(Hello World)

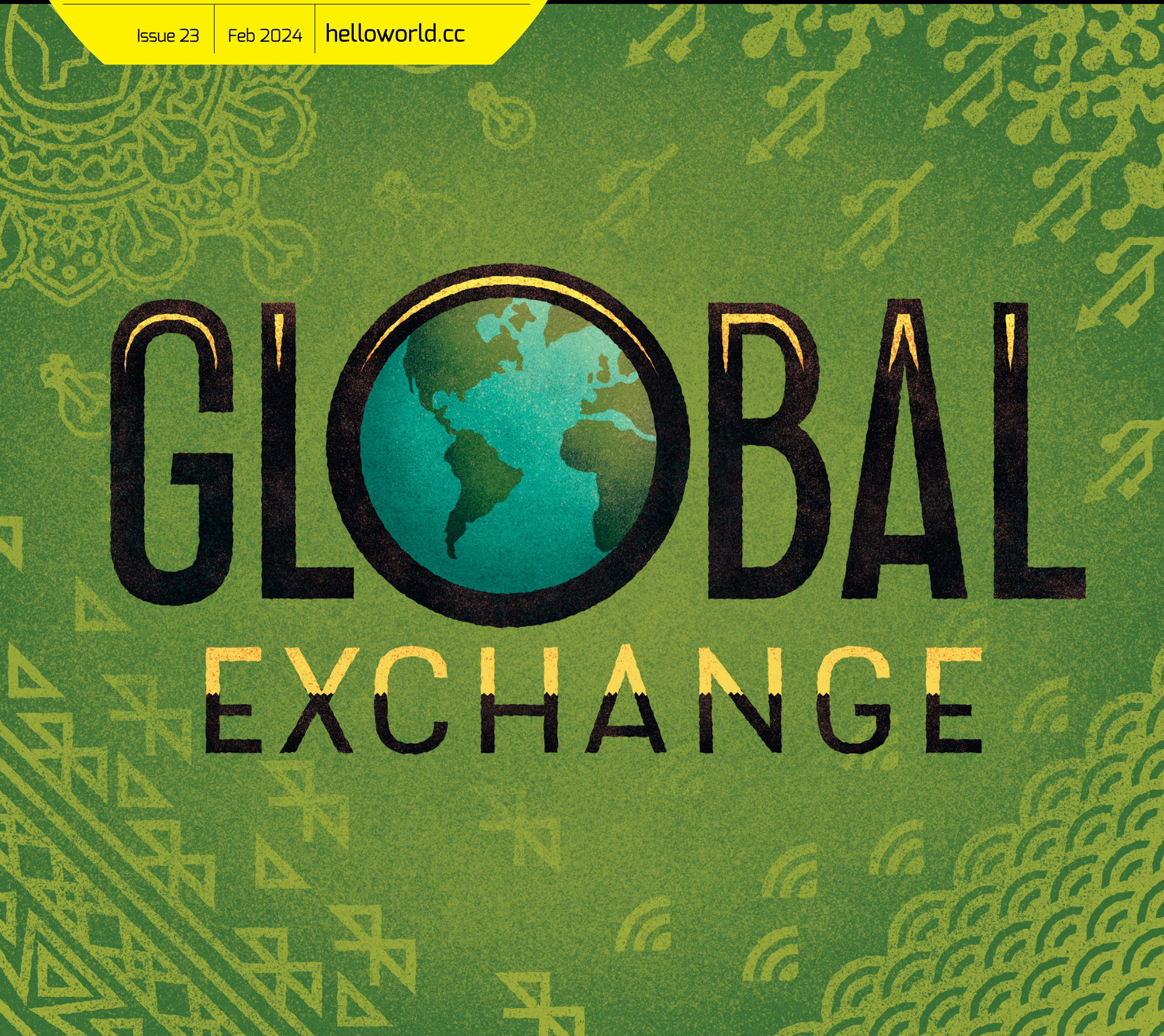
THE MAGAZINE FOR COMPUTING & DIGITAL MAKING EDUCATORS

Issue 23 | Feb 2024 | helloworld.cc

COMPUTER SCIENCE EDUCATION
Exchanging ideas from over twenty countries

WHERE ARE WE NOW?
Ten years of CSed in England

BRIDGING THE DIGITAL DIVIDE
Serving under-resourced communities in India



PLUS **CODE EDITOR** • RESEARCH SEMINARS • **IDEAS FOR BUILDING A CS COMMUNITY** • CAPE FRAMEWORK BEBRAS • **COOLEST PROJECTS** • TRANSLATION AND LOCALISATION • **REPLACEMENTS FOR REPLIT** **EXPERIENCE AI** • BBC MICRO:BIT GAMES • **CULTURALLY RELEVANT PEDAGOGY** • LESSON PLAN

Raspberry Pi Foundation

Teach and learn with the Raspberry Pi Foundation

Free for everyone anywhere in the world

Teachers' resources

Discover training, resources, and guidance to help you teach computing with confidence.

raspberrypi.org/teach

Project library

Browse 200+ online project guides that help young people create with digital technologies.

projects.raspberrypi.org

Research to practice

Explore our research into what works best in teaching school-age young people about computing.

raspberrypi.org/research

Coding at home

Watch our support tutorials and access engaging resources for your child.

raspberrypi.org/learn

Raspberry Pi Foundation, UK registered charity 1129409

HELLO, WORLD!

