the world!"

4GXG are change-makers tearing down barriers to computer science education and refusing to let preconceived notions stop young girls from accomplishing their goals. Through the power of storytelling, these tech champions with mixed experiences and diverse backgrounds are thrilled to help other girls fight for gender equality and representation in tech. Do you know a young girl who has a tech story to share with Nyla and Tera? Send your suggestions to Karen Melara at **karen@bootuppd.org**. Learn more about *For Girls By Girls*, generously supported by the National Center for Women in Technology Aspirations in Computing (**aspirations.org**), at **helloworld. cc/4GXG** and follow on Instagram at **@4GXG_BU**, **#4GXGHeroes**. (HW)

■ Tera and Nyla preparing for a *For Girls By Girls* interview to discuss all things coding and beyond

BA EDUCATIONAL TECHNOLOGY: READY, SET, GO!

The University of Winchester, UK, has launched a new
degree combining education and computing

Caroline Stockman

The University of Winchester in the UK is excited to announce that they now offer a BA in Educational Technology, combining the study of education with computing. The premise is that confident technical know-how must go hand in hand with strong ethics and a broad view of the importance of education in society. It keeps career routes into teaching, IT support, or the edtech industry wide open.

A unique course

In the first year of the course, students learn about the foundational principles of educational theory from sociological, historical, and political angles. They study security threats and the related operational impact, and look at how to implement and maintain protection mechanisms. In software development, students design, write, and test their own programs, and write technical and/or user documentation. This sits alongside database design and management, and a broader understanding of digital business and strategy.

In the second year, students continue developing their knowledge and skills in cybersecurity and risk management, data analytics, and software testing, with an option to pursue an interest in artificial intelligence. An integrated project allows students to showcase these skills in a sector-relevant way. Broader theory on the social and political impact of education paves the way for the ethical and human-centred adoption of these technical skills in an educational context.

In the third year, students can complete a dissertation or computing project of their own choice, alongside modules on strategic innovation, cloud computing, and intellectual property and data rights. There is also a variety of optional modules to develop specialised interests, such as early childhood or special needs education, film as education, and education for refugees.

> " I GAINED SO MUCH CONFIDENCE IN MY RESEARCH AND WRITING ABILITY, WHICH HAS BEEN KEY TO MY SUCCESS

ALUMNA STORY: JOSEPHINE COOPER

"When choosing the Technology in Education module [before the full degree was available], I did so thinking purely about enjoyment, without realising it would help with my career - but I am now working as an account manager at a software company! The mix of contemporary technologies and theorists gave such an interesting insight into the development of technology, and gave more context to where we are today. Having theorists that were slightly more challenging to read meant I gained so much more confidence in my research and writing ability, which was key to my success. I only wish the full degree had been an option when I started! I think so many students will gain so much from this, whether they go into teaching or not."

The degree will be equally valuable to students who are looking to become computing teachers or pursue a wide variety of other careers, such as school IT support, learning technology, software development, e-learning training, product management, risk assessment, and cybersecurity.

Society needs more technically skilful people with a strong understanding of safeguarding, special needs, pastoral care, and the many other facets that make up the complex realities of learning. Drop by our next open day, or visit **helloworld.cc/ winchester**, to find out more! *Indicative course content, changes may occur. Now open for 2023 applications via UCAS.* (HW)

CAS LEADERSHIP COACHING ACADEMY SCALES UP

A coaching academy for UK-based computing educators is expanding its programme, offering more opportunities to develop coaching and leadership skills

Pete Marshman

The Computing at School (CAS) Leadership Coaching Academy is a coaching professional development programme for computing educators. Certified by BCS, The Chartered Institute for IT, the programme includes training to help educators develop the coaching and leadership skills they need to take on senior leadership roles. The Academy launched in autumn 2020, training 49 coaches, and will be expanding its offering later this year.

Why take part?

Educators can either participate as a coach or volunteer as a coachee. Coaches receive a developmental coaching journey provided by the Academy's executive coaches Les Duggan, Yota Dimitriadi, and Derek Peaple. They are trained, over roughly 17 hours, to become effective coaches and develop the leadership skills relevant to senior roles. The training includes, as part of an introductory module, exploring the rationale for coaching; the powers of effective questioning; coaching intervention; and developing a clear action plan to ensure growth.

After this, participants unpick coaching performance through the study of emotional intelligence; the use of feedback to unlock intrinsic motivation; and coaching as an effective leadership style. Donna Robertson, who completed the programme last year, outlined her experience: "Since starting, I have grown in confidence and now reflect more on my own practice. I feel I have grown and developed as a leader at school and improved my interpersonal skills to develop computer science within the school and with my colleagues."

Coachees do not receive a certificate, but will be coached to develop their leadership skills and teaching performance by the participating coaches. They are guided to reflect on their achievements and define their own journey to becoming outstanding in their vast and varied roles, challenging their thinking and changing their behaviour.

it also provides tier one coaches with experienced role models. It gives these newer coaches a chance to hear tier two coaches' experiences through coaching surgeries, which each have a key focus such as well-being or questioning.

Get involved

The programme is free of charge, and is currently being held online for UK-based computing educators at primary and secondary levels, and those in further and higher education.

> ## COACHES WILL NOW HAVE THE CHANCE TO LEARN FROM THE EXPERIENCES OF ROLE MODELS THROUGH COACHING SURGERIES

Evolving the Academy

Following the success of the Academy in its first year, it has expanded to provide two sequential development pathways and develop tier one and tier two coaches. After the first year of training and certification (tier one), coaches can enrol on the tier two programme, which covers three themes:

- Leading yourself
- Leading others (teams)
- Leading organisations (schools)

Not only does this further extend participants' practice as coaches, but

The coachee role is only available to teachers who have completed the Computer Science Accelerator Programme, BCS scholars, and those from industry supporting students between the ages of 16 and 24 from less privileged backgrounds who wish to pursue study in a digital or IT-related subject.

Teachers and senior leaders wishing to consider the culture of coaching in their institution can work through a coaching culture audit questionnaire at **helloworld. cc/coachsurvey**. To find out more about the Academy and register your interest, visit **helloworld.cc/cascoach**. (HW)

GOING TO CSTA'S ANNUAL CONFERENCE?

Hello World and the Raspberry Pi Foundation will be at CSTA's annual conference in Chicago and we'd love to meet you. Join one of our sessions! Read more at **helloworld.cc/cstaagenda**:

- **Thursday 14 July, 9 am-12 pm:** 'Semantic Waves and Wavy Lessons: Connecting Theory to Practical Activities and Back Again' with James Robinson and Jane Waite

- **Friday 15 July, 1-2 pm:** 'Write for a Global Computing Community with Hello World Magazine' with Gemma Coleman

- **Saturday 16 July, 4-5 pm:** Keynote — 'Developing a Toolkit for Teaching Computer Science in School' with Sue Sentance

- **Sunday 17 July, 9-10 am:** 'Exploring the Hello World Big Book of Computing Pedagogy' with Sue Sentance

Visit us at Booth 521 for free magazines and more!

GETTING PHYSICAL WITH PROGRAMMING

Physical computing devices can engage young people with special educational needs and disabilities and help them make sense of programming concepts

As described in the 'Get hands-on' section of Hello World's *The Big Book of Computing Pedagogy*, physical computing is "a broad term to describe activities in which learners write programs to interact with the real world using specialist hardware" (helloworld.cc/ bigbook). At a basic level, this may mean programming a floor robot to move, while more complex devices may have a number of inputs and outputs that can be used to record and interact with the environment, such as sensors, buzzers, and accelerometers. There is a huge variety of physical computing devices currently available to schools, and they seem to be very engaging for a wide range of young people. So, what are the benefits and barriers of using them with learners with additional needs and disabilities, and which are the best ones to use?

Benefits

Physical computing can be very engaging for students, and this in itself is a reason to do it, as it may be the best way to teach some complex programming concepts to our harder-to-reach learners. Children will naturally pick up and tinker with robotic toys, and these are far more accessible to get started with than a screen-based programming language that may require students to log in, open software, and use a keyboard or mouse. A floor robot with buttons can be used instantly, with immediate outcomes. These quick wins

Catherine Elliott is the SEND lead for the Sheffield eLearning Service (**sheffieldclc.net**), and since the announcement of the computing curriculum in 2013 she has been working on ways to make the subject accessible for all learners. She is a member of the CAS Include working group, and leads the SEND Virtual and the Sheffield and South Yorkshire Secondary CAS Communities (**@catherinelliott**).

are also useful for building confidence, so starting small and achievable is always important.

Part of the appeal of physical computing devices is their sensory nature. Young people can interact with them physically through touch, sight, and sound. This variety is also essential for including young people with sensory difficulties such as visual or hearing impairments. For example, educators can check out Code Jumper (**codejumper.com**), an example of a physical programming language that can be used by visually impaired learners to learn to code (see page 22 of Hello World issue 11 for more information).

Seymour Papert referred to the original floor turtles as "objects-to-think-with" (**helloworld.cc/papert1980**), and further evidence is emerging that indicates that the tangible nature of physical computing supports learners' program comprehension (see page 130 of *The Big Book of Computing Pedagogy* at **helloworld.cc/bigbook**). A survey of teachers working in special schools and similar settings in 2018 certainly reflected this: almost half of the respondents mentioned physical computing as a successful strategy for teaching programming (**helloworld.cc/elliott2018**).

Barriers

There are, however, a number of barriers to consider when working with physical computing devices. For teachers,

■ Students with visual impairments who participated in a beta test of the technology behind Code Jumper

classroom management can be more challenging, and starting off by modelling concepts in an unplugged way will help retain focus. Staff confidence with more complex devices can also be an issue.

Devices like the Crumble and Raspberry Pi, which are used with other peripherals such as sensors and motors, will require students to check and debug both the program and the physical system, which may lead to frustration when things don't work. Young people with physical impairments may find manipulating small components demanding, and the collaborative nature of the devices will need to be carefully managed with students who find pair and group work challenging.

What works well with SEND learners?

There is an increasing number of physical computing devices available for young people to learn about programming, from Bee-Bots to Ozobots, and from micro:bits to Crumbles. Each device has its limitations, and choosing carefully will make a difference for your learners.

For learners starting out with programming, floor robots are great, and the ubiquitous Bee-Bot is robust and accessible (**helloworld.cc/beebot**). The new TTS Light Up Glow and Go Bot (**helloworld.cc/lightup**) is a good precursor to this, with much simpler controls (for example, pressing an arrow moves it in that direction). The Code-a-Pillar is perfect for teaching about sequence, as each segment of the caterpillar carries a different command.

For pupils working at these early levels, the clarity of the buttons or programming interface is key — and we can support this with coding cards and the physical modelling of any movement.

Students starting to work with repetition, selection, and variables have a number of options. Robots are engaging and easy to get started with, but they can be expensive. Microcontrollers, such as micro:bit, Crumble, and CodeBug, can be a cheap alternative. They are accessible devices, and there is an immediate joy in making the LEDs light up. For some learners, the all-in-one nature of the micro:bit, which has built-in touch and sound sensors, speaker, compass, and so on, makes it easier to use, as there are fewer components to manipulate and debug.

As with any programming activities, good pedagogy is key. Scaffolding work with code using PRIMM, Parson's Problems, and targeted debugging projects will all help learners build up their knowledge of key concepts and allow them to engage with the code with a degree of security. This variety of activities also reflects the aims of CAST's Universal Design for Learning (UDL) framework (**helloworld.cc/cast**) to provide learners with myriad methods of engagement, action, and expression.

Despite the barriers, it is clearly worthwhile persevering with physical computing activities. Introducing physical computing allows the diverse strengths and interests of our young people to be developed in different ways, which is truly at the heart of the inclusive classroom. (HW)

#INSIGHTS

REPAIR CAFÉS IN COMPUTING EDUCATION

STORY BY Katharine Childs

Many technology items are disposed of each year, either because they are broken, are no longer needed, or have been upgraded. Researchers from Germany have identified this as an opportunity to develop a scheme of work for computing, while at the same time highlighting the importance of sustainability in hardware and software use. They hypothesised that by repairing defective devices, students would come to understand better how these devices work, and therefore meet some of the goals of their curriculum.

The research team visited three schools in Germany to deliver computing lessons based around the concept of a repair café, where defective items are repaired or restored rather than thrown away. This idea was translated into a series of lessons about using and repairing smartphones. Learners first of all explored the materials used in smartphones and reflected on their personal use of these devices. They then spent time moving around three repair workstations, examining broken smartphones and looking at how they could be repaired or repurposed. Finally, learners reflected on their own ecological footprint and what they had learnt about digital hardware and software.

An educational repair café

In the classroom, repair workstations were set up for three different categories of activity: fixing cable breaks, fixing display breaks, and tinkering to upcycle devices. Each workstation had a mentor to support learners in investigating faults themselves by using the question prompt, "Why isn't this feature or device working?" At the display breaks and cable breaks workstations, a mentor was on hand to provide guidance with further questions about the hardware and software used to make the smartphone work. On the other hand, the tinkering workstation offered a more open-ended approach, asking learners to think about how a smartphone could be upcycled to be used for a different purpose, such as a bicycle computer. It was interesting to note that students visited each of the three workstations equally.

The feedback from the participants showed there had been a positive impact in prompting learners to think about the sustainability of their smartphone use. Working with items that were already broken also gave them confidence to explore how to repair the technology. This is a different type of experience from other computing lessons, in which devices such as laptops or tablets are provided and are expected to be carefully looked after. The researchers also asked learners to complete a questionnaire two weeks after the lessons, and this showed that 10 of the 67 participants had gone on to repair another smartphone after taking part in the lessons.

Links to computing education

The project drew on a theory called duality reconstruction that has been developed by a researcher called Carsten Schulte (**helloworld.cc/schulte2008**). This theory argues that in computing education, it is equally important to teach learners about the function of a digital device as about the structure. For example, in the repair café lessons, learners discovered more about the role that smartphones play in society,

REPAIR CAFÉ START-UP TIPS

- Connect with a network of repair cafés in your region; a great place to start is **repaircafe.org**

- Ask for volunteers from your local community to act as mentors

- Use video tutorials to learn about common faults and how to fix them

- Value upcycling as much as repair - both lead to more sustainable uses of digital devices!

- Look for opportunities to solve problems in groups and promote teamwork

■ Learners examined broken smartphones and investigated how they could be repaired or repurposed

as well as experimenting with broken smartphones to find out how they work. This brought a socio-technical perspective to the lessons that helped make the interaction between the technology and society more visible.

Using this approach in the computing classroom may seem counter-intuitive when compared to the approach of splitting the curriculum into topics and teaching each topic sequentially. However, the findings from this project suggest that learners understand better how smartphones work when they also think about how they are manufactured and used. Including societal implications of computing can provide learners with useful contexts about how computing is used in real-world problem-solving, and can also help to increase learners' motivation for studying the subject.

Working together

The final aspect of this research project looked at collaborative problem-solving. The lessons were structured to include time for group work and group discussion, to acknowledge and leverage the range

of experiences among learners. At the workstations, learners formed small groups to carry out repairs. The paper doesn't mention whether these groups were self-selecting or assigned, but the researchers did carry out observations of group behaviours in order to evaluate whether the collaboration was effective.

FURTHER READING

✔ Schmidt, A. K., & Schulte, C. (2019). Das RetiBNE Café. *Informatik für alle.* helloworld.cc/issue19insights1 (in German)

> ❝ 10 OF THE 67 PARTICIPANTS WENT ON TO REPAIR ANOTHER SMARTPHONE AFTER TAKING PART IN THE LESSONS

In the findings, the ideal group size for the repair workstation activity was either two or three learners working together. The researchers noticed that in groups of four or more learners, at least one learner would become disinterested and disengaged. Some groups were also observed taking part in work that wasn't related to the task, and although no further details are given about the nature of this, it is possible that the groups became distracted. Further investigation into effective pedagogies to

set group size expectations and maintain task focus would be helpful to make sure the lessons met their learning objectives.

This research was conducted as a case study in a small number of schools, and the results indicate that this approach may be more widely helpful. If you're thinking about setting up a repair café in your school to promote sustainable computing, either as a formal or informal learning activity, look at the start-up tips in the boxout for ideas on where to begin. (HW)

© iVazoUSky/stock.adobe.com

■ Pupils are increasingly eager to discuss climate-related issues in their classes

TEACHERS' VIEWS ON CLIMATE CHANGE EDUCATION

STORY BY Bonnie Sheppard

Fridays for Future, Extinction Rebellion, the fossil fuel crisis — climate change and sustainability are hot topics that increasingly affect and interest students of all ages. A recent study of teachers in England showed that nearly all teachers had noticed an increased eagerness to discuss climate-related issues in their classes, especially in primary education. However, the national

curricula of most countries confine climate change education to science and geography lessons in secondary school. Teachers of other subjects and those with less training in climate science therefore often feel ill-equipped to address their students' questions or incorporate climate- and sustainability-related examples into their learning modules. The study also suggests that empowering

teachers across all subjects to deliver climate education could solve many shortcomings of the current approach, and also improve the dialogue about climate change and sustainability.

Earlier, cross-curricular, better funded

If there is one key takeaway message from research into climate change education in the past few years, it's that there is currently

" THERE IS NO CURRICULUM THAT GIVES PUPILS AN OVERVIEW OF THE EFFECTS OF CLIMATE CHANGE ON SOCIETY

no curriculum that gives students a broad overview of the effects of climate change on society. In general, more emphasis is put on factual knowledge, rather than encouraging faceted discussions and productive actions. For instance, the national curriculum in England currently introduces climate change topics only in secondary school, and focuses on climatic processes and low-level activities, such as cycling to school. Considering the far-reaching effects of climate change on world economies, politics, and society, this approach limits the potential for engagement and impactful activities.

When British researchers asked teachers about their preferences for climate change education, they were surprised at the strong consensus about the delivery of climate-related teaching and learning. Primary-school teachers commonly favoured the introduction of the topic to pupils as young as five to seven, with the inclusion of both science-related topics and conservation projects in the curriculum. For pupils aged eight to fourteen, the focus could shift to science, wider societal impacts, social justice, and action-based sustainability activities such as private and public advocacy and local campaigning. To achieve such a broad delivery of climate education, the polled teachers favoured the inclusion of these topics and discussions across all subjects of the curriculum, while a third of teachers suggested integrating climate change education into computing classes.

Facilitating cross-curricular teaching is closely linked to teachers feeling confident and well-resourced. Teachers in England have therefore placed climate change in the top three priorities for additional funding, directly after literacy and STEM education. In the meantime, educators can still incorporate relevant activities and discussions about sustainability into the computing classroom. Real-world examples could not only raise their motivation for computing education in general, but also enable students to understand the interconnectedness of climate change. **(HW)**

ACTIVITIES TO TRY IN YOUR CLASSROOM

■ When discussing how to use technology responsibly, introduce your students to the concept of green computing. You could start with Beverly Clarke's article about green information technology in issue 17 of Hello World (**helloworld.cc/greenit**) or Neil Gordon's article on page 22 of this issue, and develop ideas to make computing more sustainable at your school.

■ Collaborate with teachers of other subjects to organise action-based projects for Earth Day (**earthday.org**).

■ Look for free climate-based exercises online. For example, **steamlabs.ca** offers materials for a lesson on green energy that uses AI and machine learning, and Microsoft FarmBeats for Students lesson plans could be used to explore the influences of varying climates on food production (**helloworld.cc/ farmbeats**).

■ There are several Scratch-based projects from the Raspberry Pi Foundation that focus on protecting the environment for future generations (**helloworld. cc/scratchplanet**).

©gurldeyskaya/stock.adobe.com

■ Teachers in England placed climate change in the top three priorities for additional funding

In each issue, the CAS Assessment Working Group share their current activities, findings, or reports relating to computing assessment.

In this column, their first for Hello World, they share one member's journey of creating self-marking Python resources and outline the pedagogical approach behind it.

SELF-MARKING PYTHON RESOURCES

In their first column, the **CAS Assessment Working Group** share one member's self-marking Python resources to improve student feedback

ABOUT THE CAS ASSESSMENT WORKING GROUP

The Computing at School (CAS) Assessment Working Group is a CAS community. It aims to provide clear statements of informed opinions relating to current issues of formative and summative assessment in the teaching of computing, reflecting academic and professional understanding. The group presents the teacher's voice but is informed by colleagues in awarding organisations, commerce, and academia (**helloworld.cc/casawg**).

Marking is challenging. It comes in big batches, and it usually has to be done very quickly. After marking 20 pupils' work, it can become tedious or tiring. Some assessments, such as mock exams, do sadly need to be marked by hand, but the biggest frustration is when assessment becomes a repetitive administrative task. We might tell our students that computers were invented to save humans time and effort, and if they find themselves getting frustrated by doing the same thing more than once, there's probably a better way of working.

For the last few years, Pete Dring, head of computing at a secondary academy in York, UK, has been developing self-marking Python challenges, funded by Google's Educator PD grant (**helloworld.cc/googlegrant**). The project aims to help boost students' confidence, resilience, and independence when understanding, writing, and debugging code. It's also designed to reduce teacher workload — an automated tool generating individual feedback allows students to make informed actions to address gaps in their knowledge more effectively than they would if they had to wait for their teacher to mark their work. Teacher time can then shift from marking to helping students understand and act on the feedback effectively.

Self-marking challenges

After working with his students to design and test early integrated development environment (IDE) prototypes (for example, **helloworld.cc/peteprototype**), he developed **create.withcode.uk**. This free site allows beginners to work through different Python concepts and skills and gives students instant feedback as they adapt code that works, debug code injected with common mistakes, and apply their knowledge to create their own solutions.

These self-marking challenges aim to stretch lower-secondary students or support upper-secondary students as they get started with text-based programming (**helloworld.cc/selfmarking**). They were originally designed as independent learning activities to complement explicit instruction in lessons, but lockdowns emphasised the need for resources that pupils could work through at home, while still demonstrating progress to their teacher.

The online Python environment used by **create.withcode.uk** offers a subset of the full features available from an offline IDE, but it's designed to be simple enough for beginners to use, and has the advantage of allowing students to access code they wrote in a previous lesson, even if they didn't save it.

Pedagogy in practice

Learning to program can present a number of barriers. Some students struggle to type, and problem-solving with algorithms and code is hard when you're still prodding the keyboard with individual fingers. Some students struggle to understand code, and all the error messages and online tutorials require you to learn a whole new lexicon that is beyond the vocabulary of most beginner programmers. Other students lack the confidence to make mistakes; debugging is hard if you're terrified that you're going to break something. Finally, some students lack motivation. Copying out
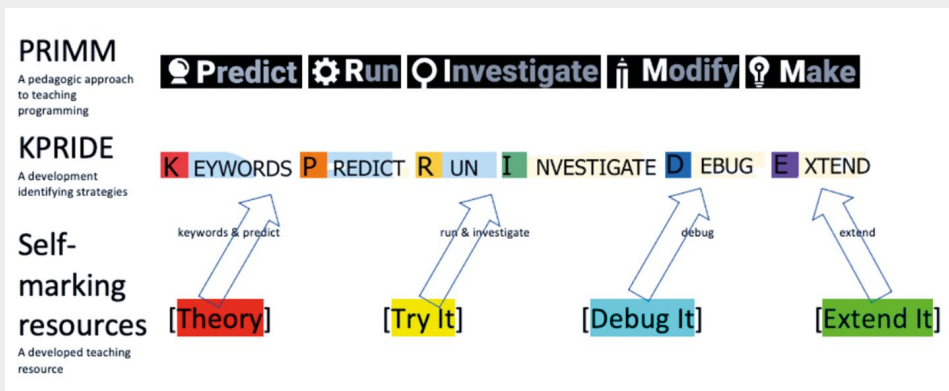
Figure 1: The resources build on the PRIMM model with the addition of KPRIDE

code that someone else has written, to solve a problem that someone else has defined, isn't a particularly creative process.

It's good to see more and more pedagogical research into strategies that empower teachers to tackle each of these programming comprehension blockages. The approach taken by Pete Dring for the self-marking challenges builds on Sue Sentance's PRIMM model (**helloworld.cc/primm**) but adds in some additional support with the structure of Keywords–Predict–Run–Investigate–Debug–Extend, or KPRIDE (**helloworld.cc/kpride**).

Each resource is split into four sections: Theory, Try It, Debug It, and Extend It (see **Figure 1**). The Theory section aims to introduce one new concept at a time in a logical, sequential way. Try It activities give students some working code that they can use as a simple reference, and include some self-marking challenges to encourage tinkering and experimentation. Debug It activities give students code that contains common mistakes so that they can build up confidence and resilience when faced with error messages. Extend It activities encourage creativity with open-ended challenges to apply the skills that students have learnt.

Motivation and quick wins

At the start of the first lockdown, Pete collaborated with local schools to set up a free weekly coding challenge that allowed students to access short live-coding support videos and interactive online resources that they could complete at home. **Live.withcode.uk** also tackled the typing skills deficit identified earlier

by challenging students to type out the code shown in the live-coding videos as accurately and as quickly as they could.

Although these weekly challenges aren't running at the moment, you can still give the existing challenges to your classes as starter activities or optional extra homework activities. Differentiation is a huge challenge when teaching programming, but any student can

> **AUTOMATED INDIVIDUAL FEEDBACK ALLOWS STUDENTS TO ADDRESS GAPS IN THEIR KNOWLEDGE MORE EFFECTIVELY**

engage with a type race, most can learn to master the keyword identification activities, and the self-marking challenges can keep students busy in those precious moments at the start of a lesson.

There is a wide variety of free and paid platforms that offer auto-grading for quizzes or code. Anything that reduces marking workload while increasing the quality and quantity of student feedback is a win. The challenge can be using all the assessment data to enable students to celebrate their successes and know where (and how) to concentrate their efforts to improve. Pete reflects, "I'm privileged to have worked on these resources with brilliant colleagues and excellent students. I hope that these resources save other teachers some time and support students to break through that frustration barrier to enjoy creative problem-solving with code." **(HW)**

TOP TIP

A top tip for students getting started is to run their code regularly by pressing Ctrl + Enter to run the whole program, or Ctrl + Full Stop to run a line at a time and see what each variable stores.

THE CENTRE FOR AI & CLIMATE

"RESPONSIBLE AI IS ON EVERYONE"

Gemma Coleman speaks to Pete Clutton-Brock and Sims Witherspoon, the co-founders of the Centre for AI & Climate, about AI's potential to accelerate climate action

"I t's certainly not a silver bullet … But if we really put our effort behind this technology and apply it to a wider range of potential use cases, then it could really accelerate the transition to net zero." It was over a series of coffees and conversations like this that Pete Clutton-Brock, Sims Witherspoon, and their co-founders formed the Centre for AI & Climate (c-ai-c.org), bringing together experts from different industries, all with a passion for protecting the planet. "If there was a question around how academics and researchers were thinking about this, we could pull in [our colleagues] Jack or Aidan," Sims explains. "From the technology and

business side, myself and others would weigh in, and Pete has a policy background — we saw the utility of having that group conversation where people were caring about the same problem from different perspectives, and able to work towards ideas with collaborative solutions."

Climate change and rapid technological transformation are some of the twenty-first century's most pressing issues, with artificial intelligence (AI) playing a potentially vital role connecting the two. So, what specific climate change challenges can AI address? Sims pauses and takes a deep breath. "Well, there are many, across various domains. AI can help us to solve challenges in energy, forestry and land use, industry, agriculture … the list is long." Ultimately, to address climate change, we have to reduce emissions to zero across the whole of the economy. Pete adds, "The power sector, the transport sector, heavy industry and buildings … these are the core sectors that are really critical. And there's a whole ton of examples where AI is already being useful to help those sectors and systems transition."

AI's power to accelerate climate action lies in its ability to ingest huge amounts of data, to optimise complex

CENTRE FOR AI & CLIMATE

■ The Centre for AI & Climate brings together experts from different industries, all with a passion for protecting the planet

systems, and to improve forecasting. "AI can be really helpful in terms of distilling data into useful insights," Pete outlines. "That can be collecting and exploring satellite data identifying areas at risk of illegal logging, or data looking at installations that are emitting methane when they shouldn't be."

The duo point to a number of current-use cases that they shared in a recent report with the Global Partnership on Artificial Intelligence, highlighting that this isn't just a hope for the future (**helloworld.cc/ aireport**). The UK National Grid, for example, already uses AI to optimise electricity demand forecasts and renewable supply. Aionics, a US-based start-up, uses machine learning to cut down on the research and development stage of building batteries, which can be used for electric cars. And Kuzi, a tool developed by a Kenyan company, aggregates various data and uses AI to make predictions about locust breeding locations and migration routes, to mitigate potentially devastating locust outbreaks.

The potential for AI to support action on climate change is clearly high, but it's a powerful tool, so it's vital to ensure it's being used in the right way. "This is a really complex space and even with the best of intentions, applications can sometimes drive suboptimal outcomes," Sims warns. "We must remember that AI is domain-agnostic — it's just a technology, so it can be used in oil and gas exploration and extraction, or other applications that are not good for the environment. There are also the knock-on effects of AI — autonomous vehicles, for instance, offer efficiency gains, but maybe also bring more drivers onto the roads, so more emissions. We must understand the many implications of what we build and make responsible choices."

Responsibility is the key word here, and an important role of the Centre is bringing together people with different expertise and facilitating discussions. Sims explains, "Having that collaboration and ongoing conversation allows

us to drive towards the more climate-positive outcomes [of AI], and to make folks aware of the applications that do run counter to climate goals." Pete adds, "Responsible AI is on everyone — it's something that everyone needs to be thinking about, whether that's the data scientists developing tools, or the governments who are setting the frameworks and regulatory environments in which these tools operate, or the businesses and companies that use this technology in their operations."

As time goes on, the need to address climate change is becoming more urgent, and as Pete puts it, "There's going to be increasing tailwinds for policymakers, companies and everyone, really, to use whatever tools we have in our toolbox to accelerate the transition [to net zero] as fast as we can." AI-readiness is the next big task on the Centre's agenda, and they'll be working with the Global Partnership on Artificial Intelligence again to educate the sectors vital in the fight against climate change (energy, transport, agriculture, and so on) to highlight what's possible with AI and what's needed to become AI-ready. "These cross-domain conversations

> ## " AI CAN ACCELERATE THE TRANSITION TO NET ZERO

are so important," Sims stresses. "There's a lot of curiosity within those industries, so we need to be able to communicate the potential effectively."

Ensuring that the right skills are being developed in schools, then, is going to be of the utmost importance. "The more we are able to upskill in cross-disciplinary education, the better off we'll be in the future. One of the gaps we have right now is people who have deep domain knowledge as well as machine learning or computer science backgrounds," Sims explains. "If you're able to understand the domain problem and also help the domain experts translate that into computational language that algorithms understand … that is going to be such a vital skill. If students are interested in this space, though, get involved now — they shouldn't feel it's something they should wait until they've got a PhD to get stuck into!" It may not be a silver bullet, but it's clear that there will be plenty more coffees and conversations to come about this powerful tool. (HW)

For more information about the Centre for AI & Climate, visit **c-ai-c.org**.

COMPUTING AND SUSTAINABLE DEVELOPMENT

Neil Gordon offers an overview of the positive and negative impacts of computing, linked to the United Nation's Sustainable Development Goals

As a computing educator, you will understand the potential of computing to change the world for good. But do we always tell students? In this article, we will explore why green computing is important and look at the negative and positive impacts of computing, as well as ways of exploring these themes with students.

Green computing – why bother?

You may be familiar with the idea of sustainable development, "development that meets the needs of the present without compromising the ability of future generations to meet their own needs" (helloworld.cc/brundtland1987). Shortly after the turn of the millennium, education was recognised as a way of ensuring that the next generation understood sustainable development and its importance. In 2015, the United Nations (UN) turned sustainability concepts into 17 specific Sustainable Development Goals (SDGs), including Climate Action, Sustainable Cities and Communities, Responsible Consumption and Production, and more.