My six-year-old is learning about money in maths at school. It made me wonder how long we've been teaching and learning maths this way. I read that in Europe in the 1300s, reckoning schools were set up because merchants and accountants wanted to teach their sons about money (**helloworld.cc/MathHistory**). These parents wanted a better future for their sons.

Now we look to computer science as the path to a better future. While making this Hello World issue, I've met incredible educators, researchers, and volunteers who want to create a better future for sons, daughters, and all young people by teaching them about computing — whether they have access to formal computer science education or not.

You'll hear from countries that are far along on their computing education journeys and incorporating artificial intelligence into their curricula, and from countries that are newer to computer science education and looking to fast-track their students' education. Ethel Tshukudu's article on her research using the CAPE framework is a fascinating comparison of computer science education in four low- to middle-income African countries (page 32). Divya Joseph and Bobby Whyte's article describes the different methods our team in India are using to broaden access to computing (page 52). Finally, Dana Rensi's article about a solar-powered Raspberry Pi computing lab in the middle of the Peruvian rainforest will surprise and delight you (page 28).

I hope that reading this issue of Hello World fills you with the sense that you are part of a global community of important educators changing the world.

Meg Wang
Editor

FEATURED THIS ISSUE

ILIANA RAMIREZ

Iliana, human capital director of CSOFTMTY, describes on pages 26–27 how volunteers are at the heart of Ciberistas, a tech training programme for young people in Mexico.

RANDAL ROUSSEAU

Randal is a librarian in Cape Town, South Africa. He describes how he teaches children to code through ingenious, unplugged activities on pages 36–37.

DR NADIA AL-ABOODY

Nadia is a bit of a celebrity at the Raspberry Pi Foundation. On pages 50–51, read why she started her first Code Club in Iraq and how she has gone on to establish eight more clubs.

(HW)

Hello World is the official magazine of the Raspberry Pi Foundation

Raspberry Pi Foundation

EDITORIAL

Editor
Meg Wang

Subeditors
Louise Richmond and Amy Rutter

Subscriptions
Sean Sayers

Social Media
Neena Patel

DESIGN

criticalmedia.co.uk

Head of Design
Lee Allen

Designer
Sam Ribbits

Photography
The Raspberry Pi Foundation, Adobe Stock

Graphics
Rob Jervis

Cover
© MUTI

CONTRIBUTORS

James Abela, Salomey Afua Addo, Nadia Al-Aboody, Darren Bayliss, Katharine Childs, Tonya Coats, Elena Coman, Michael Conterio, Andrew Csizmadia, Penelope DeFreitas, Liz Eaton, Catherine Elliott, Vicente Forcada, Helen Gardner, Tracy Gardner, Sway Grantham, Matthew Griffin, Ben Hall, Graham Hastings, Shanika Hope, Nic Hughes, Divya Joseph, Juho Kahila, Diana Kirby, Chris Lovell, Mamta Manaktala, Rujeko Moyo, Kaye North, Alan O'Donohoe, Iliana Ramirez, Dana Rensi, Chris Roffey, Corey Rogers, Randal Rousseau, Sue Sentance, Ben Schafer, Cindy Smits, Bonnie Sheppard, Ethel Tshukudu, Joanne Vincent, Matthew Wimpenny-Smith, Bobby Whyte

Contributing Partner

bcs Academy of Computing
CAS

This magazine is printed on paper sourced from sustainable forests and the printer operates an environmental management system which has been assessed as conforming to ISO 14001.

Hello World is published by the Raspberry Pi Foundation, 37 Hills Road, Cambridge, CB2 1NT. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to skills, products, or services referred to in the magazine.

(HW) CONTENTS

NEWS, FEATURES, AND OPINION

Credit: @Rixie/stock.adobe.com

LEARNING

RESOURCES & LESSON PLANS

CONVERSATION

REVIEWS

EXPERIENCE AI: EXPANDING CONTENT AND REACH

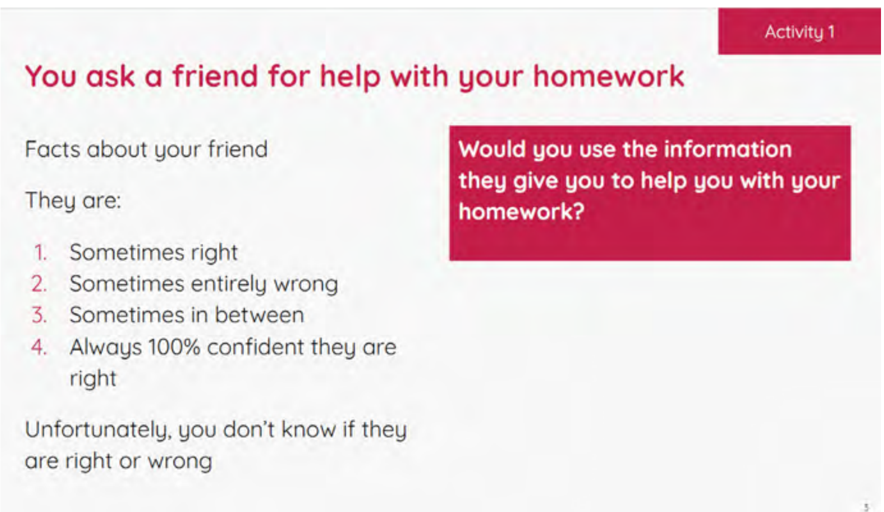
The Raspberry Pi Foundation and Google DeepMind are building a global network of educational organisations to bring AI literacy to teachers and students all over the world

Liz Eaton

We launched Experience AI in September 2022 to help teachers and students learn about AI (artificial intelligence) technologies and how they are changing the world. Developed by the Raspberry Pi Foundation and Google DeepMind, Experience AI provides everything that teachers need to deliver engaging lessons with confidence. The lessons will inspire and educate young people about AI and the role that it could play in their lives.

Comprising six carefully designed lessons for secondary-school students (ages 11–14, Key Stage 3), Experience AI includes lesson plans, classroom resources, worksheets, hands-on activities, and videos that introduce a wide range of AI applications and the underlying technologies that make them work. The materials are designed to be relatable to young people and can be taught by any teacher, whether or not they have a technical background. Alongside the classroom resources, we offer teacher professional development, including an online course that provides an introduction to machine learning and AI.

The materials are grounded in real-world contexts and emphasise the potential for young people to positively change the world through mastery of AI technologies.



The 'large language models – PSHE' lesson explores the trustworthiness of AI

New lessons

As well as the six AI lessons, there are two additional cross-curricular lessons (**helloworld.cc/AILessons**).

- **Large language models (LLMs) — PSHE** This sequence of activities is designed to educate students about the development of large language models (LLMs) and relates to PSHE (personal, social, health, and economic) education. The activities will give students the opportunity to explore the purpose and functionality of LLMs while also examining the critical aspect of trustworthiness in their output.

- **Ecosystems and AI — Biology** This lesson encourages students to explore the impact of environmental changes on the organisms in an ecosystem — in this case, the Serengeti National Park in Tanzania. They will consider the problems of measuring biodiversity in order to maintain it, before learning about AI and the benefits that AI applications are bringing to conservation in the Serengeti. There is scope for a wider look at societal attitudes to AI and the uses of AI in science.

Experience AI network

Since launching the first resources, we have seen significant demand from

Credit: © "Elephants - Serengeti National Park safari - Tanzania, Africa" by David Berkowitz, used under CC BY 2.0

■ We collaborated with the Biology Education Research Group (BERG) at the UK's Royal Society of Biology to make sure the 'Ecosystems' lesson is relevant and accessible for Key Stage 3 teachers and their students

teachers and students all over the world, with over 367,000 students already learning with Experience AI.

Building on that initial success, we are now building a global network of educational organisations to expand the reach and impact of the programme. We're translating and localising the materials, promoting them to schools, and supporting teachers' professional development.

The first four organisations that we are working with are already doing fantastic work to democratise digital skills in their part of the world. Digital Moment in Canada (**digitalmoment.org**), Tech Kidz Africa based in Kenya (**techkidzafrica.co.ke**), Penang Science Cluster in Malaysia (**pscpen.com**), and Asociația Techsoup in Romania (**asociatiatechsoup.ro)** are already working in partnership with the Raspberry Pi Foundation, and we are excited to be deepening and expanding our collaboration to include AI literacy.

Get involved

These are the first of what we hope will become a global network of organisations supporting teachers to equip students with a foundational understanding of AI technologies through Experience AI. If you want to get involved in inspiring the next generation of AI leaders, we would love to hear from you (**helloworld.cc/ExpAIgetinvolved**). (HW)

EXPERIENCE AI CHALLENGE

The Experience AI Challenge invites and supports young people aged up to 18 to design and make their own AI applications. You and the young people you work with will learn how to make a machine learning (ML) classifier that organises data types, such as audio, text, or images, into different groupings that you specify.

This is a chance to have a taste of what it's like to get creative with the powerful technology of machine learning. And equally exciting: every young, UK-based creator will get feedback and encouragement from us at the Raspberry Pi Foundation. Download the Challenge resources now! (**helloworld.cc/AIChallenge**).

Key information

- Free to take part
- Designed for beginners, based on the tools Scratch and Machine Learning for Kids
- Tailored resources for young people and mentors to support you to take part
- Open for official submissions by UK-based young people aged up to 18 and their mentors
- Finishes on 24 May 2024

Note: Young people and their mentors around the world are welcome to access the resources and make AI projects. However, at this time, only UK-based submissions will receive feedback on their project.

RESEARCH SEMINARS 2024

Teach programming with and without AI in our new seminar series

Bonnie Sheppard

How do you best teach programming in school? It's one of the core questions for primary and secondary computing teachers. That's why we're making it the focus of our free 2024 online seminar series. You're invited to attend and hear about the newest research about the teaching and learning of programming, with or without AI tools.

Seminars for all

Our monthly online seminars are not only for computing educators, but also for anyone who is passionate about teaching young people to program computers. The seminar participants are a diverse community of teachers, technology enthusiasts, industry professionals, coding club volunteers, and researchers.

With the seminars, we aim to bridge the gap between the newest research and practical teaching. Whether you are an educator in a traditional classroom setting, or a mentor guiding learners in a CoderDojo or Code Club, you will gain insights from leading researchers about how school-age learners engage with programming.