As computer scientists, we need to recognise our duty and our role in helping. Green, sustainable computing is about understanding the impact that computing has, and making sure that the positive contributions outweigh any negative ones. There are other benefits to this than only moral ones. For example, effective solutions can also be better for us commercially and in education, and looking at the social context and applications can broaden interest in computing. This, in turn, can help to address the imbalance in the profile of students choosing to study computing.

The negative impact

Calculations around computing's contribution to the global carbon footprint vary, but the contribution is typically estimated to be between 5 and 12 percent, which is roughly equivalent to that of the aviation industry. Every time we turn a computer on, and every time we run software, we are adding to the demand for energy. A large proportion of that energy comes from fossil fuels, and even if you live in a country that is striving to be carbon neutral, it is likely that activities such as online searches are run on cloud services in other countries, where fossil fuels may be the primary energy source. The continual cycle of replacing devices also means we are creating demand for manufacturing and transport, and creating piles of e-waste.

As educators, we should demonstrate the sort of behaviours and approaches that help reduce computing-related waste and pollution, and help improve others' lives and society more widely. Some examples of this, linked to Goal 11 (Sustainable Cities and Communities) and Goal 12 (Responsible Consumption and Production), are:

- **Reduce energy consumption:** introduce students to the vampire device problem. Vampires are electrical appliances that drain power when they are left on standby or aren't in use. Ask students in the last class of the day to use 'off' rather than 'standby' in a computer lab; to use 'hibernate' rather than 'sleep'; and to choose low-power options where available.
- **Avoid creating waste:** upgrade your systems, and if that isn't viable, donate

NEIL GORDON

Neil is a National Teaching Fellow and a senior lecturer in computer science at the University of Hull, UK. He has over 20 years' experience teaching computing within higher education, including designing and teaching modules on sustainable computing.
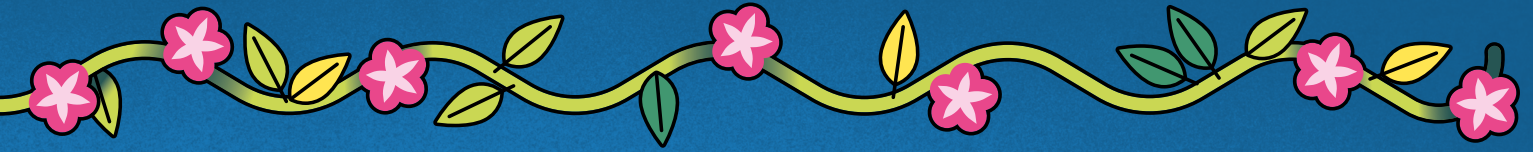
■ Computer scientists can help change the world for good, for example by creating technology for smart and sustainable cities

them to charities that will reuse and share them (see James Abela and John Ling's article on page 24 for an example of this).

■ **Think about the problems you are trying to solve:** can you solve a problem in a more efficient way? While modern computer power means it can be tempting to simply find a solution, finding an efficient algorithm to solve a problem means it needs less resources, which means less pollution. For software that may be run millions of times, this could be as simple as utilising De Morgan's laws and replacing 'if not(a) and not(b)' with 'if not(a or b)' (**helloworld.cc/demorgan**). Another example would be to make sure you use an efficient sort algorithm, such as Quicksort.

The positive impact

Of course, this isn't to say that we should turn off all our devices now. As well as giving students advice about the ways to minimise the negative impacts of technology, we should balance this by exploring the positive things we can do as computer scientists.

For example, we can solve many world problems using computer science:

■ **Improve logistics:** if we can minimise the routing of goods around the world, we can save resources. This is an application of the travelling salesman algorithm problem: given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city exactly once and returns to the origin city?

■ **Develop sustainable cities:** we can create smart buildings and smart energy grids that minimise waste energy and optimise use, using features such as lighting and heating that switch on when a room is occupied and go into low-power mode when it is empty.

■ **Educate people:** we can create games that help to educate people about issues of sustainability; **gamesforchange.org** includes great examples of games that help us to understand other people's perspectives and challenges, and my university uses games to teach about environmental problems such as flooding (**helloworld.cc/floodgame**).

■ **Solve world problems with technology:** computing technology has been called upon during the coronavirus pandemic, from the modelling of the virus itself, to the modelling and visualisation of how it spreads, to solutions to social distancing through online meetings, online learning, and online shopping.

Some of these are clearly in line with the UN SDGs, so a good project or task is to get your students to explore the SDG themes and investigate how computing can help address them. There are lots of examples, from health (Goal 3) and education (Goal 4) to sustainable cities (Goal 11). Some of the examples above also cover consumption (Goal 12) and helping to address climate change (Goal 13).

These discussions can help students to think about what has been done with computer science, and what is emerging, so they can see that computing has changed the world, and increasingly so — and we can ensure that this change is for the common good. I believe computer science can address the Sustainable Development Goals and we can make the world better through computing — I hope you do too. (HW)

REFURBISHING COMPUTERS FOR REFUGEES

James Abela introduces his 15-year-old student John Ling and tells the tale of how he sourced, refurbished, and distributed computers for Rohingya refugees in Malaysia

T he local residency association in our area of Kuala Lumpur wanted to help students in need during the pandemic by providing them with used computers donated by the community. However, nobody in the association had the skills needed to do repairs, and so when a member of the association heard that John was good at fixing computers, they were keen for him to get involved.

John had high hopes that he could make a real difference to the Rohingya refugee centres. Although these refugees considered themselves to be the lucky ones in many ways, because they had escaped Myanmar, Malaysia was taking them in only on a temporary basis. Their schools were very basic, and this was made worse by a lockdown that meant they had no access at all to education.

Unfortunately, the residency association only got one computer donation, and it was a 22-year-old IBM ThinkPad that John could not get the required spare part for. John felt really bad about this, and so he asked everyone he knew whether they had a machine, in any condition, that they might give to the cause. Emma Jones, one of John's neighbours, recalls, "I had these three broken old laptops and they were doing nothing in my back room. They weren't worth repairing, so I gave them to John to see if he could do anything with them." John seized the opportunity and repaired

■ John repairing computers

them. Initially they were used for distance learning, and then they were used in the refugee school.

The corporate response

John also explored other positive leads. For example, he contacted AmBank, who said that if he could prove his scheme by refurbishing ten computers, they would donate the machines that had finished their five-year cycle of useful life at the bank.

He set up a Facebook page, which quickly gained traction (**helloworld.cc/ pluggedin**). John started to get a few more donations and gratefully accepted them, writing each person a heartfelt letter of thanks. One of these letters went to Jane Wilson, and she showed her husband, who was the country president for Chubb. He was so impressed that he asked around in Chubb to see which computers might be due for renewal soon and could be sent to John for refurbishment. John had now also proven his scheme with ten machines, and so AmBank released their computers, too.

Operation: refurbish

Altogether, 21 desktops were donated, and so John now needed help to get them all processed and sent to the centres. To do this, he gathered four of his friends for a refurbishment party. The corporate machines were all mechanically sound and had all been very thoroughly wiped

THE REFURBISHMENT PROCESS

John used this process for refurbishment:

1. Identify faulty parts by pressing the power button and seeing how far along the process the computer gets; many cannot reach the login page

2. Unplug and disassemble the computer to isolate the casing and the motherboard

3. Clean the casing with soapy water and leave it to dry

4. Replace faulty parts

5. Update memory depending on DIMM type (DDR2 - 4GB, DDR3/DDR3L/DDR4 - 8GB)

6. Clean cooling fan and CPU heatsink

7. Apply new CPU thermal paste

8. Reassemble

9. Plug in and install operating system

■ The refurbished computers being used in the Rohingya centre classroom

clean. The first thing that John did was to install an operating system; each of the motherboards had a licence key for Windows 7, and as that was what the teachers were familiar with, that's what he installed. If you do not have a licence key, Linux is an excellent open-source option and **distrochooser.de** can help you choose the right distribution.

The team then split into two. One team labelled the ports so that it was easy to plug in peripherals, and the other team set up USB Wi-Fi dongles and installed the software and teaching materials. These machines were then ready for delivery to the centres. The process for computers that were donated by individuals rather than corporations was generally more extensive — see 'The refurbishment process' boxout for more information.

Plugged in Malaysia

To bring it together and make the project more sustainable, John has now branded his operation 'Plugged in Malaysia' and is trying to get more people involved. He is pleased that he has made a good start in equipping centres with computers, and now that people have heard of the scheme, he is getting more donations, including from some unlikely sources. He ordered some cupcakes to thank the IT team at AmBank, and it turned out that the baker was married to someone who works for DHL. The project is now investigating whether DHL too can donate their machines at the end of their corporate life.

In essence, his plan was simple, but it took a lot of resilience to implement: "I found two ways to get computers for the centres. The first was from people who have old computers that don't work anymore, and some of those only need minor repairs. Even if only the monitor works, I can always put in a Raspberry Pi to make it a useful device, and the rest of the computer might make useful parts for another refurbishment. The second route is from companies. Most companies need reliable computers, and many have an end-of-life policy after three to five years where they might sell them off cheaply. Their corporate philanthropy would go much further, though, if they donated them to me for refurbishment so I can get them to children who really need them. I did this last summer when I was 15. If I can do it, I am sure there are many others who have the skills to do this for their communities."
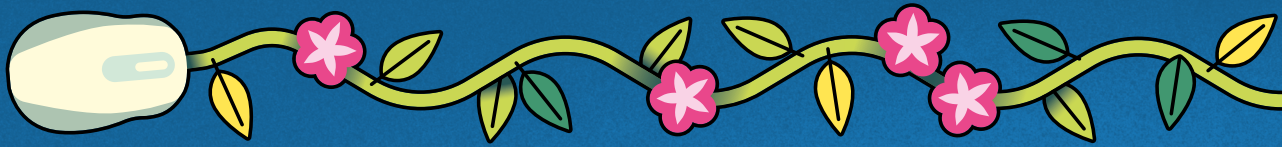
So far, John and his friends have managed to put 56 computers back into service (using parts from 94 computers) and helped more than 850 students in Rohingya teaching centres. He has also started distributing computers to MYReaders centres, which help low-income families gain access to electronic reading materials (**myreaders.org.my**). He hopes that people will continue to donate, and he is now training up more students to refurbish machines, so that the charity can have access to a hub of volunteers in Kuala Lumpur. (HW)

JOHN LING & JAMES ABELA

John is an A-level computer science student at Garden International School in Kuala Lumpur, Malaysia. He is the founder of Plugged in Malaysia (**helloworld.cc/ pluggedin**). John collects, refurbishes, and distributes old computers to give to underprivileged students for learning. James is head of computing at Garden International. He is a Raspberry Pi Certified Educator, founder of the South East Asian Computer Science Teachers Association, and author of *The Gamified Classroom* (**@eslweb**).

REAL-WORLD PROBLEM-SOLVING

Developing cross-curricular, problem-centred computing lessons from scratch can be daunting; **Peter Gaynord** shares his experience of this with a real-life environmental case study

The prospect of developing your own unit of work from scratch can feel very daunting. With the number of free resources available, it begs the question, why do it? Firstly, it gives you the opportunity to deliver computing that is interwoven with the rest of your curriculum. It also naturally lends itself to a constructionist approach to learning through meaningful engagement with real-world problem-solving. In this article, I am going to share my experience of developing a ten-lesson unit of physical computing for students aged nine to ten that is linked to the more general topic of the environment.

PETER GAYNORD
Peter is a computing teacher at Saffron Walden County High School in the UK, where he is also the primary lead and a facilitator for the London, Essex, and Hertfordshire NCCE hub. Prior to that, he spent 14 years as a general primary teacher. He is also a CAS Master Teacher and a Raspberry Pi Certified Educator.

The problem

To engage children in the process of problem-solving, it is important that the problem is presented as a real and meaningful one. To introduce the topic of the environment, we showed pupils a video of the Panama Canal, including information about the staggering amount of CO_2 that is saved by ships taking this route instead of the alternative, longer routes that use more fuel. However, we explained that because of the special geographical features, a moving bridge needed to be constructed over the canal. The students' challenge was first to design a solution to the problem, and then to make a working model.

The model would use physical computing as part of the solution to the problem. The children would program a single-geared motor using a Crumble microcontroller to slowly lift and lower the bridge by the desired amount. We decided to issue a warning to drivers that the road bridge was about to close using a Sparkle, a programmable LED. Ultimately, the raising and lowering of the bridge would happen automatically when a ship approached. For this purpose, we would use an ultrasonic sensor to detect the presence of the ship.
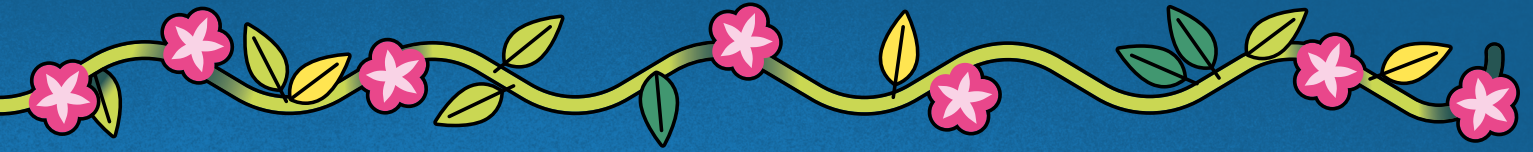
Building the required skills

To develop the skills required to use the Crumble microcontroller, we led some discrete computing lessons based largely on the Teach Computing Curriculum's 'Programming A — Selection in physical computing' unit (**helloworld.cc/tccphysical**).

In these lessons, the children developed the skill of sensing and responding differently to conditions using the selection programming construct. They learnt this key concept alongside controlling and connecting the motor, the Sparkle, and the ultrasonic sensor.

For students to succeed, we also had to teach them skills from other subjects, and consider at what stage it would be most useful to introduce them. For example, before asking children to document their designs, we first needed to teach the design technology (DT) objectives for communicating ideas through sketches. Most other DT objectives that covered the practical skills to make a model were interwoven as the project progressed. At the end of the project, we guided the children through how to evaluate their design ideas and reflect on the process of model making. Before pupils designed their solutions, we also had to introduce some science for them to apply to their designs. We covered increasing forces using levers, pulleys, and gears, as well as the greenhouse effect and how burning fossil fuels contributes to global warming.

It is very important not to specify a solution for students at the beginning, otherwise the whole project becomes craft instead of problem-solving. However, it is important to spend some time thinking about any practical aspects of the model building that may need extra scaffolding. Experience suggested that it was important to limit the scale of the children's models.
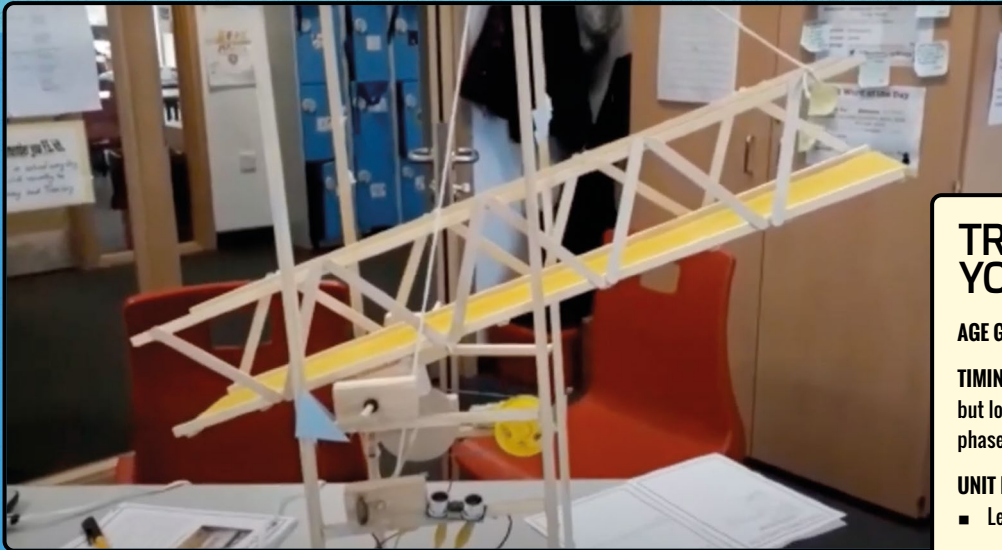
■ Pupils faced all sorts of challenges when building their bridges and had to rethink their initial designs

We did this by showing them a completed central bridge span and later, guiding the building of this component so that all bridges had the same scale. It also turned out to be very important that the children were limited in their model building to using one single-geared motor. This would ensure that all children engaged with actively thinking about how to utilise the lever and pulley system to increase force, instead of relying on using more motors to lift the bridge.

collectively to make a moving bridge in the context of a sustainability issue.

It also brings to the fore the key problem-solving skills of curiosity, resilience, resourcefulness, creativity, and adaptability. I recall children facing all sorts of challenges and having to rethink and improve on their initial designs. I remember having to physically model the need to check 'for'

> ❝ I RECALL THE SHEER DELIGHT ON A CHILD'S FACE AS THEY SAID THEY'D "CRACKED IT"

Benefits of problem-centred learning

There are many advantages for children of learning in this cross-curricular, problem-centred way instead of learning being delivered in distinct and separate subject silos. In this project, for example, the children didn't simply learn and store the skill of how to recognise 'for' conditions in a loop, or how levers and pulleys increase force. Instead, they were also challenged to apply all these skills and knowledge

conditions repeatedly, as some children found that they weren't detecting the ship's presence and needed to debug their code and add the loop. I also remember the sheer delight on one child's face when they told me they had "cracked it".

The DT programme of study for five- to eleven-year-olds in England outlines how children should acquire subject knowledge in subjects such as science, engineering, and computing, but then challenges them to apply this and take risks to become resourceful, innovative, enterprising, and capable citizens. Likewise, the computing programme states that computing has deep

links with maths, science, and design technology, and enables students to link computational thinking with these in order to both understand and change the world. The UK Government's own strategy for educating children about sustainability states that although we should present children with the truth about climate change, we must also foster hope and determination in them to meet the challenge ahead (**helloworld.cc/climatepaper**). You can see how we did this through this project, with a video of some of the bridge models in action at **helloworld.cc/bridgevid**. (HW)

TRY IT FOR YOURSELF!

AGE GROUP: 9-10

TIMINGS: One hour in the earlier lessons, but longer sessions during the construction phase, to allow time for setting up the room

UNIT BREAKDOWN:

- Lesson 1: Levers and pulleys - science
- Lesson 2: Controlling motors and LEDs with the Crumble - computing
- Lesson 3: Making selections using an ultrasonic sensor - computing
- Lesson 4: Climate change and sustainability - introduction to the challenge
- Lesson 5: Designing a solution - DT
- Lesson 6: Guided lesson on building the central span - DT
- Lessons 7-9: Constructing the models - DT, science, computing
- Lesson 10: Final evaluation and celebration

COMPUTING AND SUSTAINABILITY RESEARCH TO INSPIRE STUDENTS

Researchers at the University of Cambridge outline their computational approaches to addressing global environmental challenges

Providing students with context about why they are learning something is a valuable engagement tool, especially when that context is as meaningful and widely relevant as environmental issues. Here, we hear from researchers at the UKRI Centre for Doctoral Training in the Application of Artificial Intelligence to the study of Environmental Risks (**helloworld.cc/ai4er**), who outline their current work at the intersection of artificial intelligence and environmental challenges. You can use these profiles to inspire your older students when introducing related topic areas, helping them to understand how the knowledge and skills they are developing in their computer science studies can be built upon and applied by researchers to develop solutions to global environmental problems. **(HW)**

ASSESSING AIR QUALITY AND HUMAN HEALTH

Name: Michelle Wan

What does your research involve? My work draws upon multiple STEM disciplines including atmospheric chemistry, epidemiology, computer science, and sociology. I research the impact of air quality on human health, with particular interest in the context of social inequity in urban areas. My current research focuses on Greater London and vehicle-emitted pollutants such as nitrogen dioxide and particulates. I work with large data sets and explore the applications of machine learning techniques, like specific types of neural networks which can handle data across detailed spatial and temporal ranges. I aim to develop a modelling framework which will estimate the health impacts of potential future climate and emissions scenarios.

Why is your research important? The various types of air pollutants have well-established long- and short-term effects on disease incidence and death rates. In addition to air pollution, social inequality has also been strongly linked to health; indeed, both factors have made headlines since the beginning of the Covid pandemic. My research aims to demonstrate the importance of taking a more holistic view of air quality and health, and highlight the devastating impact that social inequity continues to have.

I initially joined the AI4ER graduate programme having previously worked with air quality data, and hoped to gain experience with machine learning techniques to leverage the increasingly large data sets that we can now gather.

Computing topic areas to which teachers can link this research:

- Machine learning
- AI ethics
- Big data

PREDICTING PRECIPITATION

Name: Kenza Tazi

What does your research involve? I create computer models to improve our understanding of rain and snowfall in mountainous areas. I use machine learning methods that can not only predict precipitation more accurately, but can also infer the probability of more extreme precipitation events. **Why is this challenging?** The complex landscape of mountains means that some valleys receive a lot of snow and rain while neighbouring valleys receive relatively little. There are also few direct precipitation measurements in mountains. Observations from satellite and climate model simulations are useful, but do not resolve precipitation at a fine enough scale to make decisions such as where to build a new dam.

Why is your research important? Mountains provide fresh water for over half the world's population. This includes water for drinking, domestic use, industry, agriculture, and energy through hydropower plants. How the distribution of precipitation will change as a result of climate change is even more ambiguous. However, there will be more extreme precipitation, in general, which will lead to more floods, landslides, and droughts. Improving our understanding of rain and snowfall is especially important in areas such as the Himalayas where there are even fewer measurements, and water scarcity could lead to mass migration and violent conflict. The results from the models will help communities adapt to a changing climate.

Computing topic areas to which teachers can link this research:

- Machine learning
- Bayesian inference (**helloworld.cc/bayesian**)

UNDERSTANDING FUTURE EXTREME FLOODS

Name: Tudor Suciu

What does your research involve? My research looks at how global warming is changing extreme weather events and how this is impacting people around the globe, with a focus on coastal flooding. My work is data-driven, as opposed to similar projects that focus on using physics-based models. I deal on a daily basis with enormous data sets of past recordings of sea level and floods, and past weather data from satellites and weather stations. I also need future predictions of weather parameters, which we have from global climate models.

The main aim is to use machine learning methods to find relevant statistics on coastal flooding. One important note is that I don't want to have a precise forecast of coastal floodings, such as saying that on 14 March 2032 (for example) we will have a huge storm, as this would be impossible to predict using climate models. Instead, we need to focus on a large time frame, for example saying that the biggest storm to hit the UK in the past 100 years, in 1953, has a return period that decreased by n-times.

Why is your research important? Through my work, I would like to contribute to tackling the climate crisis and, more importantly, helping people! There is value in knowing how the statistics (frequency and intensity) of future coastal floods will be affected by climate change under different emissions scenarios. People living by coasts require the best protection against those disastrous events, when water is being pushed from the sea towards the land by storms. My research can help, as knowing that the biggest storms will get more intense or more frequent can help direct more funds to building better protective infrastructure.

Computing topic areas to which teachers can link this research:

- Climate science
- Data science/machine learning/statistics

TOMORROW'S ENGINEERS

Can young people use engineering to save the planet?
Neha Okhandiar looks back at the very first Schools COP

Tomorrow's Engineers Week is an annual celebration of all engineers run by EngineeringUK with the aim of inspiring young people (**helloworld.cc/tomorrowsengineers**). The highlight of the 2021 event (**#TEWeek21**) was the Tomorrow's Engineers Week Schools COP. Taking place during the second week of the much-anticipated COP26 climate summit, schools and young people from around the UK came together to discuss the importance of net zero and explore how engineers are at the heart of tackling the climate crisis.

Held in Glasgow, UK, COP26 aimed to respond to the climate emergency our planet is facing — from the increase in extreme weather events, to how our cities cope with polluting vehicles, and how we achieve net zero. At EngineeringUK, we are clear that these are not just issues for policymakers and activists — we need engineers and scientists to help come up with solutions, too. The Schools COP gave the engineering community an opportunity to speak with one voice about the importance of engineering in tackling climate change. Recent evidence shows that young people who agree that engineers are important for improving the environment were seven times more likely to be interested in a career that involved engineering (**helloworld.cc/engineeringuk2021**).

Creating a satisfying summit

Armed with that knowledge, we invited schools and youth groups to be climate change ambassadors and take part in our summit — the UK's first Schools COP. The discussion was hosted by STEM champion and broadcaster Fayon Dixon and former BBC newsreader Susan Bookbinder. There were four breakout rooms mirroring the topics being debated at the main COP26. The topics covered were:

- Youth and public involvement
- Science and innovation
- Nature, cities, regions, and the built environment
- Climate crisis and me

The summit also featured two inspiring talks from Jaz Rabadia MBE and Thaddeus Anim-Somuah. Jaz is an energy manager who helps businesses to reduce the amount of energy they use and lower their carbon footprint. Thaddeus is a chemical engineer who leads his company's commitment to sustainability, which includes a target of reducing carbon emissions by 25 percent between 2018 and 2024. Students had the chance to ask Jaz and Thaddeus questions and discover how engineers are at the heart of solutions to achieve net zero by 2050.

At the summit, we heard from 70 students from 30 schools across the UK about what they feel they can do as young people, how they can make changes to their own behaviours, and what issues were the most alarming for them, as well as the all-important conversation about how engineering, science, and innovation

■ Students from one of the participating schools tuning into the Schools COP

> ## " THE STUDENTS' PASSION AND ENTHUSIASM TO ACHIEVE CHANGE WAS REMARKABLE

can help achieve net zero. Their passion and enthusiasm to achieve change were remarkable — and their knowledge about the climate emergency was humbling.

Fizzing with ideas

There were so many amazing ideas that came through. Some link up with technologies currently being explored, such as ocean fertilisation to improve carbon capture by marine plants. Other brilliant suggestions for tackling climate change made by the students included:

■ Better recycling of plastics, and using biopolymers to create plastics that are less harmful to the environment

■ Promoting positive practices and educating young people through social media and advertising

■ Developing self-sufficient agri-biomes for animal farming, to capture methane

to be converted into energy sources

■ Investment in new energy sources to improve new buildings and retrofit onto older ones, and planting more trees to increase biodiversity

For those who missed out on the live event, the session was recorded (**helloworld.cc/copvideo**) and released alongside a lesson plan for teachers to use in personal, social, health, and citizenship education (PSHCE), careers, geography, or science lessons (**helloworld.cc/coplesson**). Over 300 schools watched the broadcast on demand and over 33,000 students took part, with over 75 percent of teachers saying it inspired students to consider a career in engineering.

Tomorrow's Engineers Week returns for its tenth anniversary in 2022 — sign up to our mailing list now at **helloworld.cc/temail** to hear when more plans are announced. (HW)

TRY OUT THIS SCHOOLS COP ACTIVITY

Explain to students that COP26 had a variety of goals to achieve (you can find a summary at **helloworld.cc/cop26**).

Split students into four groups and ask them to consider the following questions about how their future jobs could help to achieve these goals:

GROUP ONE: What can you do to tackle climate change in your future career?

GROUP TWO: What can your school or local community do to tackle climate change?

GROUP THREE: What can your country do to tackle climate change?

GROUP FOUR: What engineering solutions do you think could tackle climate change?

You can find more lesson ideas at **helloworld.cc/coplesson**.

NEHA OKHANDIAR
Neha is the senior media manager at EngineeringUK and has worked in science communications for nearly 20 years, translating complex STEM ideas for the general public (**@Neha_Oh**).

ENCOURAGING SUSTAINABILITY DISCUSSIONS THROUGH 3D DESIGN

How can we encourage discussion about climate change through designing 3D objects?
Richard Smith introduces the create, question, discuss process

During a lesson in which students were creating 3D models of wind turbines, a ten-year-old student asked, "Why do most turbines have three blades rather than four?" This sort of question can really support good-quality discussion in the context of climate change, and I view this as part of a create, question, discuss process. Students first create something; then, as they become involved

in the creative process of an object, a range of questions are likely to come to mind, which, in turn, informs discussion. In this article, I will outline how we can encourage discussion about climate change with primary-school students through the design of 3D objects.

The 'create' stage

I asked students to create a model of a solar panel on the online 3D platform Tinkercad (**tinkercad.com**). Before getting started, it's helpful for teachers to explain that we can make 3D objects from materials such as cardboard, clay, or paper, or a digital model can be created using computer-aided design (CAD) software. I have also found it useful for the teacher to explain four concepts using Tinkercad:

- Models are created on a flat plane that can be rotated and angled
- Models are made of existing 3D shapes that can be dragged around and resized
- The 3D shapes can either be solid or hollow
- The individual 3D shapes can be grouped together to create a new shape that can be moved, resized, or copied and pasted

The great thing about creating a solar panel is that only cuboids are required as a basis. These can be copied and grouped. Students can use other shapes such as a cylinder for the solar panel stand, if they choose to. During the creation stage, I use words such as 'array' and 'polygon' to illustrate that computing and maths have vocabulary in common.
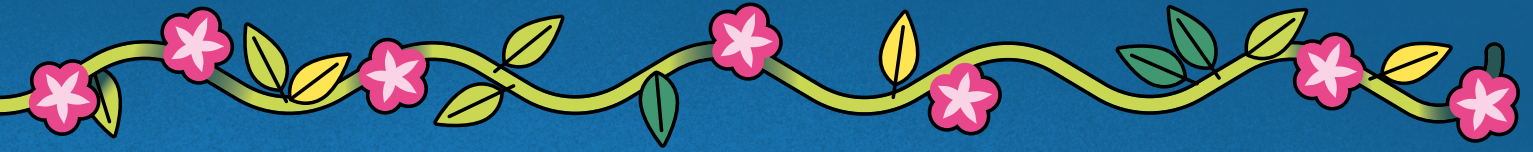
When students use Tinkercad, I find that there is a range of reactions. Some students find it easy to create and manipulate the 3D shapes. By using different viewpoints, they manage to line objects up perfectly. For these students, you could stretch them to use the 'codeblocks' option in which they use block-based coding to create their shapes (**helloworld.cc/codeblocks**). Others really struggle to rotate, resize, and lift objects above the flat plane. To support these students, you can:

- **Simplify the project:** for example, a basic version would be made up of just two shapes: a cuboid to represent the panel and a second cuboid for the stand.
- **Introduce keyboard shortcuts:** for example, Ctrl + Up Arrow lifts the shape upwards. I provide students with a help sheet (**helloworld.cc/tinkerhelp**).

RICHARD SMITH
Richard works in five schools a week supporting students and staff to become confident and safe users of technology. He helps run a Computing at School Community Hub in Telford, UK. He is the founder of the company AmazingICT and regularly shares ideas on Twitter via **@amazingict**.

EXTRA SUPPORT

- **helloworld.cc/tcc3d:** a series of six useful 3D-modelling lessons, for students aged ten to eleven, written by England's National Centre for Computer Education

- **tinkercad.com/learn** and **tinkercad.com/ teach:** Tinkercad provides helpful advice for teachers and learners

- **amazingict.com** and **amazingict@ icloud.com:** feel free to contact Richard with questions

- **Use terminology:** for example, use 'bird's-eye view' and 'looking at it from the front' to reinforce language such as 'top view' and 'front view'.

You can access my full lesson plan at **helloworld.cc/solarlesson** and an accompanying video at **helloworld.cc/solarvideo**.

The 'question' and 'discuss' stages

The creation stage allows questions to occur naturally, as opposed to being set

■ We can encourage discussion about climate change through the design of 3D objects such as solar panels

in understanding — for example, when students ask, "How does the solar panel get electricity from the Sun?", rather than "How is solar energy from the Sun converted into electricity?" This stage also directly feeds the discussion stage, in which discussions can take place either

> ## " THIS PROCESS IS A POWERFUL WAY OF ALLOWING FOR RICH, STUDENT-LED DISCUSSION ON MEANINGFUL TOPICS

by the teacher. Here are three examples of questions from my students as they were creating their model solar panels:

1. How are they made waterproof?
2. Why don't we use more of them?
3. How do they work?

The question stage provides an opportunity for educators to spot gaps

informally or formally, with associated research via books or the internet.