What to expect

Each seminar begins with an expert presenter delivering their latest research findings in an accessible way. We then move into small groups to encourage discussion and idea exchange. Finally, we come back together for a Q&A session with the presenter.

Here's what attendees had to say about our previous seminars:

"As a first-time attendee of your seminars, I was impressed by the welcoming atmosphere."

"Several seminars (including this one) provided valuable insights into different approaches to teaching computing and technology."

> WE CONNECT THE LATEST RESEARCH WITH PRACTICAL TEACHING

"I enjoyed the fact that there were people from different countries, and we had a chance to see what happens elsewhere and how that may be similar and different to what we do here."

Upcoming seminars

- **12 March 2024: the impact of AI tools on the student experience in programming courses: a preliminary study with an intersectional analysis approach**

AI tools such as ChatGPT have significant potential benefits for students: they allow them to work constructively on their own schedule, and provide an easy way to ask for help. However, before these tools can be meaningfully introduced into a course, educators must assess their impact on a student's ability to learn.

Yash Tadimalla and Prof. Mary Lou Maher have performed a preliminary study to understand university students' experiences of learning to program using AI tools.

- **16 April 2024: generative AI in programming education: bridging the gap from school to what lies ahead**

University students who study computing are very likely to continue on a path in computing, but most school students choose to go into non-computing disciplines for further/higher education and their careers.

Dr Brett A. Becker's talk will focus on how educators can effectively utilise generative AI in secondary-school programming education and how it can be leveraged so that students can be best prepared for continuing their education or beginning their careers, regardless of discipline.

The seminars take place at 5–6.30pm GMT / 6–7.30pm CET / 12 noon–1.30pm ET / 9–10.30am PT.

How to sign up

To ensure you don't miss out, sign up now to receive calendar invitations and access links for each seminar on the day; you can also catch up on our previous seminars (**helloworld.cc/RPFseminars**). (HW)

A NEW TEXT-BASED PROGRAMMING TOOL FOR YOUNG PEOPLE

Find out why the Raspberry Pi Foundation created a code editor specifically for young learners

Joanne Vincent

When we started developing our Code Editor (helloworld.cc/CodeEditor), we set out to create a free online tool to help make learning text-based programming simple and accessible for children. We wanted it to be designed specifically for learning rather than for professional use. Created with input from educators, the tool would be informed by our understanding of pedagogy and computing education.

We launched the editor through the Astro Pi programme in September 2022, and it is now available in beta as a standalone web-based code editor.

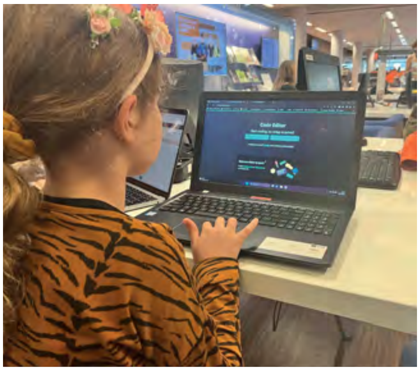
Design considerations

There were many factors to consider when creating a tool for young learners. We wanted young people to use the editor to write and run code in a web browser without the need to install any additional software or create an account (although if you're logged in to a Raspberry Pi Foundation account, your coding project will be saved automatically).

We were also concerned about safety, so we decided that new features would only be made available once we had ensured that they complied with our safeguarding policies and with the age-appropriate design code of the Information Commissioner's Office (ICO). The ICO is the UK's independent body set up to uphold information rights.

We also thought about accessibility, and continue to optimise the editor for mobile and tablet use. Whether young people code in Python or HTML/CSS/JavaScript, we wanted to enable them to easily switch settings between light and dark modes, and between small, medium, and large text sizes.

Finally, we made the editor available as an open-source project so that other organisations could benefit from the work.



Our free online tool helps make learning text-based programming simple and accessible for kids aged nine and up

Using the editor

Having our own editor allows us to offer a reliable, free-forever development environment for learners, utilising the libraries and features we think will benefit them the most.

You can use the Code Editor as a standalone tool, or with the support of our Python and Web design paths on our Projects site (**helloworld.cc/projectpaths**).

We are also adapting selected projects to embed the Code Editor, providing an all-in-one view for learners — no more switching tabs or windows! You can try out the beta version in the Hello World (**helloworld.cc/helloworld**), Target Practice **helloworld.cc/targetpractice),** and Anime Expressions (**helloworld.cc/animeexpressions**) projects.

THE CODE EDITOR

"The Editor looks really nice! I have tried the Python part, and it is intuitive and concise. My little program worked no problem, and I am sure the Editor will be easy, intuitive, and quick to learn for the young [learners]."

- Volunteer in the CoderDojo community

Research

At the end of last year, we carried out a survey to help us understand the tools and features educators use for text-based programming in the classroom. Common themes included an easy-to-use editor with student-friendly error messages, and learning management features. Such insight helps to steer future editor developments, and we also plan to explore offline capabilities.

If you would like to try out the Code Editor, we'd love to hear your feedback: **editor.raspberrypi.org.** (HW)

Code Editor developments have been made possible with generous support from Endless and the Cisco Foundation.

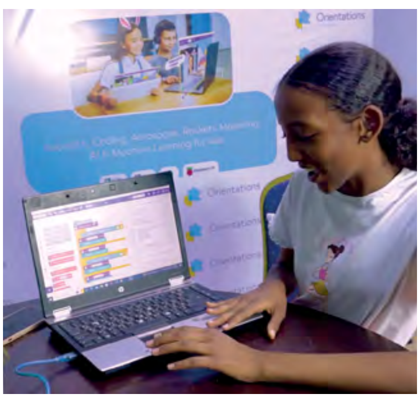
GET CREATIVE WITH COOLEST PROJECTS

Young creators all over the world can celebrate the makes they're proudest of at Coolest Projects

Helen Gardner

From Australia to Zimbabwe, young people around the world are using their incredible coding and digital making skills to solve problems in their communities, express their unique identities, and make projects just for fun.

Coolest Projects is a worldwide showcase of young people's digital making, and it's back in 2024. The Coolest Projects online showcase is open to any young person aged up to 18 from anywhere in the world, and is an opportunity for creators to share projects they are proud of or get inspired to make something new. All participants in the showcase get personalised feedback on their projects, certificates, and fun digital swag to celebrate their achievements, as well as the opportunity to share what they've made with the world.



Afaf's video about her Smart Agricultural System

Solving community problems

Young creators across the globe are using digital making skills to solve problems that matter to them and their communities, and participants in the 2023 Coolest Projects online showcase created some ingenious digital solutions. Eima (aged 12), from a CoderDojo in Japan, created a Scratch game called Clean the Sea (**helloworld.cc/CleanTheSea**). The game aims to educate and inspire people in a fun way to reduce their plastic waste. Eima says, "I visited an exhibit at an aquarium that showed how plastic waste in the sea is causing suffering for marine life. The reason I created this game is to let people know that there is a lot of plastic waste in the sea. Through this game, I would like to contribute to the reduction of plastic waste."

Afaf (aged 13), from a Code Club in Sudan, wanted to help her local community. She was inspired to make her Smart Agricultural System project (**helloworld.cc/SmartAgriculturalSystem**) because "agriculture is the backbone of my country". Afaf's hardware project uses smart technology — including an Arduino Uno, Bluetooth, and temperature and humidity sensors — to track, monitor, automate, and analyse farming operations. By monitoring growing conditions, Afaf hopes farmers will be able to use scarce resources more efficiently, and support better food security. She has even thought of next steps to improve her project further, adding more sensors and connecting the project to the Internet of Things.

Expressing identity

Digital making can also help young creators express themselves and explore their hobbies and interests in new ways, which was the case for Keren (aged 16) from the US. Inspired by her passion for dance, she made an app, Smart Dance Generator (**helloworld.cc/SmartDanceGenerator**). The app uses Python and Dart to choreograph dance moves. Keren noticed that her peers and coaches can struggle with dancer's block, when they find it hard to create original and creative choreography. She says, "This app works as a way for dancers and choreographers to gain inspiration and be able to focus on enjoying their craft. I hope that my app can be used to successfully eliminate stress from a pressure-relieving activity and offer motivation for dancers struggling with crafting authentic choreography."

Digital projects can be successful in conveying a message that's important to the creator without being complex. Chandhan (aged 13) from India submitted his first ever Scratch project to the 2023 Coolest Projects online showcase. He created an animation welcoming people to his Code Club, Pratham (**helloworld.cc/PrathamAnimation**). And Sneha (aged 13),

also from India, used Scratch to animate her friend's name, using animated sprites and sound recordings (**helloworld.cc/SonuAnimation**).

Muhammad (aged 14) in Indonesia also used code to express himself and his identity when he created a website about Sundanese culture (**helloworld.cc/SundaneseCulture**). He says, "I chose to make a Sundanese cultural project because I wanted to introduce Indonesian culture, especially the West Java region, in a creative and modern way so that its sustainability was maintained." Muhammad designed the website using PHP and JavaScript along with CSS and Bootstrap. His next steps are to test the website with different devices and browsers to ensure it's accessible to all, so that more people can understand his culture.

Get involved!

The Coolest Projects online showcase is open to any young creator from anywhere in the world. Registration for this year's showcase is open from 14 February to 22 May 2024. Coolest Projects is completely free to enter and projects are welcome at all levels, from beginner to advanced and everything in between. We love to see ideas and works in progress too, so projects don't need to be complete to participate.

If the young people you work with have made a project they're proud of, enter it into the online showcase so they can share their make with the world and get feedback, recognition, and rewards. We'll be celebrating all 2024 participants in a special livestream on 26 June 2024, so save the date!

If you want to work with young people on a new project, check out more incredible projects from all over the world in the 2023 showcase gallery for inspiration (**coolestprojects.org**), and have a look at our guidance for mentors to help you support young people to enter. The Raspberry Pi Foundation's projects site is another brilliant place to start. The introduction to Scratch path (**helloworld.cc/scratchintro**) is perfect for beginner learners, and the new introduction to micro:bit path (**helloworld.cc/microbit_intro**) is a great first step in learning physical computing. For more

Eima's Coolest Projects submission, Clean the Sea

advanced coders, the more Python path (**helloworld.cc/morePython**) has fun projects, including a cool codebreaker.

When you're ready to register your creations, head to (**helloworld.cc/CPtakepart)** to upload your projects.