These discussions tend to fall into two categories. The first category is where the discussion centres around quite specific or narrow questions, such as "Do the solar panels overheat?" These types of question provide valuable opportunities for students to go away and carry out some research — in this example, research into materials and

scientific processes. The second category is where discussion centres around wider or less specific questions, such as "Why don't we use more of them?" These types of discussion often link well to other subjects and can be used as a starting point for broader project work, displays, assemblies, and even a home/school project that involves parents and carers.

As a next step, I am going to help students to create other objects linked to sustainability through the create, question, discuss process. For example, in the case of the solar panel model, it is critical that we consider how to store the energy created for later use. Students could therefore design batteries to store energy, using basic shapes such as cuboids and cylinders. Questions such as "How can the batteries be recycled?" are then likely to arise, which will inform discussion. Overall, this process is a powerful way of allowing for rich, student-led discussion on meaningful topics such as sustainability — try it out for yourself! **(HW)**

PROJECTS FOR A SUSTAINABLE FUTURE

Rosa Brown takes a look at the inspiring tech projects entered by young creators into Coolest Projects 2022 on the themes of sustainability and the environment

C oolest Projects is a celebration of young people who learn new skills, experiment, and make something important to them with technology. In this year's international showcase over 300 entries were submitted about the environment. Here are some of these projects and a glimpse into how children across the world are using technology to look after their environment.

One simple change

The idea that a small change can lead to a big impact may be familiar to many of us. Two entries for this year's Coolest Projects were from young people who put this idea into practice with clever inventions to make positive changes to the environment.

Arik (15) from the UK noticed that whenever his family had lots of people at their house, getting the right drink for everyone was a challenge, often resulting in wasted, spilt drinks. To solve this problem, Arik was inspired to make his very own Liquid Dispenser. The dispenser can hold two litres of any liquid, and its outer body is made from reused cardboard (**helloworld. cc/liquiddispenser**). It is very easy for people to get their drink with the press of a button. As Arik says, "You don't need a plastic bottle, you just need a cup!"

Amrit (13), Kingston (12), and Henry (12) from Canada were also inspired to make a tech project to reduce waste, "because we wanted to create something that hopefully would create a meaningful impact on the world". The Eco Light is a light that automatically turns off when someone leaves their house, avoiding wasted electricity (**helloworld.cc/ecolight**). The project uses a micro:bit to detect the signal strength and decide whether the LED should be on (if someone is in the house) or off (if the house is empty).

Thinking local and global

One of my favourite things about Coolest Projects is seeing young people inventing things that can effect positive changes in the community on both a local and a global level. This year, Sashrika (11) from the US shared her Gas Leak Detector project, which she designed to help people who heat their homes with diesel (**helloworld.cc/gasleak**). On the east coast of the USA, many people store their gas tanks in the basement, which means they may not realise if the gas is leaking, and the consequences could be severe. So Sashrika combined programming with physical computing to make a device that can detect if there is a gas leak and send a notification to your phone.

Sashrika has even thought about how she could make more changes to her project to meet her environmental goals: "I would probably add a solar panel, because there are lots of houses that have outdoor oil tanks. Solar panel[s] will reduce electricity consumption and reduce CO_2 emission[s]."

Renewable energy sources are also important to Amr from Syria, who created his own Smart Wind Turbine to "generate clean energy, because we have to reduce pollution from burning fossil fuels" (**helloworld.cc/smartwind**). Amr's wind turbine is connected to a micro:bit to measure the electricity generated by a fan. Through his tests, Amr recorded that more electricity was generated when the turbine faced the direction of the wind. So he made a wind vane to determine the wind's direction and added another micro:bit to communicate the results to the turbine.

Other young people have created

■ Sashrika, aged eleven, shows off her Gas Leak Detector project

■ Andrea and Yuliana, aged nine and ten, exhibit their Bee Counter project

" MANY YOUNG CREATORS MADE PROJECTS THAT AIMED TO EDUCATE AND INSPIRE

ROSA BROWN
Rosa is a copy editor at the Raspberry Pi Foundation. She was a judge at Coolest Projects 2022.

projects to make the world a better place for future generations. Naira and Rhythm from India wanted to create houses to benefit both people and the environment. They carried out a survey, and from their results they designed the Net Zero Home (**helloworld.cc/netzerohome**). Naira and Rhythm's model "has been constructed by a method of Green construction which utilizes environment[tally] friendly techniques to create a sustainable and energy-efficient final product". Net Zero Home offers an idea for homes that are comfortable for people of all abilities and ages, while also being sustainable.

Andrea (9) and Yuliana (10) from the US were also thinking about future generations when they designed their Bee Counter project (**helloworld.cc/beecounter**). They used sensors and programmed a micro:bit to record bees' activity around a hive: "It would be impossible to feed our global population without honeybees. Bees support the growth of most plants. But unfortunately, bees have been declining for years." By monitoring the bees, Andrea and

Yuliana hope they can see (and then fix) any problems there might be with the hive, and maintaining the bees' home will help them continue to have a positive influence on our environment.

Knowledge is power
Some young creators use Coolest Projects as an opportunity to educate and inspire people to make environmental changes in their own lives. Whether through a game, app, or website, this year there have been numerous projects that demonstrate how important the environment is to young people.

Sabrina (13) from the UK has created her own website, A Guide to Climate Change (**helloworld.cc/climateguide**). It includes images, text, and graphics relating to the Earth's temperature change, and includes suggestions on how people can minimise their waste.

Using technology to encourage people to help save the environment is also important to Kushal (12) from India. Kushal had no experience of app development before making his Green Steps app
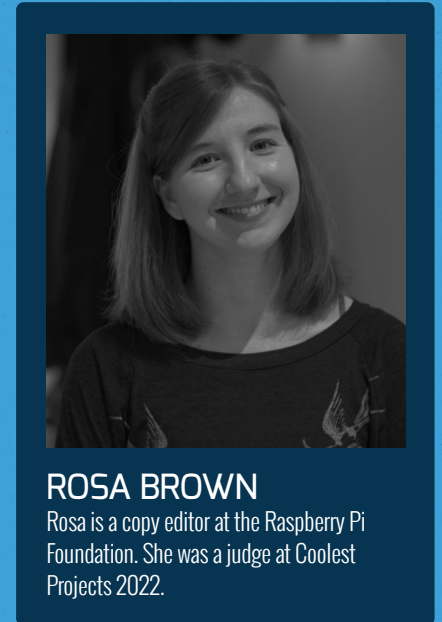
(**helloworld.cc/greensteps**). He says, "I have created a mobile app to connect like-minded people who want to do something about [the] environment." Green Steps helps people to upload and save pictures, like content from other users, and access helpful resources.

These projects are just some of the remarkable ideas we've seen young people enter for Coolest Projects this year. It's clear from the projects submitted that the context of the environment and protecting our planet resonates with so many students. As Sabrina sums it up, "Some of us don't understand how important the Earth is to us. And I hope we don't have to wait until it is gone to realise." Check out the Coolest Projects showcase at **helloworld.cc/coolestprojects2022** for even more projects about the environment, alongside other topics that have inspired young creators. (HW)
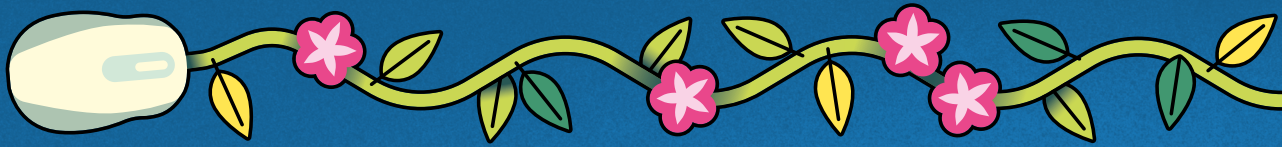
DOES UPGRADE CULTURE NEED AN UPGRADE?

Mac Bowley questions our habit of quickly replacing devices with newer versions, with some interesting discussion points for students

Technology is more embedded in our lives than ever before, and most of us now carry a computer in our pocket everywhere we go. On top of that, the length of time for which we use each individual piece of technology has rapidly decreased. This is what's referred to as upgrade culture, a cycle that sees many of us replacing our most trusted devices every two to three years with the latest products offered by tech giants such as Apple and Samsung.

How we got to this point is hard to determine, and there does not seem to be a single root cause of upgrade culture. This is why I want to start a conversation about it, so we can challenge our current perspectives and establish fact-based attitudes. I think it's time that we, as individuals and as a society, examine our relationship with new technology.

What is the natural lifespan of a device?

Digital technology is still so new that there is really no benchmark for how long digital devices should last. This means that the decision-making power has by default landed in the hands of device manufacturers and mobile network carriers, and for their profit margins, a two- or three-year life cycle of devices is beneficial.

Where do you, as a consumer, see your role in this process? Is it wrong to want to upgrade your phone after two or three years of constant use? Should phone companies slow their development, and would this hinder innovation? And, if you really need to upgrade, is there a better use for your old device than living in a drawer? These questions defy simple answers, but your students should be aware of their role in this process as consumers, so this is a great area for in-class discussion.

How does this affect the environment?

As with all our behaviours as consumers, the impact that upgrade culture has on the environment is an important concern. Environmental issues and climate change aren't anything new, but they're currently at the forefront of the global conversation, and with good reason.

There are a number of issues around the manufacture of our mobile devices, such

© sashazerg/stock.adobe.com

■ Is it wrong to want to upgrade your phone after two or three years of constant use?

■ A common trope of mobile phone adverts is the overwrought comparison of your current device with a newly launched version
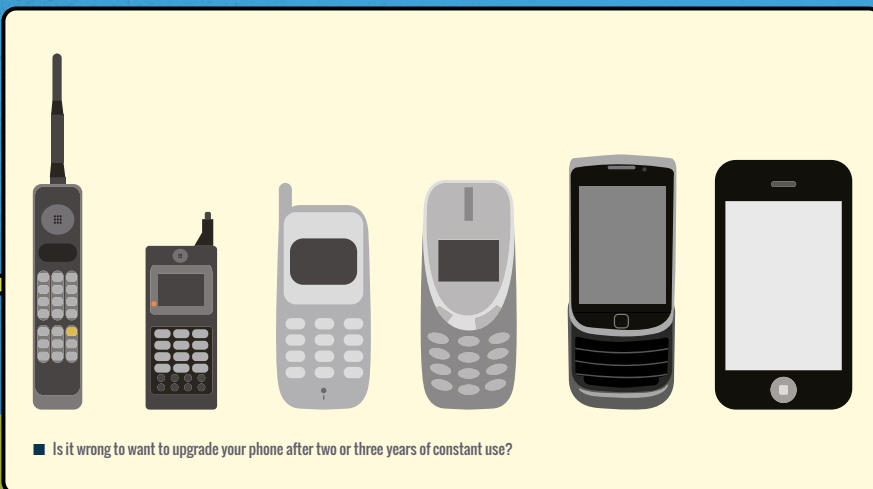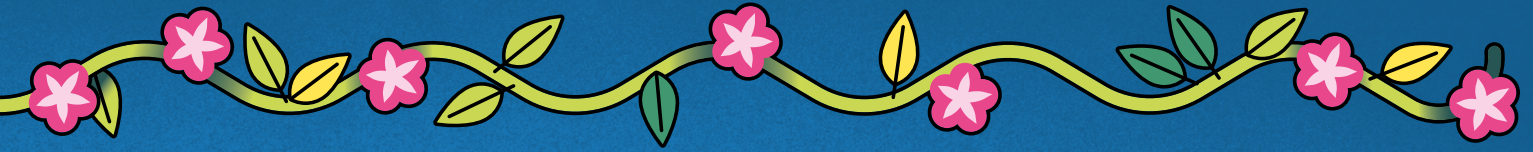
as the large amounts of energy required. Here, though, I would like to focus on another aspect of the environmental impact of device production: the materials that are used to create some of the tiny components that form our technological best friends.

> **WE HAVE LET DEVICE MANUFACTURERS AND NETWORK CARRIERS TELL US HOW LONG OUR DIGITAL DEVICES SHOULD LAST**

Some components of your phone cannot be created without using rare chemical elements, such as europium and dysprosium. (In fact, there are 82 stable non-radioactive elements in the periodic table, and 70 of them are used in some capacity in your phone; see **helloworld. cc/phoneelements**). Upgrade culture

means there is high demand for these materials, and deposits are becoming more and more depleted. If you're hoping there are renewable alternatives, you'll be disappointed: a study by researchers working at Yale University found that there are currently no alternative materials that are as effective (**helloworld.cc/ graedel2013**).
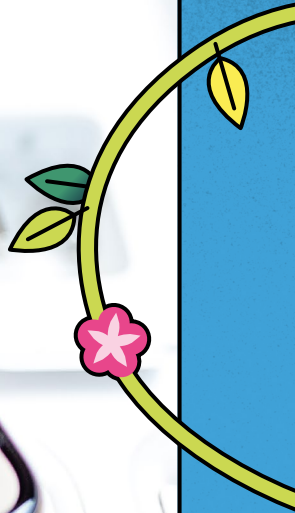
Then there's the issue of how the materials are mined. The market trading these materials is highly competitive, and more often than not, manufacturers buy from the companies that offer the lowest prices. To maintain their profit margins,

these companies have to extract as much material as possible, as cheaply as they can. As you can imagine, this leads to mining practices that are less than ethical or environmentally friendly. As many of the mines are located in distant areas of developing countries, these problems may feel remote to you, but they affect a lot of people and are a direct result of the market we are creating by upgrading our devices so frequently.

Many of us agree that we need to do what we can to counteract climate change, and that to achieve anything meaningful, we have to start looking at the way we live our lives. This includes questioning how we use technology. It will be through discussion and opinion-gathering that we can start to make more informed decisions — both as individuals and as a society. ▶

© R_yosha/stock.adobe.com

■ The length of time for which we use individual pieces of technology has rapidly declined

▶ The obsolescence question

You probably also have that one friend/colleague/family member who swears by their five-year-old mobile phone and scoffs at the prices of the newest models. These people are often labelled as sticklers who are afraid to join the modern age, but is there another way of seeing them? The truth is, if you've bought a phone in the last five years, then — barring major accidents — it will most likely still function, and be just as effective as it was when it came out of the box. So why are so many consumers upgrading to new devices every two or three years?

There isn't a single reason, but I believe marketing departments should shoulder much of the responsibility. Using marketing strategies, device manufacturers and mobile network carriers purposefully encourage us to view the phones we currently own in a negative light. A common trope of mobile phone adverts is the overwrought comparison of your current device with a newly launched version. Thus, with each passing day after a new model is released, our opinion of our device worsens, even if only on a subconscious level.

This marketing strategy is related to a business practice called planned obsolescence, which sees manufacturers purposefully limiting the durability of their products in order to sell more units. An early example of planned obsolescence is the light bulb, invented by the Edison company. It was relatively simple for the company to create a light bulb that would last 2,500 hours, but it took years and a coalition of manufacturers to make a version that reliably broke after 1,000 hours (**helloworld.cc/lightbulb**). We're all aware that the light bulb revolutionised many aspects of life, but it turns out it also had a big influence on consumer habits and on what we see as acceptable practices by technology companies.

© Onidji/stock.adobe.com

■ Manufacturers often purposefully limit the durability of their products in order to sell more units
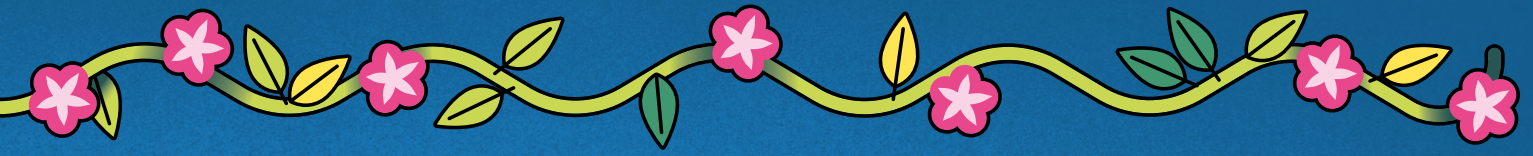
> ## "WE ARE WIDENING THE DIGITAL DIVIDE BY PLACING MORE VALUE ON NEW TECHNOLOGY THAN IS WARRANTED

MAC BOWLEY
Mac is a learning manager at the Raspberry Pi Foundation. When he isn't teaching, you can usually find him tinkering with his latest project (**@Mac_Bowley**).

The widening digital divide

The final aspect of the impact of upgrade culture that I want to examine relates to the digital divide. This term describes the societal gap between the people with access to, and 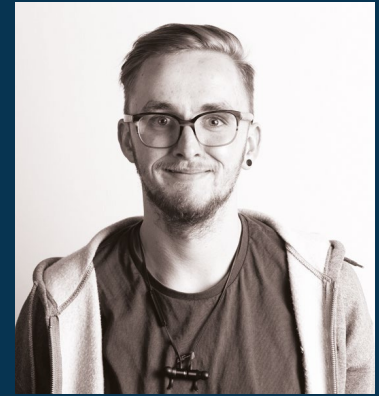competence with, the latest technology, and the people without these privileges. To be able to upgrade, say, your mobile phone to the latest model every two years, you either need a great degree of financial freedom, or you need to tie yourself to a 24-month contract that may not be easily within your means. As a society, we revere the latest technology and hold people with access to it in high regard. What does this say to people who do not have this access?

Inadvertently, we are widening the digital divide by placing more value on new technology than is warranted. Innovation is exciting, and commercial success is celebrated — but do you ever stop and ask who really benefits from this? Is your new phone really that much better than the old one, or could it be that you're mostly just basking in the social rewards of having the newest bit of kit?

What do you think? Time for you to discuss with your students! Here are some discussion starters to use with them:

- When you are in charge of buying your own phone, what can you do to make the device last longer than the two- to three-year upgrade cycle?
- Do you think upgrade culture is something that should be addressed by mobile phone manufacturers and providers, or is it caused by our own consumption habits?
- How might we address upgrade culture? Is it a problem that needs addressing?

Upgrade culture is one of the topics for which we offer you a discussion forum on our free online *Impact of Technology* course (**helloworld.cc/impactoftech**). The course, designed for educators, also covers how to facilitate classroom discussions about these topics — sign up today to take part for free! (HW)

WHAT ABOUT RASPBERRY PI TECHNOLOGY?

Obviously, this article wouldn't be complete if we didn't share our perspective as a technology company. So here's Raspberry Pi Trading CEO Eben Upton:

ON OUR HARDWARE AND SOFTWARE

"Raspberry Pi tries very hard to avoid obsoleting older products. Obviously the latest Raspberry Pi 4 (**helloworld.cc/pi4**) runs much faster than a Raspberry Pi 1 (something like 40 times faster), but a Raspberry Pi OS (**helloworld.cc/rpiOS**) image we release today will run on the very earliest Raspberry Pi prototypes from the summer of 2011. Cutting customers off from software support after a couple of years is unethical, and bad for business in the long term: fool me once, shame on you; fool me twice, shame on me. The best companies respect their customers' investment in their platforms, even if that investment happened far in the past.

"What's even more unusual about Raspberry Pi is that we aim to keep our products available for a long period of time. So not only can you run a 2020 software build on a 2014 Raspberry Pi 1B+; you can actually buy a brand-new 1B+ to run it on (**helloworld.cc/pi1b**)."

ON THE ENVIRONMENTAL IMPACT OF OUR HARDWARE

"We're constantly working to reduce the environmental footprint of Raspberry Pi. If you look at the USB connectors on Raspberry Pi 4, you'll see a chunky black component. This is the reservoir capacitor, which prevents the 5V rail from dropping too far when a new USB device is plugged in. By using a polymer electrolytic capacitor from our friends at Panasonic, we've been able to avoid the use of tantalum.

"When we launched the official USB-C power supply for Raspberry Pi 4 (**helloworld.cc/piUSBC**), one or two people on Twitter asked if we could eliminate the single-use plastic bag that surrounded the cable and plug assembly inside the box. Working with our partners at Kuantech, we found that we could easily do this for the white supplies, but not for the black ones. Why? Because when the box vibrates in transit, the plug scuffs against the case; this is visible on the black plastic, but not on the white. So for now, if you want to eliminate single-use plastics, buy a white supply. In the meantime, we'll be working to find a way (probably involving cunning origami) to eliminate plastic from the black supply."

DO YOUR :BIT!

Dan Stormont introduces the 'do your: bit' challenge, a great way to engage students with computational sustainability

In a 1999 essay, Seymour Papert and Gaston Caperton expressed this vision: "We imagine a school in which students and teachers excitedly and joyfully stretch themselves to their limits in pursuit of projects built on their own visions ... not one that merely succeeds in making apathetic students satisfy minimal standards" (**helloworld. cc/papert1999**). Over 20 years later, we often still use curricula with 'toy problems' that many computer science learners find uninteresting and difficult to relate to the real world. How can we change that?

Understandably, many young people today are concerned about the future of the world they will inherit. The United Nations Sustainable Development Goals (SDGs), adopted by all the member states of the UN in 2015, address critical issues for the future of our youth, including poverty, hunger, health, education, equality, economic opportunity, justice, and climate change (**helloworld.cc/sdg**). In this article, I will introduce a computational sustainability challenge (involving the application of computer science to sustainability) that combines physical computing with the issues students care about.

How to 'do your :bit'!

The 'do your :bit' challenge is an annual global competition that brings together the micro:bit (**microbit.org**) and the SDGs. There are three competition categories: idea only; design and code (both for ages 8 to 14); and design and code, for ages 15 to 18 (**helloworld.cc/doyourbit**). For 2022, the focus is on Goal 3, Good Health and Well-being; Goal 10, Reduced Inequalities; Goal 13, Climate Action; Goal 14, Life Below Water; and Goal 15, Life on Land. The challenge website includes goal guides to help educators introduce the SDGs, lesson plans, and micro:bit starter projects for all the focus goals.

At CoderDojo Tucson, we've worked through a number of these starter projects to help get students thinking about the SDGs. For example, for Goal 3 (Good Health and Well-being), we created a step counter to encourage people to get more exercise. We created an equality game for Goal 10 (Reduced Inequalities) based on the 'Send a smile' starter project (**helloworld. cc/microbitsmile**). The micro:bits were programmed using MakeCode (see boxout). The sending micro:bit would send a number to the receiving micro:bits; the receiving micro:bits, each of which was set to a different radio group, would display a number of hearts equal to the received number. What the holders of the receiving micro:bits didn't realise until they compared their displays, was that some of the micro:bits consistently received more hearts than others. It made a unique starting point for a conversation about equality, equity, and justice.

Another project I've used more than once is the soil moisture sensor activity (**helloworld.cc/microbitsoil**), which is an ideal starter project for Goal 15 (Life on Land). Living in the Sonoran Desert region of the southwestern United States, using water wisely is paramount. This simple project uses two nails, two leads, and a potted plant to create a simple soil moisture sensor. There is a lot of opportunity for customisation, and this is also a great project for talking about the principles of resistance and conductance in electrical circuits.

It should be apparent from these examples that there is a wide range of starter projects for the challenge. So how do you choose where to start with your

DAN STORMONT

Dan is the Champion for CoderDojo Tucson in Tucson, Arizona, USA, and a 2022 micro:bit champion. When not using micro:bits to engage young people in computing, he is a program specialist with the Microsoft Philanthropies TEALS Program.
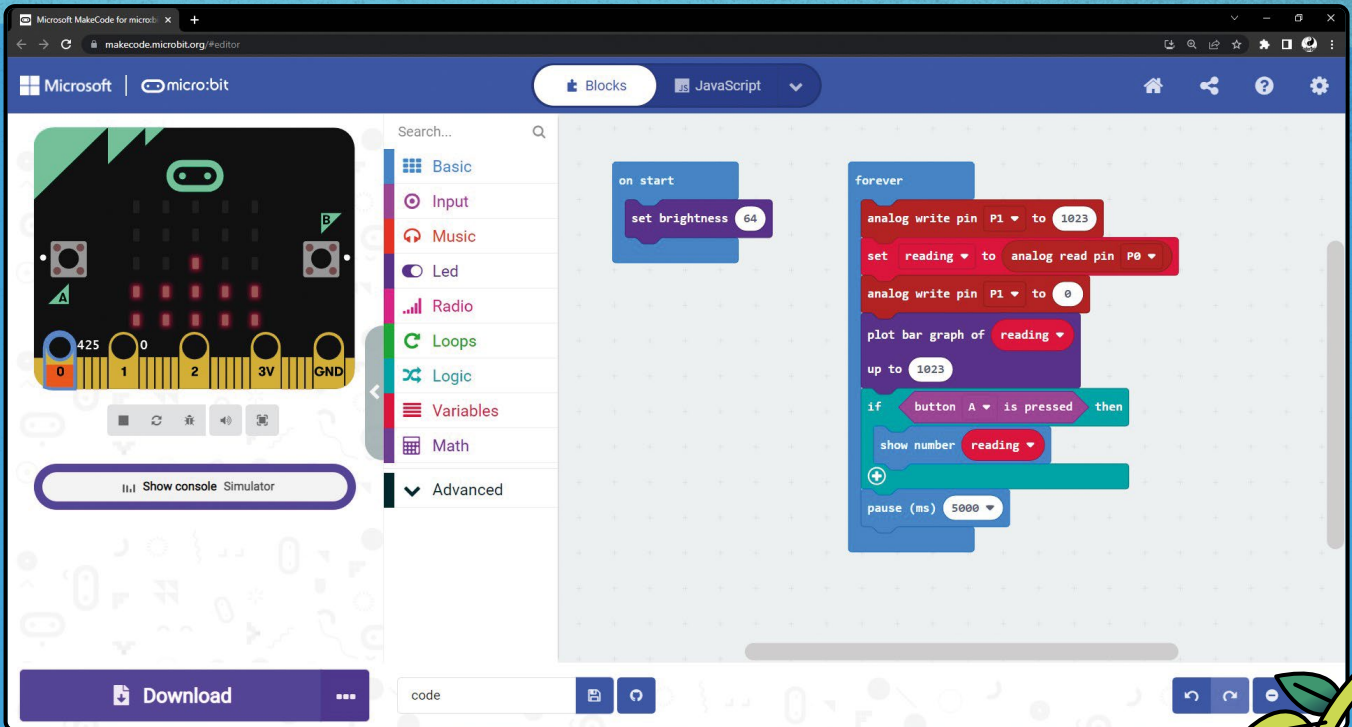
■ A simple soil moisture sensor emulated in MakeCode

> ## COMPUTATIONAL SUSTAINABILITY IS A GREAT CATALYST FOR INTRODUCING PROCESSES SUCH AS DESIGN THINKING

students? The videos and lessons about the goals provide some background and motivation; however, I found the best approach was to have an open discussion with some timely prompts to stay on topic.

For example, what do your students think about when they consider the world around them? What are they concerned about? What would they like to change? Listening to their concerns and gauging their interests should help you decide which project might be the best starting point. If they can't agree on a project, choose a couple of projects to start out with, or let them try a project on their own (with a little guidance). The starter projects have enough scaffolding for independent work. After completing a starter project,

the next step is to personalise the project. The starter projects usually offer some suggestions for next steps, and you can use these suggestions or let students try their own ideas.

Beyond 'do your :bit'
Even when the challenge has ended for the year, there's an opportunity to continue the discussion about computational sustainability. It's also a good catalyst for introducing processes such as design thinking, since you'll probably find that the young people you are working with will often struggle with getting started on a project of their own (**helloworld.cc/designthinking**).

Moving beyond the challenge promotes a discussion about the differences between

prototyping and creating a product. It's not likely the students will actually be able to track a polar bear with a micro:bit (**helloworld.cc/microbittracker**), but thinking about what would be necessary to create a sensor that could be attached to an animal in the wild is a great way to apply computer engineering to concepts. Have fun trying out the 'do your :bit' challenge and get ready to be amazed by your students' ideas! You can enter the challenge at **helloworld.cc/ doyourbit** until 15 July 2022. (HW)

NO MICRO:BITS? NO PROBLEM!

You can still program micro:bits without a device. My favourite way is to use browser-based Microsoft MakeCode for micro:bit (**makecode.microbit.org**). There are a number of tutorial projects, with a choice of block programming, JavaScript, or Python. The best part is the emulator that lets you test your code as you're writing it.

BALANCING COMPUTING & SUSTAINABILITY INTERESTS

Matt Hewlett talks about his year off-grid with his family and how it has impacted his life and his computing teaching

The picture below is where I was living when the pandemic struck. Somewhere in the distant treeline sits Betsy, a camper van that was my family's home for nine months as we wound our way through France, Spain, and Portugal. This particular stop was a permaculture-orientated smallholding in Portugal, and it is very much indicative of the types of place we visited. We lived in areas as varied as fields, eco-hotels, charity homesteads, organic farms, orange and olive groves, and off-grid mountain hideaways. Collectively, the impact these sustainable homes have had on our family has been transformative — to our entire way of life, our parenting, and of course, my outlook as an educator.

Pre-Betsy, I was a computer science teacher and head of department getting good results in a nice school. My family and I are very environmentally conscious and have always wanted to travel. So, after months of research, I handed in my notice and we set off on a sustainable adventure. We lived with communities that were sustainably-oriented, supporting whatever their vision was with tasks such as farming, chopping wood, digging, composting, feeding chickens, cutting brambles, foraging, and planting and farming food. We homeschooled our eldest as best we could and spent almost the entire year existing on an average of £400 a month — and we had never been happier. Eventually returning to work was jarring, but our daily lives have been transformed. I now work part-time, and the sustainable ideals that we struggled to maintain in a busy life are now ingrained in us.

The impact on teaching

I have always struggled with reconciling these ideals with the job I do. Although it is a job I love doing, computing can be an intensely materialistic world to inhabit. Alongside the scarily brilliant technological developments exist just as scary marketing departments promoting built-in obsolescence, needless upgrades, and gadgets that promise silicon happiness; not to mention the environmental disaster that is bitcoin and the energy cost of growing data storage. The toxic side of social media worries me, and my personal view is that screen time is the new smoking — easy to say when you're that family that doesn't have a TV. These are by-products of our area of study, but their cultural capital can make it easier to connect with and teach young people.

When these cultural and environmental aspects of computing are under discussion in the classroom, I am privileged enough to have had experiences of deep connection with the natural world that can inform me and my students. Being able to describe a world that uses little to no digital technology can be powerful. Similarly, a growing insight into our interconnectedness with one another and the natural world is a more than useful tool when helping shape an ethical understanding of the responsibilities we have when developing technology.

This is especially fun when applied to physical computing — for example, I've leveraged my passion to use micro:bits to create litter counters and help control the watering of plants and vegetables. I've also linked the theoretical use of logic gates to controlling an eco-house's power and heating. This is a great way to embed this
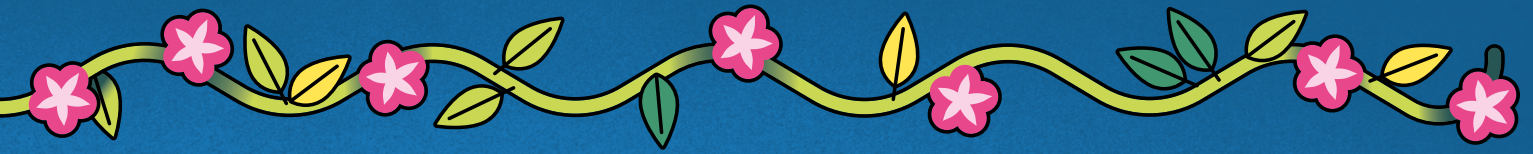
COMPUTING CAN BE A MATERIALISTIC WORLD

part of the GCSE and A-level curriculum, while also introducing concepts such as solar panels, power banks, and other developing technologies we can use in the sustainability fight.

I felt a surge of pride when my sixth-form class realised they could derail whole lessons by bringing up environmental issues they were increasingly interested in. It is, of course, worth noting that today's students are alive to environmental issues in a way that many of my generation never were. I also accept that I'm arguing against what is considered modern life, and appreciate that technology can do amazing things. In many ways, it's making the world a better place. I hope that some of the work I do will be a part of that in its own small way.

Changing views

Having home-educated our children for a year with no timetable has also reinvigorated my view of education more broadly. My eldest had times when he would throw himself into activities; at other times, we would be treated to arguments and tears. As such, we learnt his preferred rhythms for learning, and had the privilege of orientating ourselves around that. Our kids learnt to slow down as much as we did and became more creative when the alternative was boredom. I take memories of this into the classroom with me every day. I try to remember that the best learning happens for different people at different times, that creativity takes time and stimulation, and that feeling joy for a strand of study is potentially more important than the exam grade you get.

Acknowledging and reflecting on my inevitable hypocrisies is also important for my teaching and my day-to-day development. I store data on the cloud. I drive a car. I eat dairy. I like buying Raspberry Pis. These all come at an environmental cost. My daily aim is to try to be a better and more sustainable inhabitant of this planet. Sometimes I'm doing it better, and sometimes I'm doing it worse, but I'm always doing my best. This might be the most powerful thing to focus on — after all, it's no more than what we want from our students. (HW)
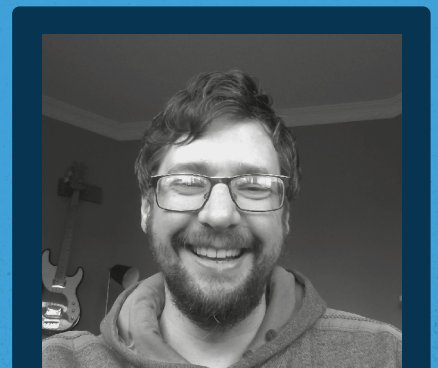
MATTHEW HEWLETT
Matthew is the computing development lead at Fairfields Primary School in Basingstoke, UK. He is a facilitator for the UK's National Centre for Computing Education and a facilitator for Isaac Computer Science. Matthew is also a resource author and an ambassador for the charity digit<all>.
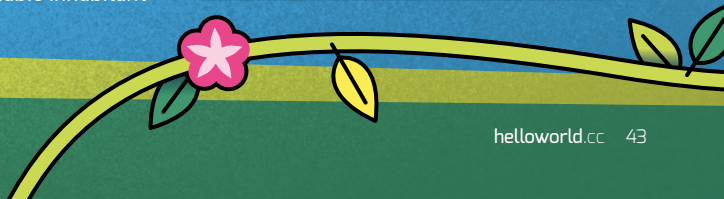
MAKING THE MOST OF A CLOZE QUESTION

Ben Garside discusses how the Isaac Computer Science team reviewed their use of cloze questions to improve learning outcomes

A s content writers for Isaac Computer Science (helloworld. cc/isaac), we meet monthly to discuss pedagogy. This involves reading research papers and reflecting on our own practice. We also discuss how we can structure our content to get the best learning outcomes for the students who visit our site. As we started to add GCSE content to the website (helloworld.cc/isaacgcse), we also added the functionality to create cloze questions. Let's take a closer look at these questions and how they can support students with their learning.

What is a cloze question?

In a traditional cloze question, students complete a passage of text by filling in words that have been deliberately removed. For example:

An argument is any item of data that is passed to a _____. The subroutine must be called with the number of arguments defined by its list of _____.

*Items to choose from:
parameters, subroutine*

Research around cloze questions tends to focus on language learning. If you have ever used a language learning app such as Duolingo, you will be familiar with these questions as they are used for the acquisition and application of new words. However, according to a paper by James Dean Brown, "research on cloze tests has been fairly inconclusive in terms of their reliability and validity, and even in terms of what cloze tests are measuring" (**helloworld.cc/brown2013**).

The last part of this sentence is especially interesting. It got us thinking about what we wanted to measure or achieve by asking this style of question. In the absence of research on cloze questions in computer science, we wrote a couple of cloze questions ourselves, before meeting for a show-and-tell session to discuss our processes.

Our findings

We discovered that the team had written a broad range of questions that moved far beyond the traditional cloze question given in the earlier example.

The question in **Figure 1** requires the student to convert the hexadecimal 9B to its binary equivalent by dragging and dropping the 0s and 1s into the spaces provided. This uses the cloze question functionality on the Isaac website, but it isn't what we think of as a traditional cloze question. The benefit of this question is that the table provides a level of scaffolding to the student, but it could also be argued that we're modelling best practice in how to find the solution.

After looking at all the example questions, we decided that the questions fitted into the categories shown in the table on the left. Here are some examples of questions that fall into these categories:

QUESTION CATEGORIES

| Type	Description
Traditional	A written sentence with missing words; provides options to fill in the gaps
Multiple-choice	A cloze question that includes distractors (i.e. has more options than gaps) or an option to match a word to the correct definition
Modelling best practice	Demonstrates a recommended approach to solving a problem or answering a question
Scaffolding	Students receive support in how to find the answer through the question
Tracing	Students need to follow the execution of the program to find the answer
Alternate conception trapping	Provides options that have been intentionally given to pick up a known or predicted alternate concept
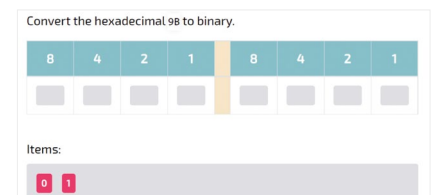
Convert the hexadecimal 9B to binary.

| 8 | 4 | 2 | 1 | | 8 | 4 | 2 | 1 |

Items:
0 1

■ Figure 1

■ Figure 2

1. Multiple-choice questions (MCQs): Figure 2 is an example of a traditional cloze question. However, we classed it as a multiple-choice question due to a distractor being added to the available gap-filling options. The distractor aims to trap a common mistake that a student might make in confusing a relational database with a flat-file database.

2. Modelling best practice: Figure 3 shows another example MCQ question, this time relating to the TCP/IP stack. There are no distractors, but the drag-and-drop functionality means that the student can match the keyword to the provided definition. This tests the student's ability to correctly identify each part of the TCP/IP stack, but it also models appropriate definitions for each one.

■ Figure 3

" **RESEARCH ON CLOZE TESTS HAS BEEN FAIRLY INCONCLUSIVE IN TERMS OF THEIR RELIABILITY AND VALIDITY**

3. Scaffolding: Scaffolding encompasses any method used by educators to provide additional support to students while learning a new skill or concept. A scaffolded task or question supports a student while they are in the zone of proximal development (**helloworld.cc/zpd**). This is the zone students pass through until they are able to solve a problem independently. While in that zone, students need support to achieve the solution.

The example question in **Figure 4** requires learners to construct a subroutine to solve a problem. The question's focus

is on testing the student's understanding of functions and return values. Instead of asking students to write this on a blank page, the question has been scaffolded so that students need only focus on the key learning outcome.

4. Tracing: Tracing is a useful skill for programmers to help them debug their own programs. The question in **Figure 5** requires students to trace the algorithm and complete the partially completed trace table.

The question is similar to a question that could appear in an exam. However, it is

■ Figure 4

partially scaffolded, with some answers already added and drag-and-drop options to provide a level of multiple choice.

5. Alternate conception trapping: Alternate conceptions (often referred to as misconceptions) are learners' beliefs about a concept that are overly simplified or inaccurate. When these beliefs contradict

```
Pseudocode

1  word = "computer"
2
3  FOR index = 0 TO LEN(word) STEP 2
4      PRINT(word[index])
5  NEXT index
6
7  IF index > LEN(word) THEN
8      PRINT("Overshot")
9  ELSE
10     PRINT("Perfect")
11 ENDIF
```

Fill in the trace table by dragging and dropping the correct values from the options given.

index	OUTPUT
0	
	c

Items:

`1` `2` `3` `4` `5` `6` `7` `8` `o` `m` `p` `u` `t` `e` `r` `
`

■ Figure 5

Hexadecimal is often used by people instead of binary for the following reasons:

Reason	True/False
It is easier to read and interpret	
It uses more digits to represent the same value	
Compared to binary, it is more likely that the digit will be written down incorrectly	

Items:

`False` `True`

■ Figure 6

reality or accepted scientific understanding, they can cause confusion for learners and affect their understanding and performance (helloworld.cc/kallia2019). One of the benefits of the Isaac website is that we are able to provide instant and targeted feedback to prevent any alternate conceptions being committed to the learners' long-term memory. **Figure 6**

feedback based on a specific wrong answer. This helps to guide students to the correct answer and to ensure the alternate conception isn't committed to their long-term memories.

During my time as a classroom teacher, I was always a bit wary of using cloze questions, though I couldn't justify why I felt that way. Now that I have had the time to think more deeply about it and read the available research, I see that cloze questions require a certain level of literacy and cognitive ability.

Thinking about my personal experience of attempting to learn a language with Duolingo during lockdown, the missing words required by the cloze questions were often made very obvious by the nature of the question. For example, if the missing

- Trap a misconception
- Help scaffold the problem
- Model best practice

It's a matter of opinion whether all the questions referenced in this article are true cloze questions. But to us, it is important to think first about the learning objective for a question, and only then to think about the tools that can be used to achieve it. As content writers for Isaac Computer Science, we're very grateful to the technical team for providing us with a mechanism to write a wider variety of questions to support students.

If you'd like a closer look at some of our cloze questions, have a go at the gameboard we created at **helloworld.cc/clozegame** for both GCSE and A-level content. **(HW)**

> ## THE BEST QUESTIONS SCAFFOLD, MODEL BEST PRACTICE, OR TRAP MISCONCEPTIONS

is a true or false question based on the advantages of hexadecimals.

This question could have been asked as a multiple-choice question, for example "Which one of the following statements is true?" The benefit of using the cloze question, though, is that students have to evaluate each statement in turn, which means they are less likely to be able to guess the answer. This makes it easier to trap the alternate conception. Using our content editor, we have the ability to predict wrong answers and to provide

word in the sentence was the Spanish for 'house', there would also be a picture of a house on the screen to help. The task was to pick out the word in Spanish to match the word I already knew in English. In computer science education, it's hard to achieve the same thing using a cloze question without making the answer too obvious.

While we were experimenting with cloze questions, the best questions we wrote were the ones that had a specific outcome in mind. For example, to:

BEN GARSIDE
Ben is a learning manager for the Raspberry Pi Foundation. He has worked on the production of the Teach Computing Curriculum, and on online courses including the recently released Introduction to Machine Learning and AI (**@BenberryPi**).

REAL-LIFE TECHNOLOGY

How can we capture our students' imaginations? **Andy Lyon** outlines how educators can relate lessons to practical uses of technology in everyday life

When trying to inspire the students who sit before me, I find nothing grips their attention more than when I relate what I'm teaching to things I have done in my personal life. I have also noticed, when I have mentored non-specialist computing teachers, that bringing in their own practical uses of technology can be incredibly beneficial to their pedagogy in the classroom. On the one side, students become far more engaged and inquisitive, and on the other side, teachers start seeing the many links between teaching computing and the digital world.

In the 'Real-world examples' boxout below, you will find some experiences of 'technology in the wild' that I have previously drawn upon in my classes. As I'm sure you're aware, these examples are just the tip of the iceberg, and none of this was possible just ten short years ago. What will become possible in the next ten years? I for one am excited to see the progress! **(HW)**

REAL-WORLD EXAMPLES

Real-world example	Computing topic links
Travel cards Travel cards can be used for more than just travel. I use them to gain entry to local public parks across the city – a unique and novel use for them. They just need topping up and then swiping at an entry gate, which is all very efficient.	Digital literacy/becoming a digital citizen
Live webcam footage of tourist destinations You can view live webcam footage of many tourist attractions to see how busy the area is, which can help you decide whether or not to visit that day.	Digital literacy/becoming a digital citizen
Car park payment via WhatsApp I'm sure we are all up to speed with the increase in the use of mobile apps to pay for parking. However, I recently discovered that you can also pay for parking by sending a correctly formatted WhatsApp message to the number of the local authority. Mind-blowing! In general, I have seen a vast increase in the number of businesses using WhatsApp as more than just the messaging platform most of us know it as.	Communication and networks
Automation apps When the sun bears down brightly in my classroom and makes the projector screen too difficult to view, automation apps come into play. Blinds can be lowered from the ceiling at the push of a button on a school tablet (or on a daily schedule), which really helps to quash the excuses when students say they can't see the questions on the board to complete the displayed exam questions!	Algorithms and programming
USB ports on public transport While using contactless payment for local transport is quite commonplace in today's connected cities, other transport perks are not as well known. We rely on our devices, so it can often be a saviour when we find USB ports on modes of public transport across the city, including metros, trams, and buses. They have saved me, and many students I teach, numerous times!	The bigger picture
Digital loyalty cards Let's not forget that teacher staple – coffee! On many mornings, I use a digital loyalty card stored in my phone's wallet to collect as many points as possible in the hopes of a free coffee in the near future!	Digital literacy/becoming a digital citizen

> ## THEY BECOME FAR MORE ENGAGED AND INQUISITIVE

ANDY LYON
Andy is from the North West of England and has been a teacher for twelve years. He now teaches internationally at Arcadia School, an all-through British school in the Middle East. Andy's school is an Apple Distinguished School and he is an Apple Teacher and Certified Google Educator (**@mrlyonresources**).

THE COMPUTING QUALITY FRAMEWORK

Claire Rawlinson and **Charlotte Wilson** introduce the Computing Quality Framework and outline how engaging with the framework could benefit your school

The Computing Quality Framework (CQF) is a free and easy-to-use online tool that helps schools review, develop, and monitor their computing curriculum (helloworld.cc/cqf). It supports UK schools in identifying strengths and areas for development in line with the 'quality of education' judgement of Ofsted, the UK's body for school inspections.

The framework splits computing into seven dimensions, as follows:

- Leadership and Vision
- Curriculum and Qualifications
- Teaching, Learning, and Assessment
- Staff Development
- Equity, Diversity, Inclusion, and SEND
- Careers Education
- Impact on Outcomes

Within each dimension, participants are provided with guidance on how to review their current provision. This includes examples of good practice as well as ideas for next steps. Institutions are able to assess themselves against five levels of provision for each dimension using a 'best fit' model, with Level 5 being the highest level. If you wish to apply for accreditation, you must provide evidence for the level you choose to review against, but this is more of a reflection on practice than any type of usual data-type evidence. Once you reach Level 4 across all seven dimensions, you can apply for the National Centre for Computing Education (NCCE) Computing Mark accreditation, awarded by BCS, the Chartered Institute for IT.

Why?

In our primary-school setting, we found the framework to be a useful tool in evaluating where we were as a subject. It gave us the opportunity to consider elements that we were overlooking in our usual annual subject action plan — for example, careers education became a bigger focus for us after using the tool.

The framework not only gave us a nationally recognised accreditation that is a real boost to any school environment, but it also gave us a structure which will allow us to review and analyse our progress more effectively. We now have an even clearer vision of where our subject needs to go in order to achieve a Level 5.

> ❝ **THE FRAMEWORK GAVE US A STRUCTURE THAT HELPED US TO ANALYSE OUR PROGRESS MORE EFFECTIVELY**

How?

During the evaluation process, the framework suggests ways in which you can achieve each level. To give an idea of what a Level 4 provision looks like, we have included some suggestions based around some of the key points from our own evaluation:

Leadership and Vision

Our school leaders show they value the importance of computing as a foundational subject by ensuring:

- A dedicated weekly one-hour computing slot for pupils aged seven to eleven throughout the year
- A specialist computing teacher (do not let this put you off — it certainly helps to provide a consistent high level of provision, but is not essential)
- High staffing levels in computing lessons, with at least two adults in the classroom
- A computing suite resourced with laptops and micro:bits, and four class sets of iPads to be used across our three-form entry school

Curriculum and Qualifications

- Follow the TCC progression (**helloworld.cc/tcc**), along with additional enrichment units and cross-curricular projects to suit your setting
- Make scaffolding available where needed
- Use results from summative assessments to feed into choices for curriculum development and support for students
- Have a key member of staff responsible for maintaining reliable hardware and the latest software — technologies and learning environments change rapidly

Teaching, Learning, and Assessment

- Use formative assessment throughout

■ A student creating a micro:bit musical glove. This is a great cross-curricular opportunity!

CLAIRE RAWLINSON & CHARLOTTE WILSON

Claire and Charlotte work at Anchorsholme Academy in the UK, the first primary school to be accredited with the Computing Mark by England's National Centre for Computing Education (NCCE). Claire is a specialist computing teacher and a computing education advisor to the NCCE (**@CRawlinson01**). Charlotte is a class teacher, teaching students aged nine to ten. She leads and coordinates computing as a subject for primary.

each lesson to address common misconceptions in a timely manner.

■ Use summative assessments at the end of each topic, but ensure these are also a learning experience for the children, for example, self-marking Google Forms or project work.

■ Look out for cross-curricular opportunities. For example, with geography, create a website about Greece; with science, create a micro:bit illumination light display; and with English, create soundscapes in Scratch to accompany story-writing.

Staff Development

■ Engage with Teach Computing training. Key staff in our setting have completed bursary-funded face-to-face courses and free online courses, and followed the Teach Computing Curriculum (TCC) course pathway to be accredited with the Teach Primary Computing Certification (**helloworld.cc/tccourses**).

■ Key computing staff engage with Computing at School (**helloworld.cc/cas**) via forums or meetings.

■ Computing lessons for pupils aged seven to eleven are led by the specialist teacher but also involve the class teacher, ensuring that the teaching of computing is robust.

Equity, Diversity, Inclusion, and SEND

■ Participate in research, such as the Raspberry Pi Foundation's 'Gender Balance in Computing' project

■ Model everything, but ensure reminder notes are available to prevent cognitive overload

■ Engage with the NCCE's twelve pedagogical principles of teaching computing (**helloworld.cc/12pp**)

■ Ensure that all children work on the same task but can find success at different stages

Careers Education

■ Look to your own staff. Our computing teacher speaks about her career as a software engineer, and our network manager hosted a network hunt and invited students to ask questions about his career.

■ Give a context to tasks — for example, by starting with 'You are a graphic designer', 'You are a game creator' and so on.

■ Discuss methods of working in the real world — for example, remote working could be simulated by teams collaboratively working on a shared file from different workstations.

Impact on Outcomes

■ Participate in external competitions or activities, for example Bebras UK (**bebras.uk**) and Astro Pi Mission Zero (**astro-pi.org**).

■ Introduce an enrichment coding class. This could be at lunchtime or in creative curriculum time.

■ Notice and support students at risk of not achieving expected standards by having at least two adults in the classroom.

■ Use Google Classroom, or a similar tool, to set work and monitor attainment and curriculum coverage. This also has a direct impact on progress in digital skills, which will be invaluable in the next stages of education and future careers.

Moving on from our Level 4 accreditation, we are now looking forward to how we can develop our provision further to achieve Level 5. The CQF has given us the focused reflection and evaluation tools to be able to do this with confidence and competence, so see how it can help you in your setting! (HW)

BAKING IMPOSSIBLE ROBOT CHALLENGE!

Karen Scranton and **Shannan Palma** share how they created a television-inspired physical computing challenge to engage eleven- to twelve-year-old girls

E arlier this year, we led a group of eleven- and twelve-year-olds through a robotics activity based on the television programme *Baking Impossible*. On the show, bakers and engineers pair up to create and stress-test devices made from edible materials and robotic parts. Inspired by the show, we gave teams of students a similar challenge, allowing them to apply what they had learnt in a previous problem-solving exercise to a new problem integrating coding and physical computing.

This innovation came about as a result of a partnership between Columbus School for Girls (CSG) and HER Academy. HER Academy is a nonprofit focused on changing school cultures around computer science while prioritising the needs and interests of girls (**theheracademy.org**). In this case, Karen, the computer science teacher at CSG, wanted to incorporate more hands-on robotics and coding into her class. In preparation for the project, we had a practice session to test the idea.

Running the practice session

We played students a short clip from the show to introduce the project. The class discussed the successes and failures of the projects in the clip, paying particular attention to engineering details such as the shape of the wheels and the distribution of weight. We then showed students their materials and introduced their mission. The task was for students to build a cargo container, made of graham crackers and royal icing, to sit atop a robot and hold a marshmallow. Without

losing their marshmallow, the robots would have to pass through an A-frame tunnel, complete a 90-degree turn, and climb and descend a ramp before passing through a cardboard arch at the finish line.

Working in groups of three, students were given a robot and an iPad and told to figure out how they worked. Students then began writing their code and took turns running their robots through the course. We told teams who were waiting their turns to make a plan or sketch a design for their cargo container. Most of them were so excited that they skipped sketching and asked for practice graham crackers to illustrate their ideas. As we were just as excited, we let them go for it. Next time, we'll make sure they sketch first and explain their thinking in writing before they get started. This will reinforce the importance of the design process. Students also asked for rulers to measure the distance between obstacles. We placed tape on the floor to mark the positions of each of the obstacles. One suggestion we got from students that we'll incorporate next time is to mark the starting position of the robots.

In the next class, we gave students trays with all their materials (see boxout) and tasked them with constructing their cargo containers. Teams worked on creating their cargo containers on top of LEGO® platform bases. They used two long rods to secure the platforms to the top of the Dash robots, using the LEGO builder bricks that come with the robots. The rods were needed to raise the height

of the platform above the robots' heads.

We let the cargo containers dry overnight and then allowed for more practice time before the final showcase. One team lost all of their code and had to recreate it at the last minute. They suggested students screenshot their code after each class going forward, so they would always have a record even if the technology glitched. They suggested writing down the measurements between obstacles for the same reason.

Student learnings

What can we say about the atmosphere in the room during that final showcase? Students took pride in explaining the thinking behind their designs and in analysing what went right or wrong. They cheered for each other. They groaned
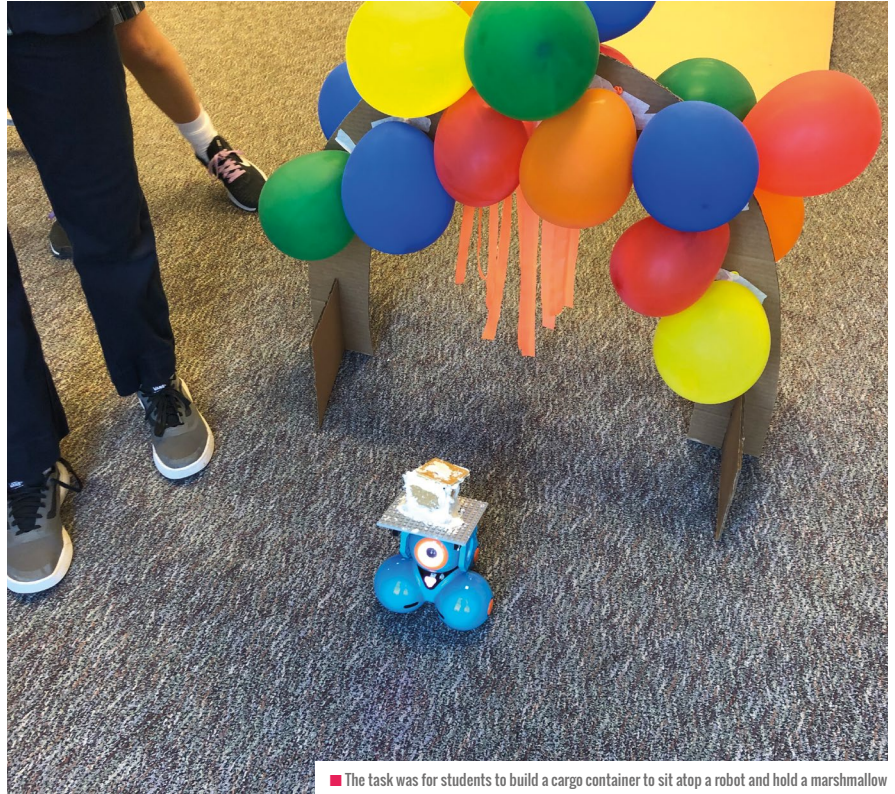

■ One of our goals was to reinforce the importance of the design process in solving problems using technology

■ The task was for students to build a cargo container to sit atop a robot and hold a marshmallow

> " THEY TOOK PRIDE IN EXPLAINING THEIR DESIGNS. THEY CHEERED FOR EACH OTHER. THEY GROANED COLLECTIVELY.

collectively. When one team's robot stalled out, they asked for another try, quickly conferred, figured out the error, and successfully reran the course.

The high-school computer science teacher — who is also the middle-school robotics coach — visited to observe and lead a special guest debrief with students from an engineering perspective. Students picked apart what went well and what they would change. They asked questions about things they'd observed but didn't understand. Some of the robots struggled with the cardboard ramp, for example. Our guest noted that the wheels were picking up dirt and debris from the floor, which was interfering with their ability to maintain friction with the surface of the ramp. Next time, we'll clean the wheels periodically throughout the project.

Students began the next class by reflecting on the project in their notebooks.

They had some great feedback for us, including recommendations to strengthen the ramp and make some of the other improvements we've already mentioned in this article. Our goals for the project were to reinforce the importance of the design process (design, build, test, iterate) in solving problems using technology, and having students practise resolving challenges as a team.

In students' own words, they learnt, "You have a lot of test runs before you go ... You have to make sure it works." They also learnt "to consider and challenge other people's ideas" and "the importance of good teamwork". And their final verdict? "Improvement is always needed. And the teacher should have had snacks. I was often hangry." There were indeed a lot of requests to eat the graham crackers. We will set some aside for snacking next time! (HW)

KAREN SCRANTON
Karen designs and teaches a computer science curriculum that includes robotics, programming, and computational thinking for grades K-6 (students aged five to twelve) at Columbus School for Girls in Bexley, Ohio, USA.

SHANNAN PALMA
Shannan directs the curriculum development and implementation needs of HER Academy (**@shannanpalma**).

EDUCATIONAL TECHNOLOGY: OPPORTUNITIES AND CHALLENGES

Emma Whewell shares the findings of **Alastair Snook's** study on primary teachers' use of educational technology, and the opportunities and challenges it presented during the pandemic

D igital technology for the classroom is becoming increasingly sophisticated, and there are a wide variety of products available to support administration, teaching, learning, and assessment. This article is drawn from Alastair's undergraduate dissertation, which aimed to understand what primary-school teachers in the UK saw as the issues and opportunities around the effective use of educational technology during the pandemic. This article will first explore the findings of existing research, before summarising the findings and recommendations from Alastair's study.

The app generation

According to the United Nations Educational, Scientific and Cultural Organization (Unesco) and research carried out for the UK's National Foundation for Educational Research, most aspects of educational technology adoption during 2020 were influenced by the impact of coronavirus, which saw over half a billion learners worldwide transitioning to extended periods of alternative methods of instruction. This shift has been a learning process for both digital native teachers (those considered to be indigenous to a digital world) and digital immigrant teachers (born before the widespread use of technology)

who are teaching the app generation (young people for whom the use of apps is an integral part of living).

Previous research has found little difference between the educational practices of these two groups of teachers (**helloworld.cc/li2015**). The successful integration of technology is still significantly correlated to perceived teacher abilities, efficacy, access to technology, and in-service training. Interestingly, teachers who are classed as digital natives have "not necessarily become more comfortable keeping pace with the fast rate of change in technology". Alastair's research identified that this was due to their views around how pupils engage with technology and how it can impact their achievement.

The ability of schools to prepare pupils for the modern world relies on the digital upskilling of staff and the integration of teaching and learning opportunities relevant to pupil experiences. However, as of September 2020, most UK teachers believed their pupils were three months behind in their curricular learning, with an increasing disparity in achievement between disadvantaged pupils and more advantaged classmates. Senior leaders reported that 28 percent of pupils had restricted access to technology at home. When asked about their experiences of online learning, nearly three-quarters of teachers felt that the limitations of remote teaching prevented them from maintaining their usual standards (**helloworld.cc/dfe2020**).

© insta_photos/stock.adobe.com

■ The shift to alternative methods of instruction using digital technology has been a learning process for digital native teachers and digital immigrant teachers alike

Pupils with special educational needs and disabilities (SEND): 3.5 | 27.2 | 44.4 | 24.8 | 0.2

Pupils with English as an additional language (EAL): 3.1 | 26.3 | 45.3 | 25.1 | 0.2

Pupils receiving pupil premium support (PP): 4.4 | 31.2 | 41.3 | 22.4 | 0.8

■ Not at all ■ Slightly ■ Moderately ■ Very ■ Extremely
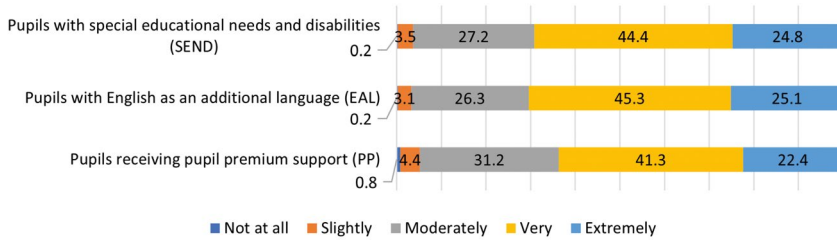
■ A bar chart displaying how beneficial respondents perceived educational technology to be in teaching different groups of primary-school pupils

Findings of this study

Research methods for Alastair's study included online questionnaires and online interviews, with 903 valid questionnaire responses and four interviews collected from primary-school teachers in the UK. The data suggests that most UK primary-school teachers were moderately confident in using educational technology in their teaching. However, the breakdown of responses indicated that longer-serving teachers felt less confident. Many cited their age as a factor in their efficacy, demonstrating an awareness of their limitations and rating themselves as 'consciously incompetent'. Respondents who had been teaching for up to a year were most likely to rate themselves at the extremes of 'very confident' or 'not very confident'. These teachers would largely fall into the digital native community, and Alastair suggests this could be due to unconscious incompetence, as early-career teachers' experience in schools is limited.

Teachers' experience of training, and the support they received for implementing new technology, were also cited as important factors in how well, how confidently, or how much they would use educational technology. Respondents found the speed at which educational technology had developed challenging, and saw student teachers and early-career teachers as valuable sources of information. There was a distinct focus on the impact of Covid on experiences with educational technology, with many respondents stating that schools were ill-prepared for the introduction of online teaching during the lockdowns, but that use of educational technology ultimately mitigated some of the effects of the first lockdown on learning and progress.

Benefits and recommendations

When asked about the perceived benefits of using educational technology with children with additional needs, pupil premium children (disadvantaged pupils for whom the school receives additional funding), and children who speak English as an additional language, 70 percent of respondents said it was 'very beneficial'. Similarly, they believed that educational technology offered specific benefits for these learners, including opportunities for improved communication, self-expression, and access to a range of specifically designed resources. Again, the responses were focused on the problems around providing disadvantaged pupils who lacked technology at home with access to remote learning during the pandemic.

ALASTAIR SNOOK

Alastair teaches pupils aged nine to ten at a primary school in Warwickshire, UK. Before achieving a first-class degree in primary education at the University of Northampton, UK, he worked for Learning Accelerators as an educational technology consultant (**@TheMrSnook**).

EMMA WHEWELL

Emma is an associate professor in teaching and learning at the University of Northampton, UK (**@UoNPEemma**).

The recommendations that have evolved from Alastair's study are threefold. Firstly, an overview of how technology is already being used in settings, and how effectively, offers a baseline from which to develop. This would allow settings to put in place training relevant to the available technology. Secondly, early-career teachers should not be assumed to have the necessary skills, knowledge, and experience to use educational technology confidently and effectively in their teaching. There is scope for working with teacher education providers so that we can better understand their prior experiences. Finally, to keep pace with the changing nature of educational technology, there is both a personal and professional responsibility to take up opportunities to update skills and professional practice. (HW)

RECOMMENDATIONS

This study concluded that schools could take the following steps to increase the opportunities and reduce the challenges of using educational technology:

■ Audit the current provision in settings
■ Develop a whole-school strategy for effectively embedding educational technology and raising the confidence of all teachers
■ Work with initial teacher education providers to understand early-career teachers' experiences of educational technology

STRATEGIES FOR PROGRAMMING IN PRIMARY SCHOOLS

Researchers share ideas from experienced teachers for engaging primary-school students in programming, mastering challenges, and appreciating opportunities

Nowadays, programming is being taught to learners as young as lower-primary school age. So how can we teach it in a way that is age-appropriate, child-friendly, and oriented to students' needs? We asked 200 international primary-school teachers from the UK (70 percent), the rest of Europe (17 percent), and the rest of the world (13 percent) about their experiences of teaching programming to young learners. We've derived strategies on how to master challenges and appreciate opportunities when programming in the primary-school classroom.