There are also in-person Coolest Projects events in some countries, including Ireland, the UK, Belgium, and others. If you'd like to get hands-on and celebrate makes near you with your local community, you can sign up to the Coolest Projects newsletter (**helloworld.cc/CPnewsletter**) to be the first to hear about any in-person events happening in your country. (HW)

COLUMN

SUPPORTING LEARNERS WITH ENGLISH AS AN ADDITIONAL LANGUAGE IN COMPUTING

How can you best support learners with English as an additional language to engage and achieve in computing lessons?

Around one in five pupils in English schools has English as an additional language (EAL), and this can present difficulties in terms of fully accessing the curriculum. However, Louisa Sampson, deputy head at Abbeyfield Primary Academy in Sheffield, where 62 percent of children have EAL, says, "The children love their computing lessons as it's an area of the curriculum where they can excel. They can tinker and make mistakes and know that this is just part of computing — so there's less fear of failure. It's hands-on, and success doesn't rely on written pieces of work in English."

Furthermore, according to data from 2017, students with EAL are overrepresented in the GCSE computer science cohort (**helloworld.cc/Roehampton2017**).

Building on this engagement, how can we ensure that all learners with EAL are fully included and able to achieve their potential in computing? I shall outline a few key approaches in this article.

Translation tools

Although not specifically for computing, it's worth knowing about the translation tools and other assistive technologies that are available to help you and your learners to communicate.

- Google Translate and Microsoft Translator (web-based or app) have a huge number of languages and are improving all the time. In the apps, you can take a photo of some text and they will translate it for you. Not every child can read well in their home language, so you can also hold a conversation using the app — each person speaks in their home language and the app will translate it and play it aloud. You can also translate a document or selected text directly in Google Docs or Microsoft Word.

- For learners who can understand more spoken language than written, the Seeing AI app (**helloworld.cc/SeeingAI**), designed for blind users, is excellent. You can use the camera to scan some printed text, for example on a worksheet, and it will read it aloud in English.

Catherine Elliott leads the Sheffield eLearning Service (**sheffieldclc.net**) and has spent a number of years working on ways to make computing accessible for all learners. She is co-chair of the CAS Include working group, and leader of the Sheffield and South Yorkshire Secondary and Virtual SEND CAS Communities (**@catherinelliott**).

- Immersive Reader (**helloworld.cc/MicrosoftReader**) is a fantastic tool that's built into Word online, the Edge browser, and as an extension in Chrome. It will read the text aloud and translate it, and there is also a picture dictionary function in some versions.

- Finally, the recent developments in generative AI are fantastic for translation. It's very quick and easy to translate a text or PDF, summarise complex texts to make them more understandable for learners, and simplify a piece of writing for any reading age.

Key vocabulary and instructions

It is important to identify the key vocabulary for each topic you are teaching, and ideally, don't include too many words, to avoid overload. Create bulletproof definitions for each one — a one-sentence summary of a key concept or idea that doesn't require any specific knowledge to understand it. Preteach these terms to students before using them in lessons, and review the language regularly.

You — or students — could also create an image-supported bilingual glossary, using the Computing at School inclusive glossaries as a starting point (**helloworld.cc/InclusiveGlossary**). Consider teaching a gesture to represent specific concepts, such as 'repetition', to help with comprehension (see my column on page 14 of Hello World issue 21, **helloworld.cc/21**, for more on this).

Create some code cards for learners to use when planning out programs — for example, have the Scratch code block or Python code on one side, and an image or translation on the reverse, to support understanding.

We need to support students in understanding key terms in English, but do they also have the opportunity to plan out programs, including writing pseudocode, in their home language? Learners preparing for formal exams will need to practise writing pseudocode in English, but students can create more precise algorithms and plan out more complex programs if they can describe fully what they want them to do in their home language.

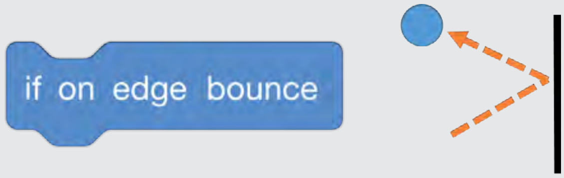
> ONE IN FIVE PUPILS HAS ENGLISH AS AN ADDITIONAL LANGUAGE

Similarly, primary pupils can create programs in Scratch in their home language — simply click on 'Settings' in the top menu and choose from a long list of languages. This changes the language on the interface, but also on each block. Once it has been coded, they can change the language back and start to learn the English instructions.

Finally, how do you ensure that students understand the task that has been set? Providing a task board, or a checklist of each step, can help. Ask ChatGPT or similar to generate a list in very simple language and translate it into another language. Add image support for poorer readers, and check comprehension before you set them off on a project.

Computing pedagogies

High-quality teaching benefits all learners, and students with EAL are no exception. Have you been using the



An example code card with the Scratch code block and a corresponding image of its function

PRIMM (Predict–Run–Investigate–Modify–Make) framework (**primmportal.com**) to scaffold learning in programming? Reading and investigating code helps with comprehension and enables students to practise key terms. Learners could discuss how programs work with a partner who speaks the same language (this also works well with pair programming). Allowing students to modify working code or complete a Parson's Problem builds confidence and provides a model for their own work.

As teachers, we should be modelling use of software and key language as part of our everyday practice. Can you create short video guides to illustrate how to use key applications or devices? Students can then play and pause these as many times as they need to to understand the steps.