USEFUL RESOURCES

Teaching materials:
- **Code.org**
- **purplemash.com**
- **helloworld.cc/tcprimarycourse**

Tools for block-based programming:
- **Analysis of code**
  - CodeMaster (Snap! and App Inventor, **helloworld.cc/codemaster**)
  - Dr. Scratch (Scratch, **drscratch.org**)
  - LitterBox (Scratch, **helloworld.cc/litterbox**)
- **Analysis of output**
  - SnapCheck (Snap!, **helloworld.cc/snapcheck**)
  - Whisker (Scratch, **helloworld.cc/whisker1** in combination with the Whisker extension at **helloworld.cc/whisker2**)

Challenges: low budget

Programming is still a rather recent development in the primary-school classroom. It therefore presents challenges not only to pupils, but also to schools and teachers. A difficulty frequently cited by the teachers we spoke to is a lack of media infrastructure at schools.

If government funds or money from fundraising aren't available to purchase those materials, there are creative ways of using a small number of devices effectively. One teacher suggested we should "pool resources across schools to share, like a hardware library". There are also plenty of collaborative teaching approaches which reduce the need for numerous devices: pair programming, for example, would halve the number of

Challenges: teacher knowledge and skills

The teachers surveyed frequently mentioned concerns about their own cognitive, affective (attitudes, motivation, self-efficacy), and pedagogical skills. An effective best-case strategy to counter these concerns is to attend teacher training, which ideally should be free, of high quality, mandatory, and regularly attended, and should take place prior to teaching. However, this isn't always possible.

Teachers also welcome time for exploring materials and trying out different teaching methods to boost their confidence. One teacher's suggested approach is "peer mentoring among teaching staff", in which experienced teachers support less confident ones. This can take the form of

> **"** ONE DIFFICULTY FREQUENTLY CITED BY THE TEACHERS WE SPOKE TO IS A LACK OF MEDIA INFRASTRUCTURE IN SCHOOLS

devices required. If only individual devices are available, one educator's suggestion of "a dedicated teacher to take out smaller groups of children, rather than trying to teach all at once" is a recommended approach, as is splitting the class into groups. Here, each group rotates through several tasks, and only one task deals with programming on digital devices. An unplugged approach that takes place away from devices altogether can also be considered. Even if you do have resources available, hands-on experiences that don't require digital media literacy can be very fruitful, especially for beginner programmers.

team teaching (planning or teaching lessons in pairs) or lesson observation (giving feedback). Such teacher collaboration might be useful, but is also time-consuming.

To make teaching programming manageable — especially in the beginning — consider using or adapting existing material. While our surveyed teachers knew some websites that provided support, many were unaware of the automated tools that are available to support teaching; you'll find some examples in the boxout.

According to our survey, experienced teachers, when compared to preservice

■ Collaborative teaching approaches such as pair programming reduce the need for numerous devices

teachers, consider programming to be much less challenging for primary-school students. This might be due to the strategies that these experienced teachers have developed over time. For example, they cite avoiding overwhelming children by teaching each concept slowly, in small steps, and in a repetitive manner. As with other subjects, the teachers simplify concepts, relate them to real life, and use examples for effective child-oriented learning.

Appreciating opportunities

Programming at primary school also provides great opportunities that many non-specialised teachers can often be unaware of due to a lack of exposure to the subject. Our surveyed teachers agreed that programming enables the building of new cognitive, metacognitive, and affective skills. In particular, the young age of the students allows them to explore new interests or talents mostly unbiased by stereotypes, which is especially important for underrepresented groups in computer science.

These new skills can be integrated into existing subjects in an interdisciplinary way, thus also having the advantage of supporting a tight curriculum. One teacher suggested that it's important to teach programming as a stand-alone lesson to teach skills, but to also include it in other

lessons to demonstrate how versatile it is. This explains why Scratch (**scratch.mit.edu**) is so popular in primary schools: the basics of Scratch can be taught in two school lessons. Using, for example, a catch game in Scratch, coordinates or angles can be explained in maths classes, as well as basic physical concepts such as motion sequences, gravitation, or measurements of time and distance.

Teachers also frequently mentioned that programming "gives them the opportunity to be confident and create something in a fun ... new and exciting way" — something to which Scratch is again well suited, as it gives pupils the opportunity to create colourful characters and landscapes. These skills can also be implemented with Ozobots (**ozobot.com**), where students control small robots by drawing coloured lines and shapes. Since Scratch is also available in several languages, it can also be integrated into language classes, for example to practise vocabulary or to gain a better understanding of syntax in general.

These cross-curricular links create a win-win situation: not only is programming integrated in a practical way, but existing subjects are also made more varied and educational. Our surveyed teachers particularly appreciated the methodological diversity that programming classes have to offer; they especially recommend coding

LUISA GREIFENSTEIN, ISABELLA GRASSL, GORDON FRASER

Luisa, Isabella, and Gordon work at the University of Passau in Germany as researchers and a professor of computer science respectively.

competitions, small group work, and pair programming to improve students' communication skills and their ability to work in a team.

The acquisition of these cognitive, metacognitive, and affective skills, which also include digital and computing literacy, is the basis for our digitalised world. Therefore, job and career opportunities should also be highlighted at an early age. Suggestions from experienced teachers include inviting experts from science or industry to share their experiences or, even better, holding courses with the children so that they can experience it for themselves.

These ideas and principles suggested by teachers can be implemented within various approaches to teaching programming. We hope that you have gained some new ideas for realising cooperative and creative programming classes in primary schools. (HW)

ONLINE COURSES FOR ALL

Michael Conterio explores how the Raspberry Pi Foundation's online courses support all kinds of educators

Since 2018, I've helped the Raspberry Pi Foundation to produce 33 online learning courses as part of England's National Centre for Computing Education. From the fundamentals of computing (**helloworld. cc/howcompswork**) to cybersecurity essentials (**helloworld.cc/cybercourse**) and helping learners take their first steps into coding with Scratch (**helloworld.cc/**

introtoscratch), these free courses provide support for educators of all kinds.

Although we've got a well-defined target audience for our courses — educators — the feedback we've received shows that there's a wide range of reasons people engage with them. This means that we've had to think carefully about how we can support a wider range of learners with different objectives and needs.

Building knowledge and skills

Unsurprisingly, a lot of people say that they took a course to develop their understanding of the computing topics that it covered — whether that's a topic new to them or a chance to review their knowledge. Another key area is developing confidence; learners talked about how becoming more secure in their understanding had built their confidence in working with their students. Other learners expressed that their increasing confidence meant they were

happier to progress to more complex areas.

As well as content knowledge, another key part of teaching is the pedagogy — the theory and practice of how to teach. In our courses, we include advice for learners about beneficial pedagogical approaches to different strands of computing and topics, and we've found that this is not only helpful to formal educators. Educators working outside formal education have found this advice useful too, with parents and volunteers saying the courses have taught them to think differently about how they can help young people develop their skills — as one participant of the *Scratch to Python* course put it, "As a parent with some coding experience, it can be challenging to find ways to approach coding with kids. This course definitely has me thinking differently about how to help my kids evolve their programming skills."

Of course, we also practise what we preach and apply that pedagogy in the courses themselves, particularly the approaches in line with our key pedagogical principles, described in Hello World's *The Big Book of Computing Pedagogy* (**helloworld.cc/bigbook**). For example, where we feel that a concept or skill would benefit from being demonstrated in a different way ('Add Variety'), we've included diagrams, animations, and videos. To ensure that these are accessible, we still make sure all the key information is also available as text. Although we generally present the concept first, we will often use real-life examples to provide concrete contexts ('Make Concrete') before returning to the concept, in a semantic wave ('Unplug, Unpack, Repack').

We also encourage our learners to

REUSING OUR MATERIALS

These courses are free, and thanks to the Open Government Licence they are released under (**helloworld.cc/openlicence**), the materials are free for you to download, reuse, and adapt. This includes the images, screencasts, and animations that we've created for the courses.

Learn to program in Python	>
Learn to program in Scratch	>
Web design and development	>
Cybersecurity and encryption	>
Get started with Raspberry Pi	>
Electronics and robotics	>
Data and databases	>
AI and machine learning	>
Computer systems and networking	>
Start a coding club	>
Teaching computing to 5- to 11-year-olds	>
Teaching computing to 11- to 18-year-olds (subject knowledge)	>
Teaching computing to 11- to 18-year-olds (pedagogy)	>

Learn to program in Python

Learn Python for free. Discover our online computing courses and learn how to code your own programs.

Programming 101: An Introduction to Python for Educators
Explore the basics of Python and code your first program.
Supported By Google

Programming 102: Think like a Computer Scientist
Take your Python skills further by exploring algorithms, list structures, sorting, and searching.
Supported By Google

Programming 103: Saving and Structuring Data
Learn how to save and structure data in external files, and import files back into your Python programs.
Supported By Google

■ Not sure where to start? Try one of our curated online course pathways

```
1  import random
2  secret = random.randint(1, 20)
3  guess = 0
4  tries = 0
5
6  print ("Try to guess a number between 1 and 20. You have 5 guesses.")
7
8  while guess != secret and tries < 5:
9      guess = int( input("What is your guess? ") )
10     tries = tries + 1
11     if guess == secret:
12         print ("You got it")
13     elif tries == 1 and guess < secret:
14         print("This is your first clue, your guess was too low")
15
16 if guess != secret:
17     print ("Wrong sorry, you have used up your five guesses, better luck nex
18     print ("The secret number was " + str(secret) )
```

■ We support learners by using screencast videos to model how you could complete programming tasks

complete relevant active tasks — for example, programming activities, unplugged activities like reviewing a logic circuit, or design tasks such as decomposing an embedded system. With the active programming tasks, we support learners by using screencast videos to model how you could complete them ('Model Everything'). As well as showing the presenter writing the code, the video includes the presenter explaining the process they go through as they do so, to demonstrate the thinking involved.

Social learning

One of the advantages of the online platforms we use is the possibilities for social learning. To encourage learners to discuss their ideas and what they've made, our courses prompt them to share their creations, thoughts, and questions. For example, a prompt might ask participants if the web today still follows its original open ethos, or ask learners to consider whether bitmap or vector images are more suitable for different scenarios.

To support these discussions and give learners a sense of confidence and achievement, our courses are regularly facilitated by experienced computing educators whose role is to encourage, provide adaptations, and answer questions.

We are keen to see our courses having an impact, so we were also happy to see

how some of our course participants were inspired by them. Some have recruited their fellow educators to form a cohort who are all progressing through our courses together, supporting each other along the way. Others have talked about coming into some of our courses with an interest in a particular branch of computing and finishing with them wanting to turn it into a career, such as Manuela, who participated in the *Introduction to Web Development* course: "I started the course with the thought of

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, playing board games, and performing stand-up comedy.

your teaching pedagogy. When you sign up for a course, you'll get free access for a limited time. If you are a teacher in England, you can sign up to our courses on Teach Computing (**teachcomputing.org/courses**),

> ❝ SOME HAVE JOINED A COURSE WITH AN INTEREST IN A BRANCH OF COMPUTING AND DECIDED TO TURN IT INTO A CAREER

acquiring some more knowledge on how to be a web developer, and I finished it wishing to do this as a profession."

Find your next course
The courses page on the Raspberry Pi website will help you make a choice about which course is best for you (**raspberrypi. org/courses**). You might want to choose a course from a particular strand of your curriculum, such as programming or networking, or you might want to find a course to help you with a particular practice, such as physical computing, or developing

and you will get free extended access to the courses, as well as a certificate when you complete a course.

Whether you're a formal or informal educator, and whatever your reason for joining a course, we will hopefully have a course that interests you and gives you the support you need. Why not join a learner like Aliyah today — she had this to say: "This was such a simple and informative course, I learnt things that I was previously confused about. I'm not an expert but I am proud to say that I now know so much more than before I started this course." **(HW)**

VIRTUAL REALITY WITH COSPACES

Andreas Dengel shares his experiences of using the programming environment CoSpaces for introducing basic components of algorithms by designing three-dimensional virtual worlds

Numerous tools have already been developed to introduce programming and modelling at lower secondary. These range from text-based development environments such as Robot Karol, EOS, and Logo, to block-based development environments and physical computing tools such as Scratch, LEGO® MINDSTORMS®, and micro:bits. A key advantage of these tools is a motivating and age-appropriate approach to teaching programming to students. By successively working through simple building blocks in descriptive contexts of algorithms (for example, statements, sequences, conditions, and loops, and sometimes also variables), the cognitive foundation is laid for high-level languages.

CoSpaces

Another environment that you can use to introduce programming to your students is the block-based development environment CoSpaces (**cospaces.io**). The key difference with CoSpaces is that it lets learners build their own 3D scenes, animate them with code, and then explore them in either virtual reality (VR) or augmented reality (AR).

Similar to game engines for professional game development, such as Unity, there is a 3D-editor mode in which you first design the scene itself, independently of the programming. Learners select 3D objects from a library (or upload their own images, videos, and sounds) before activating the 'Use in CoBlocks' button for any 3D object they want to program. You can select code blocks from various categories in code mode, analogous to other block-based languages. The categories include:

Transformations: Movement, scaling, and rotation of objects
Actions: Animation of objects, speech bubbles, display of quizzes, selection options, and text panels
Events: Simple, predefined conditions such as click and collision events, and keyboard input
Control: Loops, self-defined conditions, parallel execution of code blocks, and scene changes
Operators: Mathematical operations and random numbers

Items: Adding, grouping, and deleting objects
Data: Setting and changing variables
Functions: Creating your own functions

There are two difficulty modes in CoSpaces: beginner and advanced. In beginner mode, the scope of the categories is limited to simple blocks. Switch to advanced mode to include more complex blocks such as string concatenation, displaying videos, rounding, and list operations.

Learners can then test the programmed environment in play mode. In doing so, the view is switched back to the camera viewing angle previously defined in the editor. The camera can be set to be either static (you can only tilt), classically movable ('walking' using the keyboard), or freely movable ('flying mode' using the keyboard), or it can be panned around a defined area. When using the smartphone app, the VR setting can then be activated, leading to a split screen and enabling an immersive experience with mobile VR glasses. As no keyboard input is possible in the VR setting, the actions are defined by gaze direction and clicks on the smartphone screen.

The potential of CoSpaces

You may be thinking, another development environment for programming entry? Block-based? Again? But there are three key areas that distinguish CoSpaces from other development tools: motivation and differentiation; novelty effect and presence experience; and event orientation.

Firstly, CoSpaces sets itself apart through

CLASSROOM RECOMMENDATIONS

- Use a topic that is motivating for all students, such as programming pets
- Start with a short creative session in which students are allowed to design their static environments, then start programming to make their environments interactive
- Introduce every algorithmic component with increasing complexity - for example, single instructions, followed by sequences, repetitions, conditions, and variables (advanced)
- For every component, show one specific example and demonstrate where to find the blocks, but let the students explore more possibilities by themselves afterwards
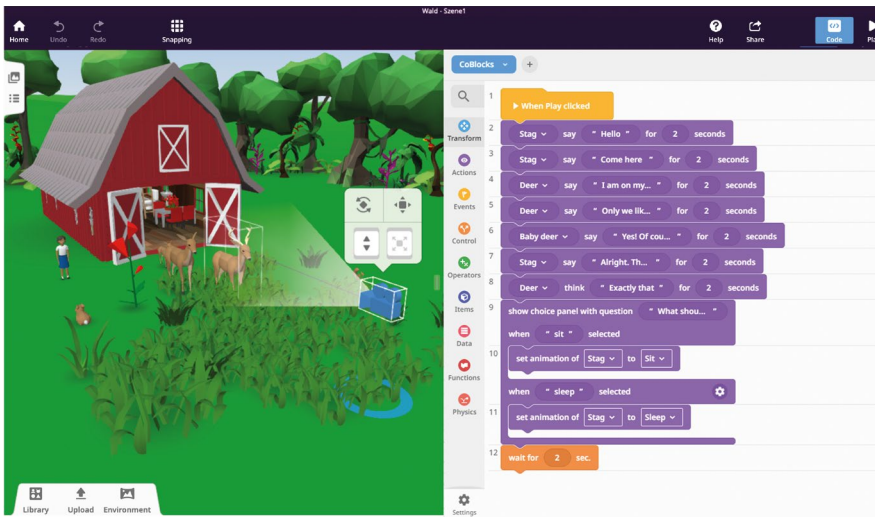
■ CoSpaces provides an editor mode (left) for designing the virtual environment and a code mode (right) to bring the objects to life and create interaction

ANDREAS DENGEL

Andreas is a full professor of computer science education at the Goethe University Frankfurt, Germany. In his research on immersive learning, he investigates the opportunities and challenges of using virtual reality as an educational technology.

3D modelling in the virtual environments. Three-dimensional approaches have a motivating effect on learners. Through the combination of a library of 3D models with animations and the ability to upload your own models, it can serve almost every area of student interest. In particular, open tasks, such as programming an animal to complete tricks, can be implemented individually. For example, in one class I ran, the kids (aged ten to eleven) chose their own pets, ranging from cats to unicorns. These tasks are particularly suitable for differentiation, as higher-performing students can continue to work creatively by adding more models to their virtual environments. In the mentioned class, this led to more questions on how objects can interact with each other, as well as comments such as, "Just look at my penguin fighting this T-Rex!"

Secondly, the use of immersive media such as VR can be really engaging for learners, due to both the novelty of the technology and the 'presence' experience (the feeling of being in a place other than where you are) that is evoked. Even though the visuals are simple, many students report feelings of 'being there', such as one student's exclamation, "I touched him! Oh my, I touched him!" when testing her lion. This immersive experience can make it easier for students to cognitively store and access their learnings at a later date.

Finally, a major advantage of CoSpaces is the embedding of predefined conditions, which are event-based. In classical didactic programming environments such as Scratch, conditions must be implemented in continuous loops for fast response times in event queries, such as mouse clicks or keyboard inputs. For this approach, however, the concept of repetition must be introduced before the conditions, which makes the introduction of conditions unnecessarily difficult. In CoSpaces, there are premade event blocks, such as **If … is clicked …**, which creates a very simple access to conditions, repetitions, and variables. Such an event orientation is also close to established tools of software development: for example, Unity and Unreal Engine work with event-oriented concepts by using C# or JavaScript.

The teaching experiences I've witnessed so far shed a positive light on CoSpaces, with students becoming engaged in programming without even noticing that they are learning about complex components of algorithms. With students' own curiosity being the main driver for learning in this environment, CoSpaces is an ideal introduction to programming and can lay the foundation for the later learning of high-level languages. **(HW)**

■ A student trying out the CoSpaces virtual reality view

DEMOCRATISING STEM EDUCATION

Sidharta Vadaparty describes how he builds open-source lab instruments that facilitate discovery-based STEM learning for older students

From my learning experience, I have observed that STEM learning typically happens in three modes:

- **Classical: students memorise concepts and formulas for tests**
- **Scripted demos: students use kits to perform experiments dictated step by step by the kit or an instructor**
- **Quantitative discovery: students collect quantitative experimental data, construct and observe patterns within the data, and then inductively discover the laws of nature**

Both classical and scripted demo learning have their drawbacks; the former leads to rote learning, and the latter demonstrates concepts well but leaves little room for exploration. Learning through quantitative discovery involves multiple attempts and failures, but, as a result, students understand STEM concepts more deeply and discover the laws themselves, giving them self-confidence in their learning.

To promote this mode of learning, students require reliable devices that enable real-time data collection and in-depth exploration. Unfortunately, many commercial instruments are built with proprietary hardware and software, forever tying schools to expensive vendors. My goal is to democratise STEM learning through open-source, affordable, portable, and modifiable hardware and software, giving students full control to explore concepts.

To achieve this goal, I started a company, GearsNGenes, as a high-school freshman (**gearsngenes.com**). Over the years, I have built scientific devices using products from Adafruit Industries and Raspberry Pi. In this article, I discuss two of my devices that enable hands-on l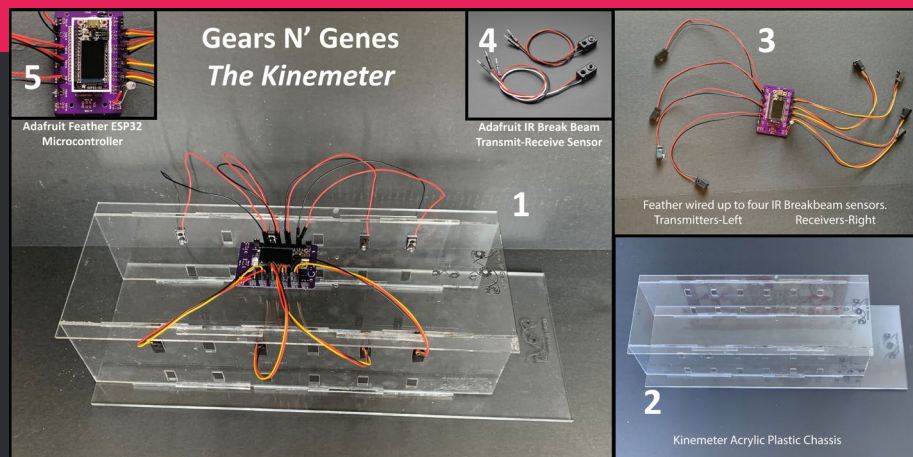earning in linear kinematics and electromagnetism, the bookends of standard physics curricula. Documentation of these devices is publicly available on the GearsNGenes website (**gearsngenes.com**), GitHub (**helloworld. cc/githubgears**), YouTube (**helloworld.cc/ ytgears**), and Instructables (**helloworld.cc/ instructablesgears**).

The devices

The Kinemeter (see the boxout for a labelled photo) is a half-metre-long device that records the times and positions of an object rolling down an inclined ramp. A series of infrared break-beam sensors capture the timestamps of the object passing through these sensors. These timestamps are then used to measure velocity and acceleration. One of the more creative applications of this device is to quantitatively measure the effects of friction on a rolling object — a task rarely explored in college, never mind in high school!
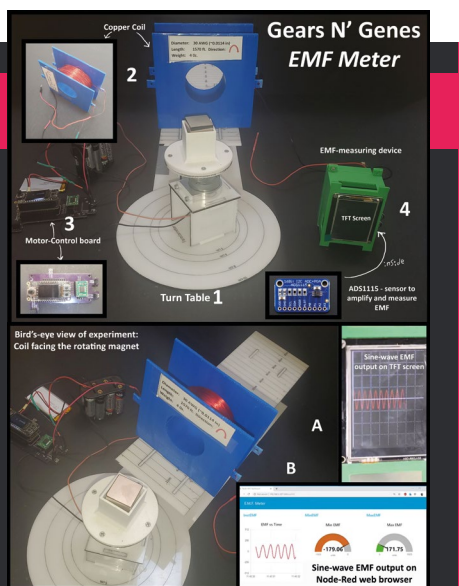
THE KINEMETER

1. **The Kinemeter device as a whole**

2. **The acrylic plastic tunnel chassis to roll objects through**

3. **The electronics of the Kinemeter: Feather microcontroller and infrared break-beam sensors**

4. **A single infrared break-beam sensor pair acts as a tripwire when objects pass between the transmitter and receiver**

5. **The Adafruit Feather ESP32**

Gears N' Genes *The Kinemeter*

Adafruit Feather ESP32 Microcontroller

Adafruit IR Break Beam Transmit-Receive Sensor

Feather wired up to four IR Breakbeam sensors. Transmitters-Left Receivers-Right

Kinemeter Acrylic Plastic Chassis

THE EMF METER

1. The turntable that rotates the magnet

2. A copper coil labelled with wire length, coil diameter, and wire gauge

3. The turntable motor controller

4. The EMF meter device measuring the coil's induced voltage

   a. The EMF published on the TFT screen of the EMF meter

   b. The EMF published to a Node-RED web browser, accessible on a phone or computer

Gears N' Genes EMF Meter

SIDHARTA VADAPARTY

Sidharta is a senior at the Georgia Institute of Technology in the USA, majoring in computer science and specialising in intelligence and devices. He is also the CEO of GearsNGenes, a company dedicated to democratising and expanding STEM education (**gearsngenes.com**).

The EMF meter (see the boxout for a labelled photo) is a collection of devices used to measure the induced voltage (EMF) in a copper coil placed in front of a rotating magnet. A turntable with a controllable speed and direction rotates the magnet. Adafruit's ADS1115 sensor then captures and amplifies the induced voltage (as small as ~0.1mV). This amplified voltage is then transmitted by an Adafruit ESP32 Feather (a microcontroller), using the MQTT protocol, to a Raspberry Pi computer that hosts an MQTT broker. Using Node-RED software, the instantaneous induced voltage is displayed on a web browser. The voltage graphed over time on the browser shows a beautiful sine wave.

Using this set-up, students can quantitatively measure the variations in the induced voltage by changing parameters such as separation of the coil from the magnet; the angular speed of the turntable; and the coil shape and length. They can also publish and share their experimental results. Such experiments lead students to derive advanced theorems we often learn only from textbooks, such as Faraday's law and Biot–Savart's law. Both these devices offer many benefits for democratising STEM education:

- **Affordable:** the devices comprise affordable equipment, such as Raspberry Pi 4s (~ $35), ESP32 Feathers (~$25), and Adafruit sensors (~$5–10).
- **Portable and collaborative:** an entire classroom's worth of data can be shared in one location with MQTT and a single Raspberry Pi broker.
- **Modifiable:** the Feather and Raspberry Pi 4 are open-source, as are the devices I built from them. Students and teachers can easily modify the hardware and software for their classrooms.

Current work

After building devices like these for quantitative discovery, I led workshops to train teachers to integrate the Kinemeter and EMF meter into their physics curriculum. My dream to bring STEM education to underserved student populations was scaled by many orders of magnitude when I met the Agastya International Foundation (**agastya.org**) and other educational organisations in India through my father, Dr Kumar Vadaparty.

My father is my mentor and a fellow proponent of DIY STEM — he is currently an executive director and distinguished engineer at Morgan Stanley, where he is part of their Makerspace program bringing STEM-based interactive workshops to young people. Through his introduction, Agastya and I are now planning to manufacture the devices in India and train the trainers. You can hear more about these efforts from my father's reflections in the boxout. I hope that these devices are one small step in the effort to democratise STEM education — watch this space! (HW)

KUMAR VADAPARTY'S REFLECTIONS

"Having seen the impact of hands-on engineering instruments on Sidharta's technical growth, I am always on the lookout for scaling such experiences to underserved populations.

On a recent visit to India, I met the leadership team at Agastya International Foundation. Agastya's dedication and yearly reach of 2 million underserved children in rural and government schools provides an excellent opportunity for introducing hands-on discovery-based learning.

Professor Radhanand of Gokaraju Rangaraju Institute of Engineering and Technology is also using the EMF meter to not only teach his students about electromagnetism, but also to prototype similar precision instruments in his college's IdeaLab. The All India Council of Technical Education (AICTE), a government agency, awarded funding to Radhanand's college for the IdeaLab as a part of a national effort to provide hands-on vocational skills among Indian engineers.

I am confident that Sidharta's instruments will find an excellent home in these organisations and will shape the futures of many students, just like they did his."

DIGITAL LITERACY AND PHILOSOPHY FOR CHILDREN

Emma Goto explores how the Philosophy for Children pedagogical approach can be used in digital literacy and e-safety lessons

Philosophy for Children (P4C) is a thinking skills programme that was created in the USA in the 1970s by Matthew Lipman and Ann Margaret Sharp. Through philosophical dialogue, it aims to develop caring, collaborative, and creative children with strong critical-thinking skills. In this approach, children choose and develop questions from a stimulus provided by the teacher about a key concept. In this article, I will introduce the structure of P4C enquiries, and then look at its application to digital literacy lessons.

P4C enquiries

During a P4C enquiry, the class sit in a circle. After a short warm-up to get children thinking and working together, the teacher shares a stimulus, such as an image, video, or picture book. The teacher encourages the children to reflect upon the stimulus, think about the related concepts, and share any questions they have. The children then work in small groups to develop questions to put forward for consideration by the wider group, and one question is then selected democratically through some form of vote. Once the class has chosen a question, the children engage in a dialogue about it.

Dialogue is more than a conversation. Dialogue builds ideas and moves thinking forward, and everyone is engaged and thinking throughout. During this dialogue, children share ideas with a partner before bringing them back to the whole group. The teacher encourages children to give reasons, share real-life examples, question assumptions, and spot connections. They build upon the ideas of others in the group, suggesting alternatives and different

> ## ENCOURAGE RESPECTFUL DISAGREEMENT — OUR OPINIONS SHOULD CHANGE WHEN PRESENTED WITH MORE VALID VIEWS

interpretations. Respectful disagreement is encouraged, prioritising rational thinking and the expectation that the children should change their opinions when presented with more justifiable views. As the time draws to a close, the children can share their last thoughts before reflecting on and evaluating the enquiry. Through these reflections, the community can identify what they did well,

how their thinking progressed, and what they might need to consider the next time they enquire together.

Digital literacy applications

This approach is a powerful way of developing children's understanding of the big ideas in a subject such as digital literacy. Within the study of e-safety, for example, we explore a wide range of philosophical concepts such as privacy, security, safety, censorship, ownership, belonging, kindness, bullying, friendship, truth, honesty, trust, freedom, protection, consent, deception, and control.

After introducing an appropriate stimulus, the teacher can facilitate as learners develop philosophical questions about these concepts. Questions are philosophical when they make us wonder, when there are

Concept-rich picture books are a great stimulus for a P4C enquiry

different viewpoints, and when they cannot be answered by asking an expert. For example, should you always tell the truth? How do we know who we can trust? Which is more important, freedom or safety? Should you be free to say anything you want? What is the difference between secrecy and privacy? By engaging in dialogue with peers, children can expand their understanding of these concepts. They can look at them in real depth, from different perspectives, and start to understand other people's views.

Digital literacy stimuli

There is an ever-increasing number of picture books related to e-safety, and these are a great place to start. For example, *Chicken Clicking* (**helloworld.cc/chicken**) tells an engaging story about a chick that makes some questionable decisions when entering a farmer's house, using his computer to make online purchases and agreeing to meet up with a new online friend in the real world. This cautionary tale draws clear links to the concepts of risk, trust, and honesty.

Videos can also be a great stimulus. The video at **helloworld.cc/safetyvid** from the UK's Child Exploitation and Online Protection Centre is a great way to get older children thinking about concepts such as security and online identities. You can also use images and artefacts. The Banksy image *Modern Prison* could encourage questions around freedom, healthy limits, and control. With a group of children used to thinking and questioning together in this way, something as simple as a key could be enough to stimulate a question about privacy.

In summary, you can use this pedagogical approach in computing lessons, particularly in sessions linked to e-safety and digital literacy, to help to develop children's understanding of key concepts. This approach will encourage them to think critically and explore issues from multiple perspectives. (HW)

EMMA GOTO

Emma is an initial teacher educator at the University of Winchester, UK, leading modules related to primary computing and Philosophy for Children. Prior to this, she supported schools with computing as an advanced skills teacher. She is a committee member of the Technology, Pedagogy, and Education Association (TPEA) (**@emmagoto**).

■ Box frame LED art is one creative way to get making!