Culturally relevant pedagogy

Culturally relevant pedagogy is a framework for teaching that emphasises the importance of incorporating and valuing all learners' knowledge, ways of learning, and heritage. Through culturally responsive teaching, learning is made more relevant to all students and can foster a greater sense of belonging in the subject. Computing projects provide a great opportunity for students with EAL to celebrate their home language and culture. Here are a few ideas:

- Create a multilingual dictionary of computing terms in Book Creator (**bookcreator.com**)with audio and images
- Design a website or create a presentation about a festival or country
- Retell traditional stories using stop-motion animation or green-screen film-making
- Explore the Translate and Text to Speech extensions in Scratch to create a bespoke translation tool or design a translation tool in Python; see an example in Scratch here: **helloworld.cc/ScratchFrench**

I hope that these strategies can help you create a more inclusive classroom that not only supports learners to fully participate in computing lessons, but also helps to celebrate the different languages and contributions of your students with English as an additional language. (HW)

#INSIGHTS

CULTURALLY ADAPTED RESOURCES FOR PRIMARY COMPUTING

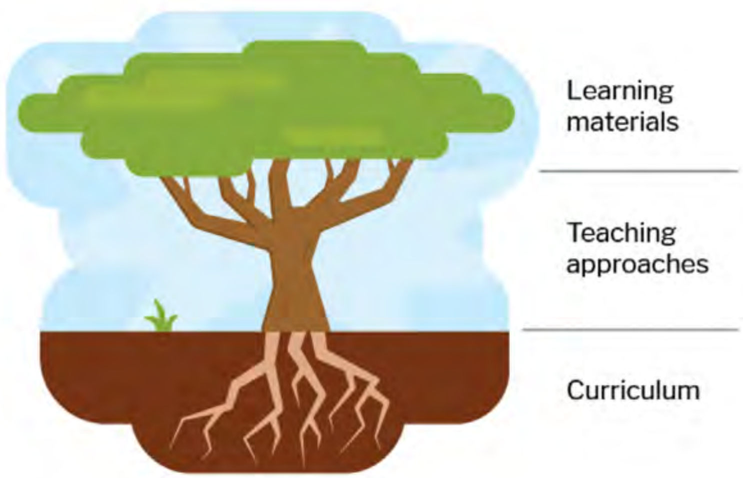
STORY BY Katharine Childs

Who chooses to study computing? In England, data from GCSE and A-level computer science entries in 2019 shows that the answer is complex (helloworld.cc/STEMcharts). Black Caribbean students were one of the most underrepresented groups in the subject, while pupils from other ethnic backgrounds, such as white British, Chinese, and Asian Indian, were well represented. This picture is reflected in the STEM workforce in England, where Black people are also underrepresented (helloworld.cc/STEMwork).

Culturally relevant and responsive teaching

That's why one of our areas of academic research aims to support computing teachers in using culturally relevant pedagogy to design and deliver equitable learning experiences that enable all learners to enjoy and succeed in computing and computer science at school. Our previous research projects within this area have involved developing guidelines for culturally relevant and responsive teaching (**helloworld.cc/17**, pages 54–57), and exploring how a small group of primary and secondary computing teachers used these guidelines in their teaching (**helloworld.cc/21**, pages 64–66).

In our latest research study, funded by Cognizant, we worked with 13 primary-school teachers in England on adapting computing lessons to incorporate culturally relevant and responsive principles and practices. Here's an insight into the workshop we ran with them, and what the teachers — and we ourselves — have taken away from it.



Cultural relevance is important in learning materials, teaching approaches, and the curriculum as a whole

Adapting lesson materials based on culturally relevant pedagogy

In the group of 13 England-based primary-school computing teachers we worked with for this study:

- One third were specialist primary computing teachers, and the other two thirds were class teachers who taught a range of subjects
- Some acted as computing subject lead or coordinator at their school
- Most had taught computing for between three and five years
- The majority worked in urban areas of England, at schools with culturally diverse catchment areas

In autumn 2022, we held a one-day workshop with the teachers to introduce culturally relevant pedagogy and explore how to adapt two six-week units of computing resources.

The first part of the workshop was a collaborative, discussion-based professional development session exploring what culturally relevant pedagogy is (**helloworld.cc/CRPquickread**). This type of pedagogy uses equitable teaching practices to:

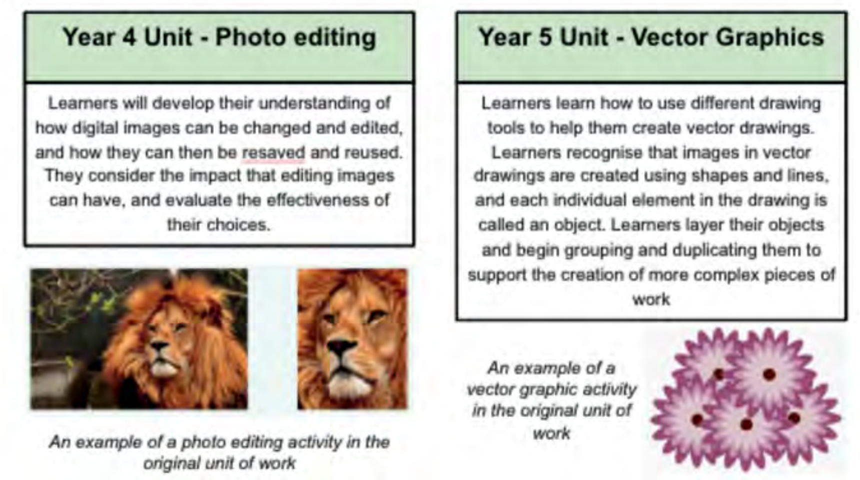
- Draw on the breadth of learners' experiences and cultural knowledge
- Facilitate projects that have personal meaning for learners
- Develop learners' critical consciousness