GOING BACK TO BASICS: PART 4

Rebecca Franks shares her final experiences of taking a group of beginner digital makers through their first physical computing projects

O ver the last year, I have been leading sessions with a group of beginner digital makers who wanted to learn how to use LEDs to make an art project. This group had no prior experience with electronics, and I wanted to provide them with the right scaffolding to develop the confidence to make something they could be really proud of.
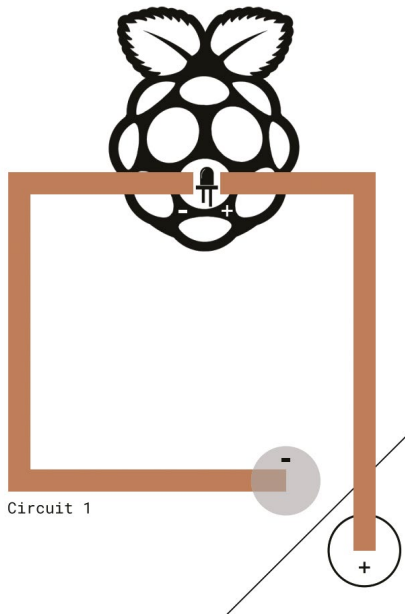
In issue 18 of Hello World, I described the third session of four, which involved the makers creating a circuit that blinked an LED using a classic electronics component called the 555 timer. In this article, I will let you know what happened next, in the final session with my makers.

Go make a thing!

By this session, my makers had a really good understanding of how to create a series and parallel circuit to light LEDs with the use of a switch. They could also blink an LED by creating the 555 timer circuit. By having this foundational understanding, they could now work on embedding their electronics into an object to 'make a thing'.

There is a lot of inspiration online for embedding LEDs into art pieces, and I encouraged my makers to explore these and develop their own ideas. One maker was interested in traditional printmaking and wanted to add LEDs to one of her prints; another had some old glass bottles that she decided to light up with LEDs. Continuing with the upcycling theme, another maker had some used gift boxes that she wasn't sure what to do with, so she decided to practise her skills and embed some LEDs into them.

Circuit 1

A paper circuit template for lighting up the Raspberry Pi logo

LEDs added to a cushion

> ## ONCE YOU REMOVE THE INITIAL BARRIERS AND PROVIDE INSPIRATION, YOU CAN REALLY LET YOUR IMAGINATION GO WILD

The last session with my makers was really enjoyable for everyone. It was great to have a session where we could all sit and make our own projects together. It is so nice to have something creative that you can just fully absorb yourself in. Once you have removed those initial barriers, you can really let your imagination go wild.

What happened next?

Taking a group of beginner makers through an electronics project was also a great opportunity for me to increase my own knowledge of how circuits worked and how they can be embedded into projects. I used this new knowledge as motivation to explore other types of media that can be used to make circuits. For example, we can replace a jumper wire with kitchen foil, copper tape, conductive thread, conductive paint, and much more! This provides so much inspiration, and options for a wide range of projects.

I used this motivation to run a paper circuits maker challenge over a conference call with all the staff at the Raspberry Pi Foundation. We used copper tape and a coin battery to light up the Raspberry Pi logo, before moving on to making other circuits using the tape. Ideas ranged

from adding lights to photographs and drawings, to making the headlights of a toy bus light up. It was some much-needed fun for everyone and they all enjoyed sharing their creations.

How can you get started?

I decided to write this series to help other people who might be a bit unsure of where they should start with their digital-making projects. After learning so much over the past year, I would absolutely say that the hardest thing to do is start. The next hardest thing to do is to keep going. I've overcome these hurdles by taking the pressure off myself to make something perfect that has a purpose. You really don't need to do that at all. Just make a thing — any thing.

Start small and try to place some time in your calendar each month to remind you to keep going. We all get carried away with work and life commitments, but the escapism of digital making is a perfect antidote to the pressures of everyday life. You will be amazed at how much time passes as you get stuck into your creations.

If you fancy having a go at something and you aren't quite sure where to start, then I recommend the new 'Introduction to

Raspberry Pi Pico' path, which is available on the Raspberry Pi projects site (helloworld.cc/picointro). You can see the first lesson activity from this path on page 86 of this issue of Hello World. Happy making! (HW)

REBECCA FRANKS

Rebecca is a learning manager at the Raspberry Pi Foundation. She creates computing resources for teachers and young digital makers. She has over 15 years' experience of teaching computing and a strong passion for inclusion in computing.

MENTORING IN THE GAMBIA

Graham Hastings describes developing a computing curriculum with a school in The Gambia and the obstacles he and the team needed to overcome

R eaders of this magazine will be well aware of digital divisions and the economic disadvantages these can inflict on those on the wrong side of the divide. This became evident in schools during the Covid lockdowns, but there exists a much greater digital divide. A 2021 World Bank report outlined how, although mobile internet availability in Africa has increased to 70 percent of the population, its uptake only sits at 25 percent, and internet coverage is far behind that of other regions (helloworld.cc/digdivide).

In May 2021, John Beer, a colleague I had been working with to provide micro:bit training to primary teachers, introduced me to Mansa-Colley Bojang School in The Gambia, one of the less developed countries in Africa. John is a member of his local Rotary Club, which partly funded the construction of the school. He put me in touch with Claude Sholade Campbell, a maths and science teacher at the school, and the newly appointed head of IT, who had recently set up the first school-based STEM club in The Gambia and was eager to add computer technology and physical computing to the club's offerings. His problem was that computing does not exist as a subject in schools in The Gambia.

The three of us embarked on a project with the initial aim of developing a computing curriculum for the school. My role would be to model the teaching and to provide professional development for teachers, using the skills I had acquired teaching online lessons during lockdowns. Claude manages the work from The Gambia, planning lessons for students and training sessions for the teachers, as well as setting up and maintaining the equipment for online teaching and learning.

Before we could begin, we had to resolve a number of fundamental technical challenges. The school only had a couple of computers, and very limited and intermittent internet access via the mobile phone network. The electricity supply was extremely unreliable, there was no data projector or Wi-Fi network, and the school had no IT budget. Generous donations from Rotary Clubs in the UK enabled us to provide the technology needed to get one classroom in the school up and running with twelve Chromebooks (for classes of 40 pupils or more), a data projector, reliable internet access with sufficient bandwidth for the computing classroom, and an uninterruptible power supply (UPS) for all of the essential mains equipment needed for online teaching.

Creating excitement

We began, not by training the teachers, but with the pupils. We wanted to create a buzz about the new subject of computing. Our thinking was that the pupils would be more receptive than teachers, and feel less threatened by lessons that made use of

> " THE PUPILS' THIRST FOR KNOWLEDGE AND DETERMINATION TO ACQUIRE THESE VITAL SKILLS SEEMED BOUNDLESS

novel technology. As access to computers is limited, we are initially focusing this programme on students aged 16–18, so that they have an introduction to computer science before leaving school.

To make the first lessons engaging and practical, I sent micro:bits out to the school and we spent the first term working on basic computer skills, coding the micro:bits, and introducing some physical computing. During the second term, we focused on programming, starting with Scratch

■ We wanted to create a buzz about the new subject of computing

GRAHAM HASTINGS
Graham is head of computing at St John's
College School, Cambridge, UK.

before moving on to Python for beginners. Admittedly, the content we covered would normally be taught to younger pupils, but many of the senior students had never used a computer. Next term, we plan to ask the senior students to pass on their newly acquired competencies by teaching the content to middle-school pupils. The students in the school's STEM club have also begun to run introductory computing courses to which they are inviting pupils and teachers from other local schools.

We were confident that the pupils' enthusiasm would raise their teachers' awareness of and curiosity about computing, which would motivate them to participate as we moved to the next phase of the project. The students did not let us down. Their thirst for knowledge and determination to acquire what they

recognised as vital skills, both for their futures and the future of their nation, seemed boundless. Now that we have taught two terms' worth of lessons, there is tremendous excitement about computing in the school. This is transmitting to the teachers, who are now asking to join the Teach the Teacher programme that we began in the summer term of 2022.

Teaching the teacher

Our initiative to introduce computing to one school was brought to the attention of The Gambian Education Ministry. The Ministry quickly realised the potential to expand the project as part of the country's overall Teach the Teacher programme, designed to create a national web of six teacher training centres across The Gambia. Once set up and operational, it is envisioned

that the scope of the centres' activities will extend well beyond teacher training. The equipment available will also enable remote adult education for the wider community, delivered from anywhere in the world, on topics as diverse as health, agritech, entrepreneurship, IT, and STEM skills.

Mansa-Colley Bojang School, now that it has been equipped for online teaching and learning, will serve as the first Teach the Teacher hub. The hub's work will be closely monitored and evaluated, and this will then provide a model on which to base additional hubs elsewhere in the country. Raising the funds to set up and equip five additional hubs will be the next big challenge, and we are open to offers of support.

I must admit to having somewhat evangelistic motives for writing this article. This project is the most rewarding thing I have done in my entire 30-plus years of teaching. There is clearly a limit to what I can achieve with my small group of collaborators, but this project is both scalable and reproducible. If reading this article has piqued your interest in either joining our project in The Gambia, or reproducing the project elsewhere, please do get in touch and we will be delighted to involve you in some way, even if just by delivering a few online lessons.

Thank you to my school, St John's College School, for granting me the time to teach lessons to Mansa-Colley Bojang School as part of their outreach programme. You can follow the progress of the project at **helloworld.cc/gambia**. (HW)

WHY DISCRETE ONLINE SAFETY TEACHING IS NOT ENOUGH

Sway Grantham shares suggestions on how to ensure your primary online safety curriculum is fit for purpose

Enter any primary school and you will find strategies for keeping yourself safe being modelled every day of the year. From not running with scissors to negotiating whether it's acceptable to declare who someone else can or can't play with at lunchtime, we are constantly guiding learners to recognise boundaries, be honest about their choices, and develop a little common sense that jumping off something really high may hurt. Yet, more often than not, I see online safety relegated to computing lessons and Safer Internet Days. One-off lessons are simply not enough to safeguard our learners.

We all want to keep our learners safe, and it's challenging to ensure our online safety curriculum stays up to date. I recommend asking yourself the following three questions when reviewing online safety in your school: "What's your context?", "What's trending?", and "Where is it relevant?"

What's your context?

It can be very tempting to start a school year or unit of work with several discrete online safety lessons. These often demonstrate to outside observers that online safety is taught in your school (there's a timetabled lesson for it!) and reassure you that learners have some safety skills before you begin using technology more prevalently. However, what we often see here is a lack of context.

For example, if we ask learners what personal information is, they can tell us it's their name, school, address, and phone number, and they know that they shouldn't share those things online. Then they share a photo of themselves on social media in their school uniform, in front of their house, with their best friend. Without the context, such as different ways and places we share information about ourselves online, learners often imagine that the only time they need to be careful about this is when someone is directly asking for their contact information.

What's trending?

The digital world that your learners are a part of is constantly changing. New apps, games, social media, and other technological advances are constantly being introduced, so teaching the same online safety lessons year-on-year isn't suitable. Keeping up with the latest trends is challenging, but remember that you work with the real experts! Ask your learners what's popular at the moment, how it works, and what safety considerations they take when using that technology. Creating a safe space for them to share openly and honestly gives you the power to support them in making the right decisions to keep themselves safe and know how to handle new situations. Equally, your online safety curriculum should be reactive and allow you to address new issues as they arise.

Where is it relevant?

The conversation around online safety is no longer solely about online safety. It's about how learners handle their digital selves in all aspects of their development, and this requires both computing and non-computing

■ Your online safety curriculum is vital if your learners are to grow up safe and empowered

skills — for example, a learner's self-image and identity, managing relationships with friends, building a reputation, avoiding bullying and isolation, and having a healthy, balanced lifestyle. If we leave these skills to be developed solely in computing lessons, we not only misrepresent their importance, but we also wouldn't have a great deal of time left to teach anything else! Online safety has to be taught throughout the school day in various contexts, whether it's a casual conversation at break about online gaming with friends, or a social story about friends who were unkind either with or without technology.

In the computing classroom

After asking yourselves those three questions, there may still be some topics to be addressed in your computing lessons that are relevant to the content you are teaching. This is a great opportunity to utilise learners' developing technological skills while addressing online safety in a contextual way.

In the Teach Computing Curriculum (TCC), we teach learners about sharing personal information throughout the 'Creating media'

units, as they consider what is and isn't OK to include in the artefacts they create (**helloworld.cc/tcc**). When learning about searching for media or information, learners explore managing information online and the considerations they need to make around copyright and ownership. Across all units, we encourage learners to manage their online accounts and think about the online reputation they are creating, as well as their rights to privacy and security. Each of these examples offers relevant opportunities to learn the skills of online safety and digital citizenship within the context of their computing units of work.

These TCC units are also linked to the Education for a Connected World Framework, created by the UK Council for Internet Safety (**helloworld.cc/connectedworld**). Even if you're not teaching in the UK, this framework is a great resource for understanding the breadth of online safety and digital citizenship. Although it includes recommended learning outcomes for learners aged 4 to 18, it doesn't recommend the most appropriate times and places to teach that content. For the TCC, we chose the most pertinent aspects to computing, but the rest is up to you.

Your online safety curriculum is vital for your learners if they are to grow up safe and empowered. Yet spending more time on topics relating to safety does not necessarily mean greater learning outcomes. Use these three questions as a starting point to review your online safety curriculum and to ensure that it is as prevalent as instructions about not running with scissors! **(HW)**

SWAY GRANTHAM
Sway is a senior learning manager at the Raspberry Pi Foundation. She leads a team developing computing resources for primary teachers (**@SwayGrantham**).
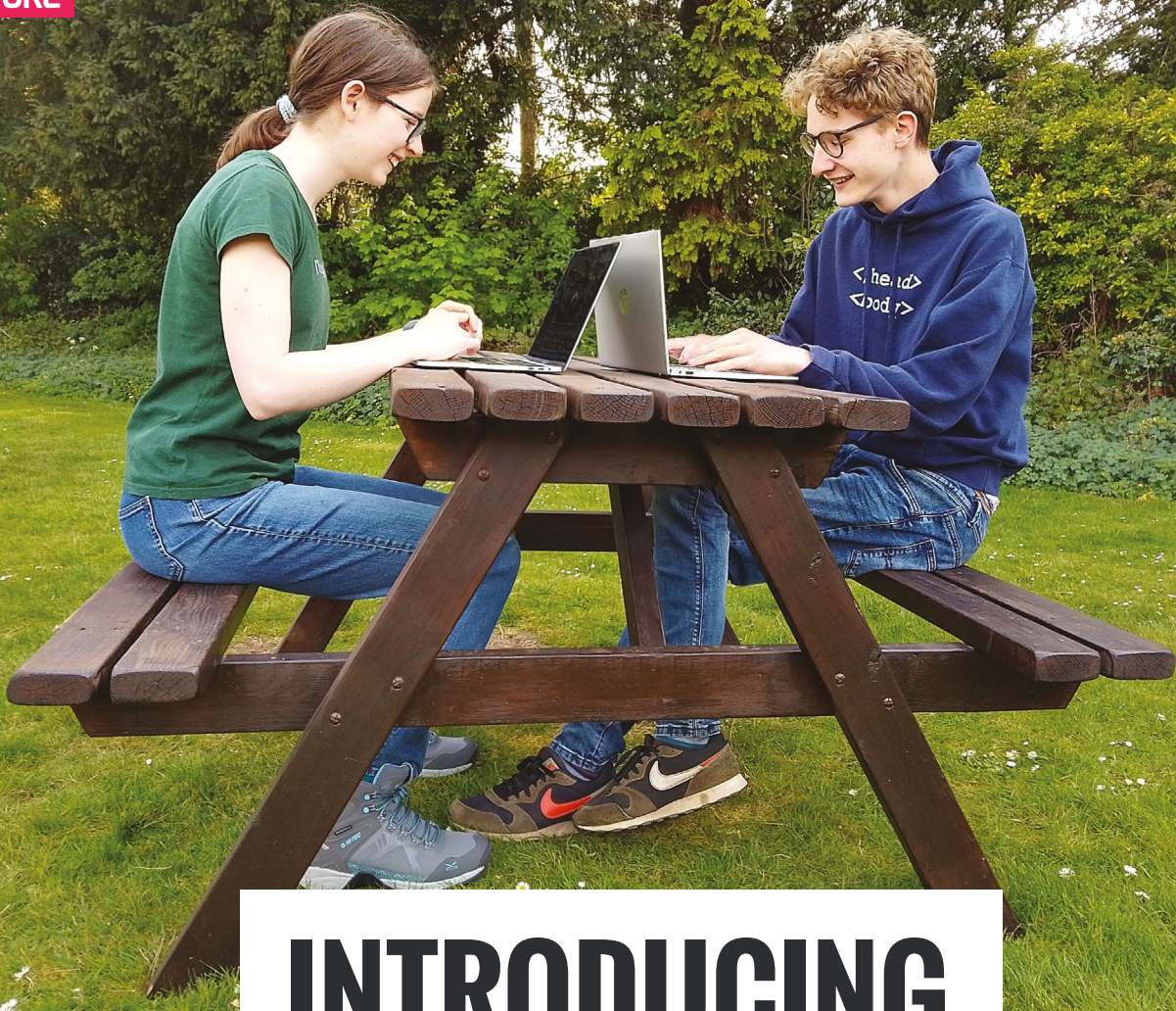
INTRODUCING MISSION ENCODEABLE

Young coding cousins **Harry and Anna Wake** introduce the website they've made to teach other young people to code

Earlier this year, at the age of 14, we launched our new website missionencodeable.com. Aiming to introduce people to coding in Python, Mission Encodeable offers free online tutorials packed with interesting exercises and projects. In this article, we discuss the website and the lessons we learnt in making it, and offer a sneak peek at what's coming next.

Our inspiration

In April 2020, as the first lockdown hit the UK and we were told to stay at home, we had our first Zoom call of many with each other. We had begun coding together using Scratch and knew we were hooked. Experimenting with our new skills, we created and launched an online quiz business on Etsy, which went on to sell hundreds of quizzes and games to families across the world. A few months later, we created our first website, Pen and Paper Puzzles, which we entered into the Raspberry Pi Foundation's Coolest Projects competition (helloworld.cc/penandpaper). We were both over the moon to have our project chosen as a judges' favourite.

We both really enjoy coding, and we wanted to share that passion with others. Sitting on the beach during a sunny holiday together in Anglesey, we came up with the idea for what is now known as Mission Encodeable. We saw a gap in the coding space for a website with more projects and exercises, and we decided that we could be the ones to fill it.

Mission Encodeable

Mission Encodeable is a free online course designed to introduce people to coding in Python. We wanted to make learning to

In the first level, users are introduced to a free online coding workspace (Replit) before we take a deep dive into the fundamentals of the Python language, looking at variables, printing to the console, and maths in Python. The guided class project is a Mad Libs game, which uses string concatenation and variables. The challenge project is a band name generator.

code more enjoyable for our friends, and as the website is aimed at young people, we began by thinking about what we would have wanted to have access to when we taught ourselves to code.

Our idea was that the website could be used either at home by individual learners, or at school as part of lessons or in coding clubs. The website has a number of levels that begin with the key programming fundamentals and progress towards more powerful and advanced concepts. Each level has a variety of interactive coding exercises that let learners apply their knowledge and see which areas they need to revisit. There are two projects on

teachers might find useful to give to their students. Based on the feedback we've had from teachers so far, we're about to start work on a set of notes that offer some suggestions about how Mission Encodeable might be used with groups. Alongside this, we've been given plenty of ideas for other resources, which we're hoping to publish on our website soon.

We chose to focus on Python for several reasons: it's free, it's a widely used language, it's highly in demand (for example, it's used at companies such as Google, Facebook, Netflix, and Amazon), and there's a great community to get involved with.

Lessons learnt

Although we wanted to inspire others to start their coding journey, we also saw the project as a great way to learn more ourselves and teach ourselves new skills. While we were familiar with how to write code in Python, we've learnt a lot in making Mission Encodeable. One of the skills we've had to develop is the ability to write tutorials that are both clear and engaging for all students — this was more difficult than we'd first imagined! We also had to teach ourselves about website design and development. Thankfully, here we had some input from Anna's parents, who

learning about how to share our work more widely, so we're finding out about branding and marketing.

So, what's next for Mission Encodeable? One great piece of advice we've had is to keep testing our website with learners and listening to what they have to say. We've used Mission Encodeable with school coding clubs and are acting on the feedback we've been given to improve our content. At the same time, we're both keen to get more content on the website, and we have started writing the next levels. If readers have any comments or suggestions, we'd love to hear them! Get in touch at **info@ missionencodeable.com**. (HW)

> ❝ WE STARTED BY THINKING ABOUT WHAT WE WOULD HAVE WANTED TO KNOW WHEN WE TAUGHT OURSELVES TO CODE

each level, such as games like hangman and Mad Libs. We've presented these as guided class projects that include instructions, and challenge projects that learners make themselves, drawing on the concepts they've learnt in that and earlier levels.

We've put together some promotional materials for teachers to use in schools, too, including posters and a launch presentation that introduces learners to Mission Encodeable. There's also a Python glossary full of examples that we hope

shared their company's brand guidelines document with us.

Another challenge we faced was the distance — we live nearly 200 miles apart. This meant working remotely using Zoom for weekly video calls, Google Docs for writing ideas, and GitHub for sharing and reviewing code.

Finally, now that the website has been launched, we've been gathering lots of feedback from learners and people in the industry, who have been really generous with their time and advice. We're also now

HARRY WAKE & ANNA WAKE

Harry and Anna are high-school students who are passionate about coding, engineering, and design. When they're not doing their homework, they're busy writing tutorials, designing website wireframes, and planning the next courses for Mission Encodeable.

HAPPY BIRTHDAY CODE CLUB!

Kat Leadbetter looks at how over 4,000 educators and children across the world celebrated a decade of Code Club, and how you can get involved

R eaching your tenth birthday is always an exciting milestone. I remember my own — full of buttercream icing in a soft-play centre, powered by the thrilling knowledge that I had finally reached double digits. For Code Club, reaching our tenth birthday has been no less exciting. Over the past decade we have witnessed thousands of extracurricular coding clubs launching across the world, supporting young people to learn digital skills for free, regardless of their background.

While the Code Club team provides resources and support, the driving force behind these clubs is the educators and volunteers who run them, generously giving their time for free to help young people access new opportunities. From overcoming a lack of access to technology, to powering through the disruption caused by the pandemic, these club leaders create the kind of change that could only have been dreamed of when Code Club started back in 2012.

A global celebration

To help celebrate the existing community and support new people in connecting to our mission, Code Club hosted a two-week birthday party from 16 to 27 May. The aim of this global celebration was to give back, say thank you to our community, and share a decade of stories.

Our first step was to create a free digital pack of resources for clubs to use during their sessions, including a birthday desktop background, printable party hats and badges, and a coding crossword. We also posted inspiring stories from the community on the Code Club Blog (**blog.codeclub.org**), to share how Code Club has shaped the lives of our young attendees and of educators such as Bob Bilsland, a veteran Code Club volunteer and advocate: "If you had asked me ten years ago to stand in front of a room and tell people what I love about Code Club, I wouldn't have done it. The Code Club community has given me the confidence to build those skills."

The highlight of our celebration was a global codealong on 26 May, at which educators and young people were invited to code a new, birthday-themed Scratch project with the Code Club team. Starting in New Zealand and moving to sessions run in Australia, India, Iraq, the UK, Ireland, and the USA, over 4,000 young people registered to join Code Club and create their own party piñata in Scratch.

Anyone could register for the codealong, regardless of whether they usually run a Code Club, and educators from 21 countries on five different continents got involved. Over each hour-long session, young people used costumes, variables, and 'forever' and 'repeat' loops to create a simple game. The project was aimed at young people aged 9 to 13 and was simple enough that beginner coders could get involved.

Ysgol y Ddwylan in Wales (**@YDdwylan**), the Orientations Training Centre in Khartoum, Sudan (**@Orientations19**), and Rudrappasamy School in Chennai, India, are just a few of the groups that took part. Classrooms hosted mini birthday parties and shared photos of their codealong sessions on Twitter using the hashtag **#CodeClubIs10**. Some sessions ran with Hindi and Arabic translations to support

■ Code Club at Park Street CofE School in Cambridge, UK, get ready to celebrate Code Club's birthday

CREATE A PARTY PIÑATA!

Try creating Code Club's fun Scratch birthday project with your students. In this project, you'll use 'forever' loops, costumes, and fun sounds to create a party game. Once you've finished your code, hit the piñata ten times to release a range of tasty birthday treats from around the world, as well as a special message to Code Club! Try it at **rpf.io/party-pinata**.

The thing I love most about code club is making cool stuff!

■ Moamen (aged nine) in Egypt added his own spin to the party piñata project

young people in India, Iraq and the Middle East get coding.

At the end of each codealong, participants were invited to share their finished projects in the Code Club Scratch studio (**helloworld.cc/ccstudio**). Moamen from Egypt, aged nine, shared his project, which he remixed to change colour, play the Happy Birthday song, and share the message, "The thing I love most about Code Club is making cool stuff" when the piñata broke open (**helloworld.cc/moamenproject**). Another young coder from Canada adapted the project to share their own important message: "I love candy and coding!" (**helloworld.cc/bdayproject**).

It takes a village

Ten years ago, when it was first tested in 25 London schools, no one predicted how global Code Club would become. Co-founder Clare Sutcliffe shared her thoughts on what made Code Club so unique at the time: "The thing that was different about

Code Club was that we provided all the materials for anyone to take and run a club. Code Club normalised coding and made it easily understandable for anyone."

Educators agree that Code Club is making an impact. When surveyed, 94 percent of Code Club educators and volunteers said that being part of a Code Club improved young people's computing and programming skills; 92 percent agreed that attending a club boosts young people's confidence to explore and learn with computers, as demonstrated by one former Code Club attendee, Kayla (now aged 17), who told us how attending Code Club has influenced her A-level choices: "I attended Code Club in primary school and went on to take computer science at GCSE ... In my graphic design A level, I'm creating a set of costumes to use in Scratch. I'm hoping to visit my old Code Club so they can use them in a project."

As summarised by Clare Sutcliffe, the community is a key factor that allows Code

" **TEN YEARS AGO NO ONE KNEW HOW GLOBAL CODE CLUB WOULD BECOME**

Club to grow and thrive: "Code Club would simply not exist were it not for the fantastic volunteers who give up their time to inspire children. These talented people use their skills, expertise, and passion to make Code Club deeply impactful to children." We hope to see you at our next celebration — here's to the next ten years!

At the time of publication, Code Club is preparing to run a space-themed codealong for UK Code Clubs as part of STEM Clubs Week, and there will be more codealongs in the coming months — make sure to follow us on Twitter (@CodeClub) and Facebook (@CodeClubUK) for more news. If you'd like to find out more about starting a Code Club at your school, head to **codeclub.org** *and be sure to join the Code Club team at our online workshop on 5 July, where we will be sharing everything you need to know to get started. You can find details of this session and more at* **helloworld.cc/ccevents**. (HW)

KAT LEADBETTER
Kat works as the community manager for Code Club in England. In addition to working with partners and supporting schools to start new clubs, she works to create resources, online training, and opportunities that help existing Code Clubs to thrive.

BROWSER-BASED CODE EXERCISES

Paul Baker and **Gyorgy Denes** introduce their new web tool and discuss how curated web-based exercises can assist students in developing coding fluency

Like most schools, we use a range of web tools to introduce coding principles at all ages. Such online learning tools are easy to deploy in schools and can also enable student access from home. We've found that progressing from block-based web tools to text-based Python coding with older students can prove challenging. Obstacles have included subscription costs to (excellent) providers such as Codio; rapidly changing service provider provision; issues with network and server reliability; and often a different look and feel to the integrated development environments (IDEs) permitted in formal exams. As a result, sometimes simple exercises have

to be downloaded and completed in IDE software when an online tool could have offered better lesson pace and feedback.

In our school, the new Pearson Edexcel GCSE Computer Science specification (for students aged 14–16) is proving to be a suitable preparation both for future coding and the A level, with its on-screen assessment in real coding in Python. However, this does require more hands-on, fluency-based coding tuition than previous specifications. On top of this, teaching tools now also need to reinforce basic IDE skills and features (such as some autocompletion, breakpoints, step-debugging, and variable tracing).

We considered several existing tools, but failed to find a free solution that provides the right mix of online Python coding, guided automarking practice, quick start time, and the feel of a professional IDE. Consequently, we created a new free web tool, PythonSponge (**pythonsponge.com**), which focuses on an IDE-style browser-based experience into which teachers can integrate their own automarked challenges or use the demonstration exercises and tools provided.
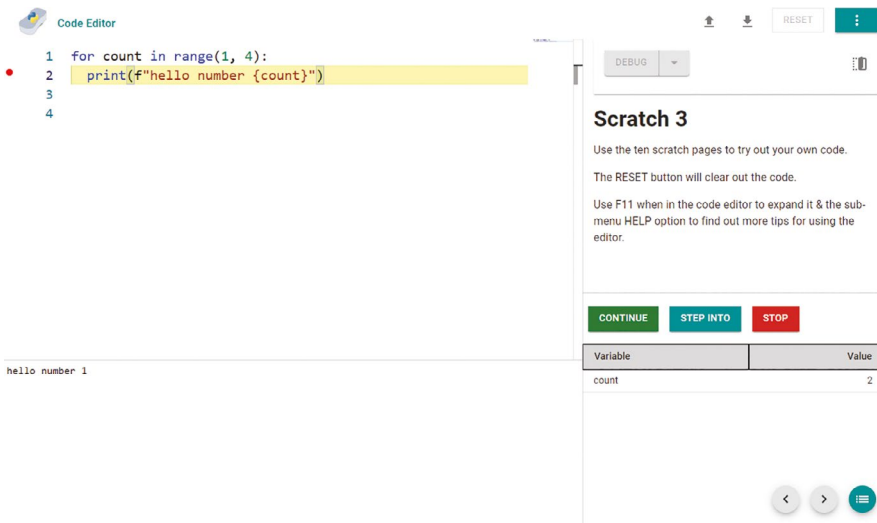
PythonSponge
Our tool builds on the latest open-source projects, including Pyodide (for running Python in a web browser — **pyodide.org**), Material UI (for creating the web interface components — **mui.com**), and the Microsoft Monaco Editor (the same code editor that powers VS Code — **helloworld.cc/monaco**). This builds IDE familiarity and supports the normal shortcut keys and debugging options, including a full command palette. Using simple keyboard shortcuts such as Alt and the Up and Down arrows for rearranging lines of codes make a big difference; line-rearrangement problems effectively become a form of Parson's Problem. For less-experienced learners, we also integrated the js-parsons library to facilitate drag-and-drop challenges, which can be then be 'run' on completion.

While existing tools such as Replit (**replit.com**) and Google Colab (**helloworld.cc/colab**) met most of our goals, the sign-up/sign-in time and account management are often prohibitive. We have designed our framework so that all code runs on the client side in the browser, so there is

WEB TOOL CHECKLIST

A general checklist that can prove helpful when considering a web tool includes:

- Relevance to course coding goals
- Quick-start learning speed
- Opportunity for fluency practice (repetition consolidation)
- Flexible access at school and at home
- Immediate feedback to students on their work
- A variety of useful contexts and learning activities
- Progress reporting to motivate the student
- Progress visibility for the teacher
- Similarity to the environment tools used in formal exams
- Pricing structure and likelihood of long-term provision

■ Automarking provides immediate feedback to motivate students

```
Code Editor
    1   for count in range(1, 4):
●   2       print(f"hello number {count}")
    3
    4
```

```
hello number 1
```

DEBUG ▾

Scratch 3

Use the ten scratch pages to try out your own code.

The RESET button will clear out the code.

Use F11 when in the code editor to expand it & the sub-menu HELP option to find out more tips for using the editor.

CONTINUE   STEP INTO   STOP

Variable	Value
count	2

■ PythonSponge offers a simple IDE-style experience in a browser and supports breakpoints, variable tracing, and step debugging

no need for any installations, there are no logins, and code can be saved/loaded for local retention. The site also autosaves a student's code into their browser's local state each time they run it, for additional protection. In short, we are for a free, minimal-fuss, quick-start experience for students so that we can always maximise the use of lesson time.

Libraries are cached to minimise network traffic after the website is loaded. We have also released a fully offline implementation that can be run on a network without any internet connection, or with an unstable connection or tight security restrictions.

Write your own challenges!
In many existing tools, such as Codewars (**codewars.com**) and futurecoder (**futurecoder.io**), self-marking code exercises, alongside a teacher-authored guide, provide additional structure and guidance to students. We wanted to go beyond this so that schools, or indeed students, feel equipped to build their own sets of challenges or example code. We call these 'books', but they are ultimately just zip files containing starter PY files, guide text, and test cases if applicable.

Students own their code, authors own their challenge books, and our open-source web page facilitates only learning. This means that the time investment schools make is somewhat protected from the fast-changing nature of the web! Over time, we'd love to build an educators' community

> ## IT'S A FREE, NO-FUSS TOOL

in which these zipped tutorials or challenge resources can be created and shared using this open format, free from concerns about web tool lock-in or pricing.

Our ongoing work is also developing less-structured areas of the site with more fun, open-ended challenges (for coding clubs or extension tasks), most of which are written by our older computer science students who come on rotation to our weekly coding club to support and write challenges for the younger students. The code editor also supports file read/write exercises alongside GCSE-style turtle challenges, and facilitates a simple graphical drawing

interface via an HTML canvas for more extension opportunities.

We hope this article provides some useful food for thought on learning Python through web tools, including PythonSponge, and that it offers scope for further school-based development and long-term sharing of banks of coding exercises. Educators can find out more about the site and how to start making their own challenges by using the walk-through video available in the teacher notes area of the site. Please do reach out with suggestions and queries, or to volunteer assistance with this open-source project! (HW)

PAUL BAKER & GYORGY DENES
Paul and Gyorgy both teach at The Perse School, Cambridge, UK. The Perse School runs the annual, free Perse Coding Team Challenge coding competition as part of the wider **ukctchallenges. org** competitions pathway (**@p_d_baker**, **gdenes.com**).

The following lesson plan is taken from the eight-lesson 'Impacts of technology' unit in the GCSE stage of the Teach Computing Curriculum (TCC). Through a range of real-world examples, this unit takes students through how to identify a specific type of technology impact - for example, legal, cultural, privacy, environmental, and ethical. Learners will then progress to identifying stakeholders who are impacted by technology, and learn how these impacts are experienced, negated, or adapted to.

ABOUT THE TCC

Teach Computing

The Teach Computing Curriculum is part of England's National Centre for Computing Education and supports students aged 5-16.

Every unit of work in the TCC contains: a unit overview; a learning graph to show the progression of skills and concepts in a unit; and lesson content, including a lesson plan, slides, and formative assessment opportunities. Find them when you sign up for a free account at **helloworld.cc/tcc**.

THE ENVIRONMENTAL IMPACT OF TECHNOLOGY

AGE RANGE

14-16 years

Introduce learners to the positive and negative impacts that technology can have on the environment

OBJECTIVES

✔ Explain the environmental effects of the use of technology

This is the sixth lesson in the TCC's 'Impacts of technology' unit. In this lesson, learners will discover the reality of technology's impact on the environment. They will also learn about how technology, if utilised strategically, can protect the environment and can be a preserver rather than a destroyer. **(HW)**

> IN THE FIRST HALF OF 2020, TOTAL GLOBAL E-WASTE AMOUNTED TO 25 MILLION TONNES
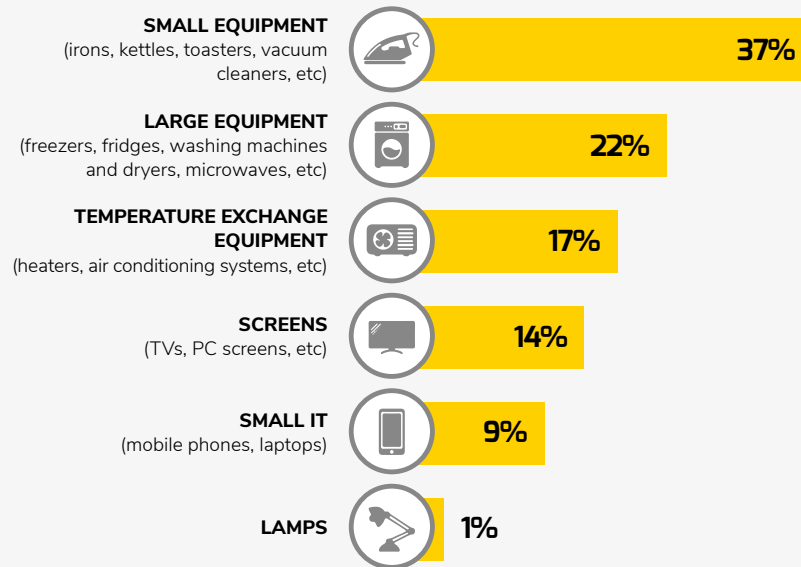
ACTIVITY 1: TECHNOLOGY AS A PRESERVER 25 MINUTES

Give individual learners the name of a technology that is or has been beneficial to the environment, for example renewable energy; electric cars; the sharing economy, with companies such as Uber and Airbnb; and energy-smart homes, including devices such as smart thermostats and motion-activated lighting.

Allow learners 15 minutes to research the technology and prepare a short presentation that would be suitable for their peers to learn from. They could include a description of the technology; its benefits to the environment; and any other pertinent information such as potential downsides.

While learners are working, choose one or two presentations that may be suitable for sharing with the class for the plenary.

GLOBAL E-WASTE IN 2020

SMALL EQUIPMENT
(irons, kettles, toasters, vacuum cleaners, etc) — **37%**

LARGE EQUIPMENT
(freezers, fridges, washing machines and dryers, microwaves, etc) — **22%**

TEMPERATURE EXCHANGE EQUIPMENT
(heaters, air conditioning systems, etc) — **17%**

SCREENS
(TVs, PC screens, etc) — **14%**

SMALL IT
(mobile phones, laptops) — **9%**

LAMPS — **1%**

© Southampton University, BBC.co.uk

■ **Figure 1** E-waste statistics for 2020

ACTIVITY 2: NEGATIVE EFFECTS OF TECHNOLOGY 12 MINUTES

As an introduction to this section, explain that technology can have negative effects on individuals and the environment — for example, the consumption of resources and energy by technology production and use, and the effects of the disposal of e-waste.

Ask learners why e-waste is harmful to the environment. Some potential answers could include that the waste is generally made of non-recyclable materials and rare chemical elements, which are being depleted. Answers could also include the harmful effects of pollution caused by disposal, and the health of the recyclers through exposure to toxins.

Next, share some statistics about technology waste. According to TheWorldCounts, the total e-waste for the first six months of 2020 amounted to 25 million tonnes. To put this into context, one compact car weighs 1.5 tonnes. You could show learners the statistics in **Figure 1**, or display the dynamic counter of global e-waste at **helloworld.cc/ewaste**.

Play learners the video at **helloworld.cc/ewastevideo**, showing the harmful effects of the physical disposal of computers. Ask them to take some notes as they watch it. When they have watched the video, hold a discussion and ask them the following questions:

■ Who are the public health researchers working with? Answer: informal recyclers who are recycling to make a living for themselves.
■ What is the goal for the informal recyclers? Answer: to recycle elements, e.g. hard drives, or to melt elements down to recover precious metals, e.g. silver and gold.
■ What does the element recovery process also release? Answer: toxic chemicals, e.g. lead, cadmium, and mercury.
■ What are the effects of this process on the individuals? Answer: elevated presence of heavy metals in their blood, and injuries from flying metal and glass.

PLENARY ACTIVITY: PRESENTATION REVIEW 10 MINUTES

Choose one or two of the presentations produced by the learners in Activity 1 and ask them to present them to the class.

RELEVANT LINKS

TCC 'Environmental impact' lesson: **helloworld.cc/environmentalimpact**

OIL SPILL CLEANER-UPPER

Relevant contexts are crucial when delivering project-based learning activities. This project allows students to experience sustainability in practice by creating a boat drone to clean up an oil spill

AGE RANGE
11-14 years

REQUIREMENTS
- micro:bit (v1/2) with access to makecode.microbit.org
- Servo driver board (automation:bit)
- Battery pack and header wires
- Materials - card, glue, mini screwdriver, sponge
- 2x servo motors
- Paddling pool (optional)

This project challenges learners to design and create an autonomous boat drone to clear up an oil spill. The tasks require a combination of design, engineering, electronics, and programming skills to complete them successfully. The activities are best done over a full day with mixed groups of four. While fiddly to set up, this project allows learners to develop a real-world artefact that solves a tangible problem. **(HW)**

ACTIVITY 1: INTRODUCING THE CONTEXT AND EQUIPMENT   10 MINUTES

To get started, introduce the environmental context to students. Spend a minute discussing the damage that oil spills do to marine ecosystems, to engage learners and show them why solving this problem matters. Explain to them that scientists have developed a new material that can absorb up to 90 times its own weight in spilled oil, which can then be squeezed out like a sponge and reused, raising hopes for an easier clean-up of oil spill sites (helloworld.cc/spillmaterial).

Next, introduce the task. A group of marine scientists have asked learners to develop an algorithm that could be used on a boat drone that they must also build, to drag a sheet of this smart material to clean up an oil spill. Present students with the success criteria for this project and allow them time to ask questions. By the end of the project, students should have built a floating oil spill cleaner-upper boat drone that starts with a button press; can autonomously navigate over an area; and can clean up an oil spill by dragging a smart material.

If your class has not used micro:bits before, you should also demonstrate how to get code from Microsoft MakeCode to the micro:bit, as well as how to locate the additional blocks needed for the peripheral board.

ACTIVITY 2: DECOMPOSITION AND IPO   5 MINUTES

Learners often rush the design element when problem-solving. It is worth spending some time modelling how they can use an Input, Process, Output (IPO) table to decompose the problem and to help design a solution both programmatically and physically (see Figure 1).

The input and output for this problem are simple, as the drone boat should start with a button press and follow a preprogrammed path. Creating the algorithm for the movement (the process) is the tricky part and will require some thought.

Explain to learners that they don't need to worry about distances at this point.

Oil spills can be small or large, and the product just needs to be able to clean an area autonomously, regardless of its size. Ask them to think instead about how the size of the area can be changed.

Input, Process, Output (IPO)

Input	Process	Output
Button press	Algorithm to control the movement of the boat drone in a path to clean up oil:	Servo motor control

■ **Figure 1** IPO table to aid decomposition and solution design

ROB LEEMAN
Rob is educational solutions manager for the Arm School Program, developing educational resources and online training to support computer science teachers. Rob also chairs the CAS Curriculum and Physical Computing working groups (**school.arm.com**, **@ArmEducation**).

ACTIVITY 3: ITERATIVELY BUILDING THE ARTEFACT 2-4 HOURS

Some learners will now be able to start the design and build with just the context and success criteria; others will require more structured guidance and support, and this scaffolding should be used at your discretion. For example, you could show a simple model solution of the boat (see Figure 2) and use open questioning to draw out why this solution works and what learners will need to consider when building their own artefacts.

You can provide more support for both the physical computing and programming tasks in this project (see 'Differentiation' box for suggestions), but remember that project-based learning is not meant to be a didactic learning experience. At this point, learners should have everything they need to work on their projects. Your role is now to facilitate the learning, and you should avoid providing solutions. Instead, encourage exploration and experimentation — enjoy this bit!
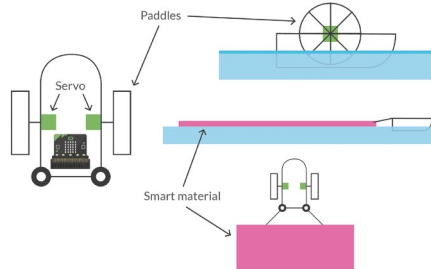

■ **Figure 2** A model hardware design solution for the boat, to scaffold learning

SUCCESS CRITERIA

- Build a floating oil spill cleaner-upper boat drone that starts with a button press
- The product should be able to navigate autonomously over an area
- The product should be able to clean up an oil spill by dragging a smart material

SAFETY REQUIREMENTS

micro:bits and peripherals are not waterproof! Electricity and water do not mix well, and you can permanently damage the electronics by getting them wet.

ACTIVITY 4: TEST TIME AND FINAL THOUGHTS 5 MINUTES

Testing the artefact and iteratively improving it is a core feature of project-based learning and is what makes these projects come alive. Get students to test the seaworthiness of the boat drone and whether the code does what you intended — you could test it in a paddling pool and use a sponge to simulate the smart material. This testing will make this experience engaging, as well as building students' resilience and confidence with STEM.

Combining smart material and autonomous drones is just one way that technology can help protect the environment. Ask students if they can think of other ways that technology can help, and what we can do to ensure that technology doesn't add to the problem, for example as e-waste.

■ Testing the boat drone out is what makes the project come alive!

DIFFERENTIATION

For Activity 3, you can support learners through active and open questioning by steering them towards a suitable solution but letting the solution come from them. There are example code snippets and a build-along on pages 67-72 of the freely available micro:course book (helloworld.cc/armmicrocourse).

You can stretch learners with these additional success criteria to extend the project:

- Adapt the program so that the navigation is done using the micro:bit's compass, so that it turns precisely 90 degrees and can stay on course more accurately.
- Add a moisture sensor so that the boat drone only starts cleaning when it's in the water.
- In large oil spills, many drones would be used at once. Adapt the program so that the boat drones don't collide with each other (learners could use radio blocks for this).

RELEVANT LINKS

- micro:course book (pages 67-72): **helloworld.cc/armmicrocourse**
- A video of this project in action: **helloworld.cc/oilvideo**
- A free professional development course on project-based learning: **helloworld.cc/pblcourse**
- All the Arm School Program resources: **helloworld.cc/armresources**
- If you are interested in physical computing, join the CAS Physical Computing Working Group: **helloworld.cc/casphysical**

ELECTRICITY GENERATION

Electricity generation is a fun, eco-conscious project that introduces learners to data analysis and visualisation using real data

Electricity generation is part of the 'Protect our Planet' path of projects from the Raspberry Pi Foundation. The series aims to show learners how they can protect the environment using the power of technology, and draws on several of the 17 United Nations Sustainable Development Goals (SDGs). Using step-by-step instructions with personalisation choices, learners create an animated data visualisation chart in Scratch in which they compare the type and amount of natural resources used by three countries across the world to generate electricity. The project explores renewable and non-renewable energy sources using data compiled by the International Energy Agency. **(HW)**

ACTIVITY 1: DEFINE YOUR COLOURS 15 MINUTES

In Scratch, you will first investigate electricity generation sources for New Zealand, before creating a key for all its energy resources. Open the starter project (**helloworld.cc/ electricitystarter**) and you should see a grey background with a chart title. When you click the green flag, an animated column appears showing New Zealand's use of resources. You can investigate the column by hovering over the bands of different colours.

You will now choose the colours to represent each of the resources on your graph.

Select the 'nonrenewable' sprite and click on the Costumes tab. In the centre of the Paint editor is a square that has no fill colour. Click on the Fill colour slider tool and create a colour of your choice by altering the Color, Saturation, and Brightness sliders. Select the Fill tool and click inside the square. You'll see the first key square on the stage change to your new colour.

You now need to change the resource in the New Zealand sprite so that the column is drawn using your new colour. Click on the New
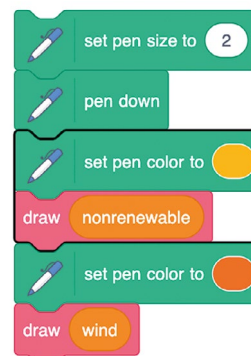

■ Figure 1

Zealand sprite and then the Code tab. There is some code already included in the script area; find the block that sets the pen colour for 'nonrenewable', as shown in **Figure 1**. Click on the coloured yellow circle and select the colour picker tool. Move the mouse pointer until you hover over the 'nonrenewable' sprite in the key, then click to select that colour.

Run the project; the non-renewable resource for New Zealand now draws in your chosen colour!

ACTIVITY 1: DEFINE YOUR COLOURS 15 MINUTES (CONT.)

Hover over the column, though, and you'll notice the 'non-renewable' speech bubble is missing. Select the Pointer sprite and find the **Non-renewable if touching colour** condition in the code (**Figure 2**). Click on the coloured circle, then select the colour picker tool. Move the mouse pointer until you hover over the non-renewable coloured square in the key, then click to select that colour.

Next, choose colours to represent the other resources. Remember to change the key square, the New Zealand sprite column colour, and the Pointer sprite speech bubble colour for each resource.
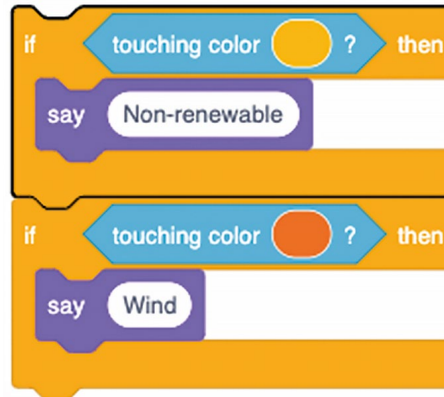

■ Figure 2

ACTIVITY 2: ADD A UK COLUMN 5 MINUTES

Next, you will add a UK column and populate it with data for comparison purposes. Duplicate the New Zealand sprite and rename it 'UK'. If you run the program, it looks as though nothing is happening, because the UK column is drawn underneath the New Zealand column. Go to the UK sprite Code tab and find the **go to x: -200 y: -140** block (**Figure 3**). Change the value of x to -20 to move the column along the horizontal axis.

Run the program to see the two side-by-side columns. Find and update the label code at the bottom of the script to move the label along the x-axis, and change the text to say 'UK'. Run your program again to check the labels and positions.
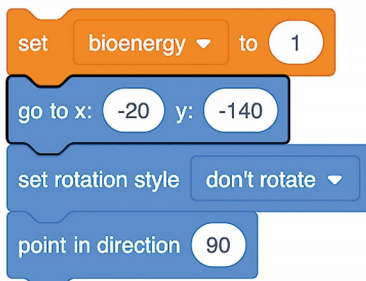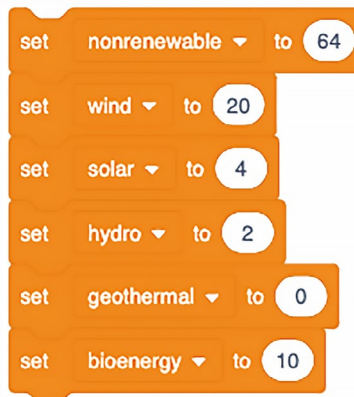

■ Figure 4

The columns currently look identical because the variables contain the same values for the UK and New Zealand. Find the **set variable** blocks in the UK script and change the values to match **Figure 4**. The values are a percentage of the country's energy generation, so the total should add up to 100.

Run the program to compare resources. You may notice that the UK uses more bioenergy but far less hydropower than New Zealand. How do the countries compare on their use of non-renewable resources?

■ Figure 3

ACTIVITY 3: CHOOSE A THIRD COUNTRY 10 MINUTES

Resource type	Brazil	Canada	Iceland	India	Ireland	Norway	Singapore	S.Africa	US
Non-renewable	18	34	0	81	64	2	99	94	83
Wind	9	5	0	5	32	4	0	3	7
Solar	1	1	0	3	0	0	1	1	2
Hydro	63	58	70	11	4	94	0	2	7
Geothermal	0	0	30	0	0	0	0	0	0
Bioenergy	9	2	0	0	0	0	0	0	1

Figure 5

You may have noticed that the country sprites do not have costumes. This is because you only need to see the related actions of the Pen and the speech bubble, not the sprite itself. Create a new sprite by clicking on Choose a Sprite and selecting the paint editor.

The table in **Figure 5** lists a number of countries and the percentage contribution of resources they use to generate electricity. Choose one of the listed countries and rename your new sprite to match.

Click on the Code tab, then the Variables blocks menu. Make a variable for each energy resource type, name each new variable, and click on the **For this sprite only** radio button so that the values are stored for this country sprite only. Remember to include all the resources even if the country doesn't use them, as use changes over time.

The new variables will appear on the stage. Click on the ticks next to the variable names in the Variables blocks menu to hide them from view.

Add code to set your new variables to the percentage of energy used by your chosen country. The code is shown in **Figure 6**; change the values to match the use of your chosen third country.
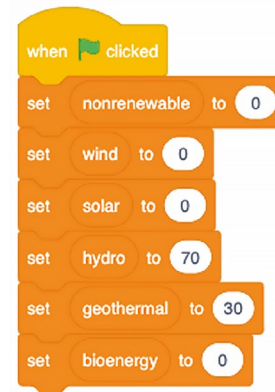

Figure 6

ACTIVITY 4: BUILD A PEN 10 MINUTES

Add **Motion** blocks to your code to move your third country sprite to the correct position, ready to draw the third column (**Figure 7**).

The **Pen extension** blocks are used to draw on the stage. Click on the **Add Extension** button and select Pen. This will add a set of **Pen** blocks to your blocks menu. Add **Pen** blocks to the end of your code and set the pen size to '2' before placing the pen down so that it can draw on the stage as the sprite moves. The pen will change colour as it draws each resource. Add a **set pen colour** block, then click on the coloured circle. Select the colour picker tool and click on the first square in your key to set the colour for Non-renewable energy. Repeat for the next energy category

ACTIVITY 4: BUILD A PEN 10 MINUTES (CONT.)

in your key and continue until you have a pen colour block for each resource type. You will need an additional **set colour** block that sets the pen colour to the stage background colour. You can select this colour by making sure your project isn't running, then hovering anywhere on the stage.

Finally, add **Pen** blocks to the start of your code to clear the stage when your project begins and lift the pen up so that it doesn't draw as the sprite moves to the start position.
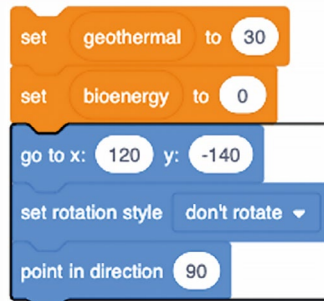
■ Figure 7

> ## THE SERIES SHOWS LEARNERS HOW TO PROTECT THE ENVIRONMENT USING TECHNOLOGY

ACTIVITY 5: DRAW THE THIRD COLUMN 10 MINUTES

Since you have set energy variables for your third country and created a pen, now is the time to build code to draw your third column.

In Scratch, you can make your own block that starts a new script whenever it is used. You can give your new block a name and add any blocks into the script for your new block.

Go to the My Blocks blocks menu and click on the **Make a Block** button before calling your new block **draw**. Your draw code is going to use the values stored in your resource variables as an input. To enable your block to use a number as an input, click on the 'Add an input number or text' button and type 'amount', then click OK.

You will see that a new define block called **draw** has been added to the Code area for your third country sprite. Add a 'repeat' loop from the Control blocks menu and drag an **amount** block from the define block header in the Code area into the repeat block value. Add **Motion** code blocks to tell the sprite how to draw one line, then position it ready to draw the next line (**Figure 8**).

Insert one of your new **draw** blocks under each **set pen color to** block in your pen code.

Insert the matching resource variable into each **draw** block so the amount stored in each variable is drawn (**Figure 9**). For the final background colour, type the value 1 to add a hidden line at the top of your column.

Lastly, add a label to the third country column with the name of your chosen country. Then run your project to see the animated data visualisation for all three countries!
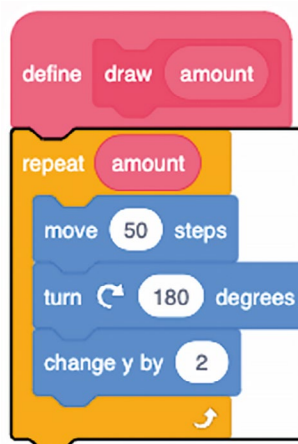
■ Figure 9

■ Figure 8

LIZ SMART
Liz is a learning manager in the informal learning team at the Raspberry Pi Foundation.

CREATE AN LED FIREFLY

Use Raspberry Pi Pico to create an LED firefly that blinks just like fireflies do in nature

REBECCA FRANKS

Rebecca is a learning manager at the Raspberry Pi Foundation. She creates computing resources for teachers and young digital makers. She has over 15 years' experience of teaching computing and a strong passion for inclusion in computing (**@FranksberryPi**).

Raspberry Pi Pico is a low-cost microcontroller that beginners and experts can use to develop electronic products. A microcontroller is a tiny computing device that can run code and interact with electronics components (such as buttons and lights). It is usually designed to complete a single task, and doesn't have an operating system.

This project uses MicroPython, which is a version of Python specifically designed to be used with microcontrollers. It also uses the new MicroPython library called 'picozero'. This is a beginner-friendly library for using common electronics components with Raspberry Pi Pico.

The LED firefly project is the perfect introduction to physical computing and is the first project in the new Introduction to Raspberry Pi Pico learning pathway. **(HW)**

AGE RANGE

9-13 years

REQUIREMENTS

- 1 x soldered Raspberry Pi Pico
- 1 x yellow LED
- 1 x resistor
- 4 x jumper wires
- 2 x servo motors
- Thonny installed on your computer

ACTIVITY 1: SET UP YOUR RASPBERRY PI PICO

5-10 MINUTES

To save lesson time, it's a good idea to complete this step on each of your Raspberry Pi Picos before the lesson. You can then test that they work with a student account to avoid any potential technical issues in class.

Start by making sure that you have the latest version of Thonny (**thonny.org**) installed on your network/computers. This will ensure that the firmware required for Raspberry Pi Pico is available to you.

Next, attach your Raspberry Pi Pico to your computer using a micro USB data cable.
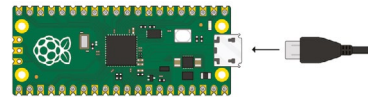
Open the Thonny editor and change the language in the bottom-right corner from Python 3 to MicroPython (Raspberry Pi Pico). This will prompt the firmware to be installed. Finally, install the picozero library to your Raspberry Pi Pico. You do this by going to Tools > Manage Packages and searching for 'picozero'.

ACTIVITY 2: LIGHT THE RASPBERRY PI PICO LED   5 MINUTES

Ask your class to plug in their Raspberry Pi Picos and open Thonny. They should then select File > New to open a new coding window. They should type the following code:

```
from picozero import pico_led
```

```
pico_led.on()
```

Next, ask them to select File > Save As and save the file in their work areas. This means that your learners will always have access to the file, even if Raspberry Pi Pico is unplugged. If your learners choose 'Raspberry Pi Pico' at this step, the files might get lost from lesson to lesson.

They should now press the 'play' button and take a look at their Raspberry Pi Pico. The onboard LED should light up (see image), meaning that they have connected everything correctly and that the firmware and picozero have been installed.

At the moment, the LED just lights up and stays on. Ask your learners to add in some new code that will make the LED switch on for one second and then switch off:

```
from picozero import pico_led
from time import sleep
```

```
pico_led.on()
sleep(1)
pico_led.off()
```

If your learners have used Python before, the sleep function might already be familiar to them. Before running this code, ask the class to predict what might happen when the code is run.

ACTIVITY 3: LIGHT YOUR FIREFLY   10 MINUTES

Timings for this activity depend on how much you have prepared prior to the lesson. Learners will need an LED with a resistor attached to the positive leg. They will also need two socket-to-socket jumper wires connected to the negative leg of the LED and the other end of the resistor (see image).

You can do this using solder and shrink wrap, or you can twist the resistor around the LED leg and secure it with electrical tape. Learners should be able to do either of these activities under supervision; the electrical tape option is the most accessible. Alternatively, to save time you can prepare this step for them before the lesson.

Once learners have their LED, resistor, and jumper wires attached, ask them to disconnect their Raspberry Pi Picos and attach the LED. They should attach the positive leg to the GP13 pin and the negative leg to the GND pin next to it.

Once they have attached their LEDs, then can reconnect their Raspberry Pi Picos to the computer. Ask learners to add a comma and 'LED' to the end of their import statement at the top of their code:

```
from picozero import pico_led, LED
```

They should then add the following code to the bottom of their programs:

```
firefly = LED(13) # Use GP13
firefly.on()
```

Keeping the code that switches the onboard LED on and off is a good idea because it helps with troubleshooting. When they run their code, they should see that the LED they have just attached now switches on.

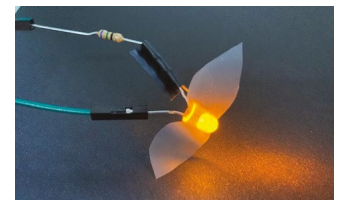## ACTIVITY 4: MAKE THE FIREFLY BLINK   5 MINUTES

The code `firefly.on()` simply switches the LED on. In order to make the firefly blink like a real firefly, this line of code needs to be replaced with a 'while' loop.

Ask learners to replace the code `firefly.on()` with the following code:

```
while True:
    firefly.on()
    sleep(0.5)
    firefly.off()
    sleep(2.5)
```

Again, before they run their code, ask them to make a prediction about what might happen. Once they have this part working, you could ask them to add some wings to their fireflies using sticky tape.

ACTIVITY 5: ADD A SWITCH!   5 MINUTES

Give your learners one pin-to-socket jumper wire and one socket-to-socket jumper wire. They will be using this as a switch. To create the connection, they will place the pin end of one jumper wire into the socket end of the other jumper wire (see image).

Ask them to attach one jumper wire to GP18 and another jumper wire to the GND pin next to it. It doesn't matter which way round they do this. Before they can code with a switch, they need to `import Switch` from the picozero library. Ask your learners to add `, Switch` to the end of the import statement on line 1.

They then need to create a variable for the switch, in order to state which pin the switch is on. They should place the code below underneath the line of code that says `firefly = LED(13) # Use GP13`.

```
switch = Switch(18) # Use GP18
```

Now they need to update their 'while' loop so that it only blinks the LED when the switch is closed. They should replace all the code within their 'while' loop with the following code:

```
if switch.is_closed: #
Wires are connected
        firefly.on()
        sleep(0.5) # Stay
on for half a second
```

```
        firefly.off()
        sleep(2.5) # Stay
off for 2.5 seconds
    else: # Wires are not
connected
        firefly.off()
        sleep(0.1) # Small
delay
```

Ask your learners to make a prediction about this new piece of code before they press 'run' and test it out! They will need to open and close the switch to see their firefly switch on and off.

SAFETY REQUIREMENTS

LEDs can blow if too much current flows through them, which is why a resistor is highly recommended for this activity. You may wish to attach the resistors to your LEDs prior to the lesson.

Always reinforce with your learners that they must disconnect their Raspberry Pi Pico from the computer before attaching any components to it.

Soldering is not required for this activity. However, if you do choose to use soldering irons with your class, please ensure that appropriate safety measures are followed.

RELEVANT LINKS

- Introduction to Raspberry Pi Pico path: **helloworld.cc/picointro**
- Pimoroni's starter kit: **helloworld.cc/picointrokit**

EMBEDDING COMPETITIONS INTO THE COMPUTER SCIENCE CURRICULUM

Gemma Coleman catches up with secondary computer science teacher **Monir El Moudden** to hear his thoughts and advice around computing competitions

Entering competitions motivates, challenges, and boosts learners' skills and knowledge. They allow students to develop not only technical skills, but also soft skills that employers value, such as teamwork, problem-solving, and time management. This mix of technical knowledge and professional skills enriches the curriculum and can really drive up the quality and the profile of computer science.

■ For the Amazon Longitude Explorer Prize, our competition finalists made an app that aimed to solve issues around food waste

How do you embed competitions into the curriculum?

There are five steps to doing this. The first step is to be committed to running competitions for your students. There are a range of competitions open to schools, and each has its own structure and rules for entering. Once you have identified a suitable competition for your learners, you need to set up and communicate the course of action to colleagues, parents, and students. Step two is to get started! Set targets, allocate resources, and choose a competition lead who will ensure entries are registered in advance. Step three is to create momentum. Celebrate the events with awards ceremonies, have an article written for your school website, and build excitement on social media. Step four is to embed an ethos that students can expect to participate in one or more competitions over the year. These events will need to be put in the calendar. The final step is to build your relationship with the organisers and collaborate with other participating schools to generate ideas and get the best out of the experience.

Why do you sign up for competitions?

Over the years, I've found that children enjoy taking their learning outside the classroom. Competitions provide an environment for ingenuity, and have allowed my students to explore practical experiences in solving real-world problems in the areas of sustainability, climate change, recycling, and food waste. The experience gained from competitions

is unmatched, offering students an exciting opportunity to bring their learning to life. With the world becoming increasingly dependent on technology, introducing competitions into the curriculum brings out the best in our learners.

Why do your pupils sign up for competitions?

Every year, the children I work with are given the opportunity to compete in some of the most challenging environments. The competitions inspire them and help them make the transition from their education to their future careers. Competitions encourage collaboration and build confidence, growing a classroom community through strengthening relationships and developing problem-solving skills.

- **Amazon Longitude Explorer Prize:** this competition aims to support students through a product life cycle to produce a solution that can help solve some of the world's biggest challenges. Our competition finalists made an app that aimed to solve issues around food waste. (**helloworld.cc/amazoncomp**).
- **The Perse Coding Team Challenge:** this competition involves students submitting short code solutions to a range of increasingly complex challenges (**helloworld.cc/perse**).
- **British Informatics Olympiad:** an annual computer programming competition for UK secondary schools and sixth form colleges, which gives students a chance to represent the UK in the prestigious International Olympiad in Informatics (**olympiad.org.uk**).

that it is regular. Clubs can be led by students, which gives them ownership over their learning and allows them to explore ideas without the formality of assessments. Children enjoy the independence as they learn specialist skills and realise that they won't always get things right, as well as gaining resilience and a winning mentality.
2. **Generate ideas:** students will spend a lot of time thinking about the problem, but they always need to consider the people they're producing their products for. I use prompts such as, 'If you had £100 to improve the environment, what would you make?' or, 'If you were trapped on a desert island, what would you depend on to survive?' News articles are another great way to get students thinking about current issues in the world.

Embedding competitions in the curriculum doesn't necessitate extra work, and many aspects of the curriculum can be delivered or enhanced by competitions. Setting up assignments or tasks may take some time initially, but the process gets a lot easier in the long term, as pupil motivation and developed skills increase each year. **(HW)**

> ## " COMPETITIONS ENCOURAGE COLLABORATION, BUILD CONFIDENCE, AND DEVELOP PROBLEM-SOLVING SKILLS

What do judges look for in a winning team?

The judges of many of these competitions will have industry experience, and look for teams that demonstrate real enthusiasm for their ideas. They want to see the product the teams have made, but ultimately, they also want to see the process taken to produce these great ideas for the world.

Are there any competitions you particularly recommend?

These are a few that we've enjoyed (and got to the finals of!):

- **PA Raspberry Pi competition:** this provides participants with a Raspberry Pi kit that they use to create a product that could solve a problem facing society (**helloworld.cc/pacomp**).
- **NASA Space Apps Challenge:** an international hackathon for coders, scientists, and designers. Teams engage with NASA's free and open data to address real-world problems on Earth and in space (**helloworld.cc/spaceapps**).

- **BAFTA Young Game Designers:** this competition is open to aspiring game developers, who use their creativity to design and create an original game (**ygd.bafta.org**).
- **The Bebras Computing Challenge:** this gets students developing their computational thinking skills through a set of timed activities (**bebras.uk**).
- **The Oxford University Computing Challenge:** an invitation-only event for students who achieved a top 10 percent score in the UK Bebras Computing Challenge. This competition aims to develop these students' skills further by producing programmed solutions to computational thinking problems (**helloworld.cc/oucc**).

Do you have any advice on preparing a team for a competition?

Absolutely. Here are my top tips, which have worked well in my school:

1. **Start a club:** this could be at lunchtime or after school, but it is important

MONIR EL MOUDDEN
Monir has been teaching computer science for over twelve years and currently works at an independent school in London, UK. In the last two years, he has embedded competitions in the curriculum and taken ten teams to competition finals, including the BIO, OUCC, and The Perse Coding Team Challenge (**@monirelmoudden**).

DEDTECH: SUSTAINABLE HARDWARE USAGE

Alan O'Donohoe describes practical, sustainable solutions for making the best use of dedtech — technology we no longer see a use for

There's something wonderful about unboxing a new gadget and setting it up, whether it's a mobile phone, tablet, or computer. However, you may have noticed that gadgets don't stay shiny and new for long. Technology is getting replaced at a faster and faster rate. Every day, notifications pop up on our devices telling us that updates are needed. Unfortunately, these updates often no longer support older devices, which shortens their lives. We're only witnessing very small incremental improvements in function, power, and performance over purchases made a couple of years ago, but this hasn't stopped equipment manufacturers trying to devise new features to make their devices appear compelling in their sales and marketing campaigns.

When it comes to managing the use of technology in education, particularly older hardware, schools therefore face some common challenges:

- **End of life:** managing the disposal of obsolete equipment ethically
- **Practical experiences:** difficulties teaching students about the maintenance of hardware, software, and networks
- **Sustainability:** providing meaningful experiences to help students understand more sustainable approaches to acquiring and using technology
- **Digital divide:** supporting students who lack access to suitable devices out of school
- **One size fits all:** replacing PCs with tablets and Chromebooks, and how that can restrict learning opportunities

In this article, I will share some practical and sustainable solutions for making the best use of dedtech — a term I coined to describe the technology we consider to be as good as dead, whether it's damaged, decommissioned, dirty, discontinued, or otherwise done for!

Switch to Linux

First things first — we may be able to salvage dedtech simply by switching its system software. I have done this with donated hardware. As part of my work, I lead digital maker workshops in community spaces. It's rare for anyone attending these workshops to bring a device with them, so I have a set of donated netbooks I use. Despite being ten years old, these netbooks are well built and in near-perfect condition. They had been purchased new, shortly before the donating school updated all their devices to a later version of Windows. This change meant that these obsolete devices were no longer of use to the school. For my workshops, though, I simply installed a version of Linux available for free from the Raspberry Pi website, which includes Scratch, Python, Google Chrome, and other software; it's called Raspberry Pi Desktop for PC and Mac (**helloworld.cc/rpidesktop**).

Linux is an operating system (OS) that is offered as a free alternative to paid operating systems such as Windows and MacOS. Many different versions of Linux are available — Ubuntu and Mint are popular in education, and they work particularly well on older computers. It's possible to boot Linux on a PC from flash memory (for example, a USB thumb drive). This might be unreliable for regular use, but can help you decide which version to use without making lasting changes to the PC. It always amuses me how much workshop participants enjoy using these devices, which otherwise would have been gathering dust in a cupboard. This is just one example of how some dedtech has been repurposed to meet alternative needs.
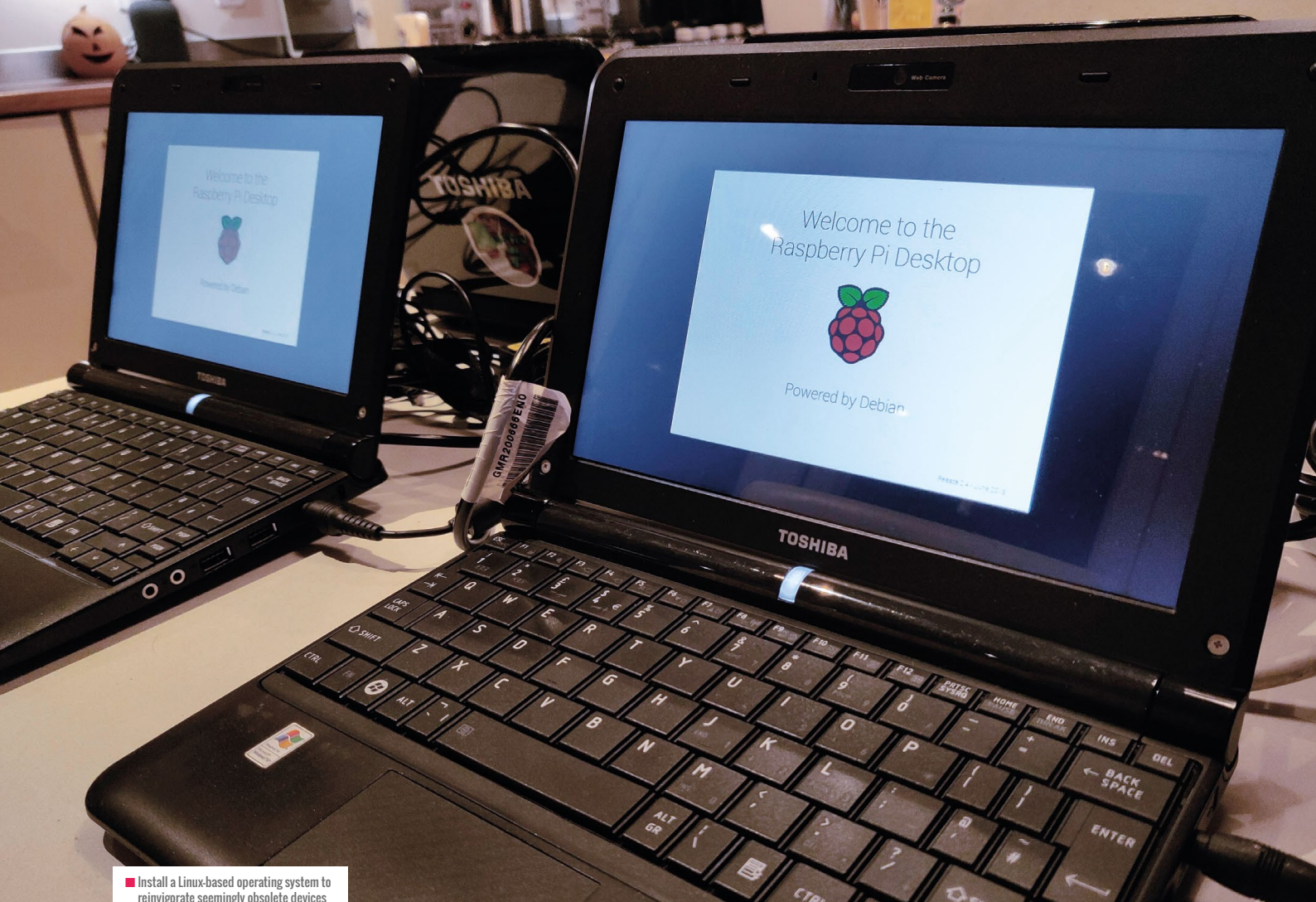
■ Install a Linux-based operating system to reinvigorate seemingly obsolete devices

Upgrade an old laptop

Sometimes, extending the life of an old device won't be as easy as changing the software you use, and you'll need to get a little more hands-on with the hardware! For example, computing teacher Mark Weddell (**@markfromlondon**) recently upgraded a laptop he had originally purchased in 2012. When he bought the PC, it had all the latest specifications — but years later, it was really showing its age. It took a long time to boot up and would often freeze for a few minutes at a time. The laptop was barely usable, and the battery had deteriorated so badly that it needed to be kept plugged in to use it, but the rest of the hardware was still in great, clean condition.

Mark took the following actions to upgrade his laptop:

1. **A fresh install of the operating system:** a registered version of Windows 10 had previously been installed on the laptop, so it was possible to choose a fresh install from the website and remove all previous software.
2. **A new SSD:** Mark replaced the previous magnetic hard drive, which was slow, with a new solid-state drive (SSD). This cost him £50 (about $60) for a 500GB drive. In order to check that the dimensions and specifications of the replacement would work correctly, Mark first removed the old drive from the laptop and compared it with online photos of the replacement.
3. **A new battery:** after removing the battery, Mark was able to note its part number, the power requirements, and the

specification rated in milliampere hour (mAh). He found that original manufacturer parts were expensive and difficult to source, so he purchased an unbranded replacement instead. There are risks associated with the use and disposal of batteries, and readers are advised to follow all guidance.

> " WE CAN ENJOY USING DEVICES THAT OTHERWISE WOULD BE GATHERING DUST

Convert to Chromebook

It is also possible to convert mobile devices with keyboards into Chromebooks (computers that use Google's Chrome browser as the OS). Although Chrome OS isn't available to download, there is an open-source version, managed by Google, called Chromium OS. Installing it is not a simple process, but it can go a long way towards extending the useful life of a laptop. One solution that makes it easier is CloudReady: Home Edition. This is free for individual use; all you need is a USB thumb drive to install it. It's meant for use in education, there are site licence options, and the devices can be managed remotely. If you search online, you'll find plenty of tutorials and videos showing you alternative approaches. ▶

■ You can dismantle devices for a junk modelling craft activity

If none of the ideas mentioned in the article pique your interest, here are a few more suggestions:

■ Preserve old devices for a stimulating classroom wall display or collection

■ Use the devices to build a sandboxed network built by students

■ Repurpose a device as a dedicated media player, as a games system, or to access older stored media

Technology teardown

For devices that are too expensive to repair, a terrific option is to use them as practical hands-on learning resources. The Science Museum has a resource called Wreck Your Tech (**helloworld. cc/wrecktech**), which you can use to help your students learn how devices are assembled, investigate parts of the machines they can't normally access, and make their learning more concrete.

If you plan this activity just right, and supply the right resources, devices can be carefully disassembled and then reassembled afterwards, and used time and time again. While it can be entertaining to tear a device apart using brute force, if it can't be restored to its original condition afterwards, how useful is the experience? Show the students methodical, non-destructive approaches. Screws should be carefully collected, stored, and labelled. Take photographs at key stages of disassembly to aid with reassembly. Your students will be learning about sustainability and developing life skills that will help boost their confidence and technique when it comes to repairing devices at home. It's likely you'll need some special tools such as torx screwdrivers and tweezers.

It's helpful to connect these kinds of activity to real-life uses. Ask your students, for example, to consider how valuable teardown activities might be to commercial organisations. (Products are acquired so that competitors can learn about the components and techniques used, and boost their own research and development.)

When a piece of equipment is no longer suitable for reuse as a device, you can also tear down the components for a junk modelling craft activity in which students construct robots, imaginary creatures, or items of jewellery, for example. This can be a great way to connect computing to other subjects, or to extracurricular clubs. The unusual shapes of connectors and the brightly coloured wires have high visual appeal and can spark creative minds. You'll see some very creative, inspiring ideas on Pinterest under the 'computer repurposed' heading.

Wrecking and junk modelling activities can be incredibly enriching, but they are not without risk. Students need to be briefed on the health and safety risks, which vary depending on the equipment being dismantled. Devices with larger batteries and optical drives that use lasers need to be handled with care. See the 'Important considerations' section for more information about risks.

▶ It's worth noting that Chromium has some limitations. For example, it reduces the option of installing conventional software such as Scratch or Python — but fortunately, web-based alternatives are available for those programs. Neither does Chromium support licensed media codecs such as MP3 and H.264, which means it's challenging to play media. In order to watch videos from streaming sites such as YouTube and Netflix, codecs may need to be installed manually.

■ You can preserve old devices for a stimulating classroom wall display or collection

Donation

You can extend the life of your tech by donating items you no longer need. When schools were forced to close due to the Covid pandemic and lessons took place online at home, there was a rush to provide students with devices to support their learning, but the demand for devices overtook the availability. It's likely that there's a local charity near you that refurbishes computers for worthy causes, and there are national charities that will organise collection.

WeeeCharity, for example, is a not-for-profit UK-registered charity that will collect your old computing equipment from your home free of charge (**weeecharity.com**). They can also securely wipe your data, again free of charge. If they are unable to repurpose a device, they dispose of it in an environmentally responsible way. You can read more about the power of donating devices in James Abela and John Ling's article on page 24.

Important considerations

While repairing, reinvigorating, and donating your tech are all worthwhile activities, there are a few risks that you'll want to consider before diving in:

Data sanitisation: Steps may need to be taken to safely and securely ensure that no data remains on a device. Although advice varies, physically removing any storage media (for example, the hard drive) before donation provides the highest levels of security. Low-level formatting software can also be used to remove data from drives (for example, Windows 10 factory reset options allow you to permanently delete data, or install HDD LLF from HDDGURU; **helloworld.cc/hddllf**).

Transfer of ownership: As policies vary from school to school,

ALAN O'DONOHOE
Alan has over 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 computing teaching, and promoting the appropriate use of technology (**@teknoteacher**).

it's worth consulting your institution's policy and discussing your plans with an appropriate member of staff, such as the business manager. This will help you to understand what is permissible when equipment reaches the end of its life and the necessary steps that must be taken.

Safety: Any learning activities that involve disassembling hardware and the use of tools will require a comprehensive risk assessment beforehand, and ought to include demonstrations of the correct procedures for the use of tools; wearing PPE (personal protective equipment); adequate levels of adult supervision; the identification and removal of hazard threats (for example, lasers in optical drives); the isolation of mains power; and the safe handling of equipment. It's recommended that any mains power leads are removed, or plugs cut off completely.

While the suggestions outlined in this guide can revitalise technology destined for landfill, they also go much further, providing valuable experiences that positively impact young people, as well as the planet. These activities will lead to lively discussions among your students, model the attitudes of responsible consumers who use resources in a sustainable manner, and develop within them the confidence and capacity to repair and refurbish rather than destroy or dispose. Hopefully, this guide will instead encourage you to be the teacher who champions dedtech and thinks twice before ordering and unboxing that shiny new gadget! (HW)

FURTHER RESOURCES

■ A video exploring dedtech in use at Todmorden Makery, a community workshop for makers in the UK: **helloworld.cc/ todmorden**

■ Raspberry Pi Desktop for PC and Mac, a good OS option for old computers: **helloworld.cc/rpidesktop**

■ Pinterest board showing creative uses of old computer parts: **helloworld.cc/pinterestcomputers**

■ WeeeCharity, a UK-based charity which repurposes old devices: **weeecharity.com**

■ Computer Aid International, a charity that matches donated computers to organisations: **computeraid.org**

ME AND MY CLASSROOM

Primary teacher **Hugh Simmonds** outlines the importance of setting up your classroom, both physically and digitally

The phrase "Fail to prepare, prepare to fail" has always struck a chord with me. Being prepared for the upcoming year is so important, and I like to use some of the summer break to organise and prepare my classroom. How a classroom is set up is important; it can help maximise children's learning and help them reach their potential. It also reflects you as a teacher, regardless of the year group you teach. In my opinion, a classroom is organised in a variety of ways, both physically and digitally: what is on the display boards, how the tables are organised, where the children sit, and how I keep track of their progress.

First impressions

One of my core beliefs is that where possible, we should be championing and showing off the work that our children produce. The classroom walls and display boards should be colourful and show the wide range of work we do. This is important because it is the first thing that any person will see when they walk into the classroom, and it sets the tone.

I focus each display board on a specific purpose. For example, we have a display board around the children's coat pegs including work about them, about each individual personality in that classroom. We have permanent display boards focused on maths and literacy, including key terms and interesting examples. They're positioned close to the children so they can easily access them.

I also have a couple of boards for other topics, and these are 'working' walls, so the displays change. At the start of the half term, they have a title and some facts, but as the term progresses, it slowly fills out with children's work. For example, my class's autumn term computing unit was based on spreadsheets, so the board had key terms and formulae added, with examples of the children's work. This board was added to incrementally each week until by the end of the term, it was full and clearly displayed the progress of the unit.

Seating plans

Another aspect of classroom organisation is planning where the children sit. I personally prefer to have children on different tables for maths, English, and other lessons. By creating a map of the class, I can position children and visualise the best position for each one. I can plot who needs to be near me, as the class teacher, and who needs to be near my teaching assistant. I can then place target children in places where they're going to get the support they need. Having a digital version of the table plan is great because at the start of the term, I can display it on the board and challenge the children to work it out. This is a problem-solving challenge for the children, but it can also highlight who the organisers within the class are.

Digital classroom set-up

Finally, the interactive whiteboard. Each morning, I display slides with instructions recapping the morning routine, so it is the first thing the children see as they come in. This can be adapted if there's a specific piece of work the children need to finish. It's visual, it's clear, and the children respond to it. By the end of the first school term, the children appear to not take any notice of it. That's until you don't display it, and then come the curious and puzzled questions: "Where's the board?" or "Why is it not up?" Clearly, the set-up of the classroom is just as important to the kids as it is to me! **(HW)**

■ Our classroom displays change throughout the year, showing off children's work

HUGH SIMMONDS
Hugh is a Year 6 class teacher, with students aged ten to eleven, at Kings Langley Primary School in the UK, where he has also been the computing subject leader for the last five years.

THE BEBRAS PUZZLE PAGE

Each issue, **Chris Roffey** shares a computational thinking problem for your students

THE PROBLEM: PHOTO TOUR

Structures, patterns, and arrangements

In the picture below, a beaver walks around a pond, starting from the position shown and moving in the direction indicated by the arrow. In which order did the beaver take pictures 1–4? *This problem was originally written by the Lithuanian Bebras team. Solution on page 96.*

Further information

Mapping services such as Google Maps often have a street view feature that allows the user to travel along roads on their computer and look around. These views are created by special cameras mounted on vehicles. The cameras take pictures in multiple directions simultaneously which are then sewn together to create a set of linked 360-degree images.

The beaver in this question takes photos around a lake from different directions, which store enough information to rebuild a 3D view of the lake. To do this, it is important not only to have more than one image, but also to know where the photos were taken from, and in what direction the camera was facing.
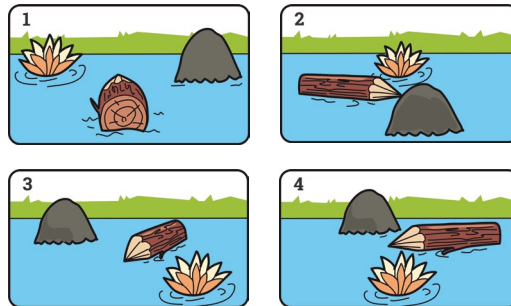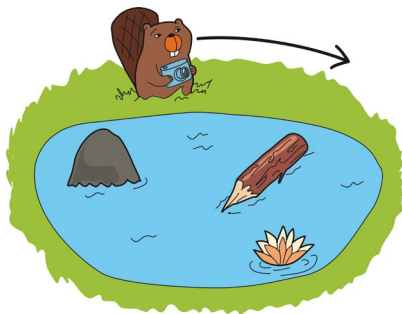
| DOMAIN	
Computer processes and hardware	
SKILLS	
Abstraction, decomposition, and pattern recognition	
AGE	
8-16 years	
DIFFICULTY RATING	
Ages 8-10 hard; ages 10-14 medium; ages 14-16 easy	

ABOUT BEBRAS

Bebras is organised in over 50 countries and aims to get students excited about computing and computational thinking. Over half a million UK students took part in the last two annual Bebras challenges. Our archived questions let you create your own auto-marking quizzes at any time during the year. To find out more and register your school, head to **bebras.uk**.
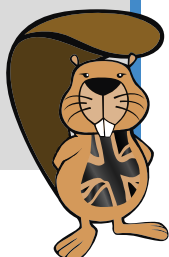
COMPUTING KEYWORD SPOTLIGHT: INFORMATICS

Defining everyday computer science-related words and phrases

Many countries in the Bebras community use the word 'informatics'. In the UK, we normally say 'computer science'. Some UK students compete in the International Olympiad in Informatics. So, what is informatics?

The Oxford English Dictionary (OED) defines informatics as: "The science of processing data for storage and retrieval; information science". The term is, however, used differently in different countries, and indeed is often a synonym for computer science. In most countries, the term is now more akin to the OED definition. Informatics involves methods of storing and retrieving information; it is about data types, and algorithms to store, manipulate, and interrogate information and data. Informatics also concerns itself with computational systems and processes that become the foundations for solving problems programmatically. For example, efficient sorting algorithms have been developed by computer scientists working in the field of informatics. These can be illustrated to pupils using unplugged activities.

Readers may be thinking that informatics sounds like a good description of the areas that Bebras questions focus on. This is not a coincidence!

TEACHING COMPUTING UNPLUGGED IN PRIMARY SCHOOLS

What does teaching primary computing without computers really look like?

Rachael Coultart

The idea that we can teach computing without computers (or 'unplugged') is becoming ever more popular in primary schools today. Perhaps this is due to a lack of reliable technology in our schools with their dwindling budgets, or maybe it is because by removing the distraction of computers, we can focus all our attention on learning new computational thinking skills. This is what the authors of this book, supported by an impressive range of contributors, set out to demonstrate.

It's an easy read, using concise language that gets to the point quickly, with just enough background information, and a potted history to keep you interested. Each chapter follows a similar format, starting with some contextual information followed by three imaginative and engaging lesson ideas. These ideas are well supported with references and resources so you can take them further. I like the flexibility that is built into the lesson ideas, with prompts for how to support or challenge learners. The 'Key Questions' and 'Reflective Questions' are particularly helpful in understanding what the lesson is trying to achieve.

This book is not just about delivering lessons that don't rely on unmanageable technology, or making cross-curricular links with computing. It is more about getting to the heart of what we mean by computational thinking and supporting our young learners to develop the skills they need to succeed in today's world.

Sometimes the activities are for computing lessons, such as understanding Boolean logic. At other times, the authors use a different subject, such as testing waterproof materials in science, where the computational thinking element is highlighted. So not only has this book enriched my understanding of computational thinking, but it has also given me a better understanding of how I can really embed it throughout all the curriculum subjects.

The authors recognise that there is work to be done to provide a bank of children's work that exemplifies different computational thinking concepts. I would suggest that completing the activities in this book would be an excellent starting point, and perhaps even the focus for an interesting research project! I certainly recommend the book as essential reading for anyone wanting to expand their understanding of what computational thinking looks like in the classroom. **(HW)**
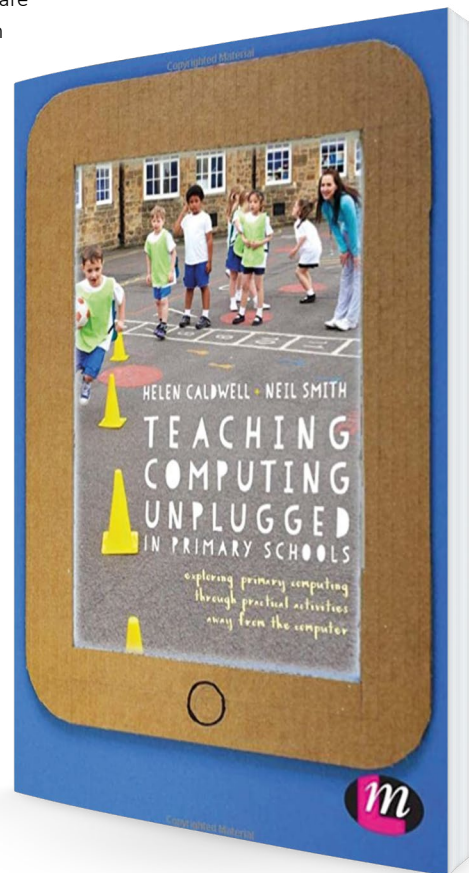
SHE'S BUILDING A ROBOT

A story of self-discovery for children aged eight to twelve

AZ, a talented young girl, loves to code but has always been too nervous to show her passion in front of her peers, until one day she takes the leap. She's tired of the school bully, Dalk, always achieving the glory, and finally gains the confidence to challenge him in a robot-building competition. AZ quickly realises she cannot take on Dalk on her own, so forms a team of other girls who each bring their own skills to develop Ada the robot.

She's Building a Robot travels through the ups and downs of digital making, as well as exploring the challenges and benefits of teamwork. The story is structured as a mission-control type countdown, which adds a level of anticipation that encourages you to read on.

I read this story with my nine-year-old daughter Lottie, who loves problem-solving and was a perfect age for the text. We particularly liked that the book did not require any previous understanding of STEM. The story is peppered with definitions and explanations that enable the reader to really engage with the content.

Lottie also enjoyed the logic problems within the story and made me stop at various points to give her time to try to work out the answers before they were revealed. The teamwork and friendship elements particularly engaged Lottie, and Ada's personality adds some nice humour along the way. As a mother working in STEM who regularly suffers from impostor syndrome, I also really enjoyed the internal monologue from the main character, in which she demonstrates self-doubt and fear, and then eventually courage to come forward to successfully lead the team. This is more than just a story about building a robot. The story encapsulates some of the real-life challenges women face in STEM, and demonstrates positively how these can be managed, providing some excellent role models for young girls. **(HW)**

BEBRAS PUZZLE SOLUTION (PAGE 95)

The photos were taken in the order: 1, 4, 3, 2.

The beaver took the photos from the positions shown in the image to the right.
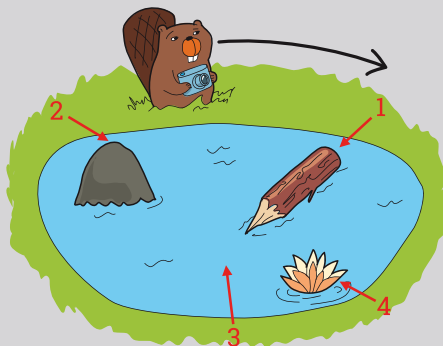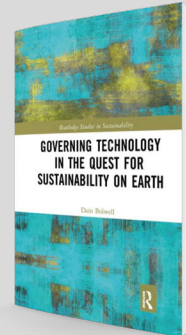
ESSENTIAL READING

Three books that focus on technology and sustainability

### GOVERNING TECHNOLOGY IN THE QUEST FOR SUSTAINABILITY ON EARTH

This academic title explores how we can most effectively govern technology to ensure a long-term and sustainable future, and includes a case study on robotics and artificial intelligence.

THERE IS NO PLANET B

Accessible, informative, and entertaining in equal parts, this book gives readers a big-picture perspective on the world's environmental challenges and includes deep dives into questions such as, 'How can we take control of technology?', 'Should we frack?', and 'What should we do first?'
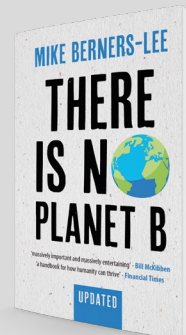
THE UPCYCLE

This powerful manifesto frames questions of resource scarcity and sustainability as questions of design. The authors envisage beneficial designs of products and buildings that will not only sustain life on Earth, but grow it.
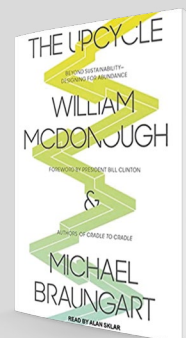
"HELLO, WORLD!"

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 and accompanying podcast 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 and the accompanying podcast are produced by the Raspberry Pi Foundation.

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 three times per year. Check out our podcast at **helloworld. cc/podcast**, too, to get more great Hello World content between issues.

(Hello World)

THE MAGAZINE FOR COMPUTING
& DIGITAL MAKING EDUCATORS

TOMORROW'S ENGINEERS
The first Schools COP

PHILOSOPHY FOR CHILDREN
A fresh approach to digital literacy

INTERVIEW: CENTRE FOR AI & CLIMATE
AI for mitigating climate change

Issue 19   Jun 2022   helloworld.cc

SUSTAINABILITY & COMPUTING

HOW CAN TECHNOLOGY BE GOOD FOR THE PLANET?

PLUS CLOZE QUESTIONS • RESEARCH CENTRE LAUNCH • REPAIR CAFÉS • COOLEST PROJECTS
COSPACES • UPGRADE CULTURE • ONLINE SAFETY • COMPUTING QUALITY FRAMEWORK
BAKING IMPOSSIBLE • BROWSER-BASED CODE EXERCISES • AI & ENVIRONMENTAL RISKS

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.

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 you'd
  like to share? We'll feature your thoughts
  and ideas.

- **Tell us your story**
  Have you had a success (or failure) you think the
  community would benefit from hearing about?

- **Write for the magazine**
  Do you have an interesting article idea? Visit
  helloworld.cc/writeforus to submit your idea.

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

NEXT ISSUE OUT
NOVEMBER 2022

THEME:
COMPUTER SYSTEMS & NETWORKS

(Hello World)

helloworld.cc