The rest of the workshop day was spent putting this learning into practice while planning how to adapt two units of computing lessons to make them culturally relevant for the teachers' particular settings. We used a design-based approach for this part of the workshop, meaning researchers and teachers worked collaboratively as equal stakeholders to decide on plans for how to alter the units.

We worked in four groups, each with three or four teachers and one or two researchers, focusing on one of two units of work from The Computing Curriculum for teaching digital skills: a unit on photo editing for Year 4 (ages 8–9), and a unit about vector graphics for Year 5 (ages 9–10) (**helloworld.cc/curriculum**).

In order to plan how the resources in these units of work could be made culturally relevant for the participating teachers' contexts, the groups used a checklist of ten areas of opportunity (**helloworld.cc/AO**). This checklist is a result of one of our previous research projects on culturally relevant pedagogy. Each group used the list to identify a variety of ways in which the units' learning objectives, activities, learning materials, and slides could be adapted. Teachers noted down their ideas and then discussed them with their group to jointly agree a plan for adapting the unit.

By the end of the day, the groups had designed four creative plans for:



We based the workshop on two Computing Curriculum units covering digital literacy skills

> THE TEACHERS REPORTED AN INCREASE IN THEIR CONFIDENCE IN ADAPTING RESOURCES TO BE CULTURALLY RELEVANT

- A Year 4 unit on photo editing that included creating an animal to represent cultural identity
- A Year 4 unit on photo editing that included creating a collage about yourself
- A Year 5 unit on vector graphics that guided learners to create their own metaverse and then add it to the class multiverse
- A Year 5 unit on vector graphics that contextualised digital skills by using them in online activities and video games

Outcomes from the workshop

Before and after the workshop, we asked the teachers to fill in a survey about themselves, their experiences of creating computing resources, and their views about culturally relevant resources. We then compared the two sets of data to see whether anything had changed over the course of the workshop.

After teachers had attended the workshop, they reported a statistically significant increase in their confidence levels in adapting resources to be culturally relevant for both themselves and others. They valued the opportunity to discuss their contexts and how to adapt materials they currently used with other teachers, because it made it a more 'authentic' and practical professional development experience.

Read our research paper to learn more (**helloworld.cc/Childs2023**) or explore the workshop materials (**helloworld.cc/CRworkshop**) to see if you can find an opportunity for integrating culturally relevant pedagogy in your classroom.

The workshop was a useful first step in understanding the value of professional development in culturally relevant pedagogy for computing teachers in England. It demonstrates that working collaboratively to co-design and adapt primary computing resources increases teachers' confidence to adapt resources for themselves and others so that they are culturally relevant. Developing a more culturally responsive and relevant approach will be important in allowing a more diverse range of pupils to access, and excel in, computing. (HW)

UK AND IRELAND: COMPARING CSED POLICY AND PROVISION

To what extent do differences in CSed policy impact teachers' experiences?

STORY BY Diana Kirby

At the Raspberry Pi Computing Education Research Centre, we've been looking specifically at computing in the UK and Ireland. Working with a group of researchers from across the different countries, we've been reviewing the development of computing as a school subject in England, Scotland, Wales, Northern Ireland, and the Republic of Ireland, and using survey data to explore the experiences of computing teachers (helloworld.cc/UKICTS).

Computing policies and curricula in the UK and Ireland

The five countries that make up the UK and Ireland have distinct and separate education systems, each setting its own educational policy priorities and offering different national curricula, qualifications, and teacher education opportunities.

As in many other parts of the world, there have been significant developments in computing policies and curricula over the past decade, requiring computing teachers to develop new subject knowledge and teaching strategies.

In England, major changes to the national curriculum for computing were implemented in 2014, replacing the subject of information and communication technology (ICT) with computing. Computing is mandatory from age 5 to 16, and is a broad subject covering digital literacy and information technology as well as elements of computer science (CS). From age 14, learners can choose to take qualifications such as GCSE and A-level CS. In 2018, the Department for Education funded the establishment of the National Centre for Computing Education, which provides teacher training and student resources. This represents one of the most substantial moves in the world towards educating all children in computing.

In Scotland, students are entitled to a Broad General Education (BGE) from age 3 to 15, and study computing science as part of the Technologies subject area. The computing science curriculum was updated in 2016. From age 15, young people can choose to take National and Higher qualifications in computing science. In 2022, the Scottish government invested £1.3 million to transform computing science in schools and establish the Scottish Teachers Advancing Computing Science organisation, which works with schools to spread best practice.

> TEACHERS TOOK PART IN A SURVEY DESIGNED TO EXPLORE THEIR EXPERIENCES OF TEACHING COMPUTING

In Wales, the new Curriculum for Wales was introduced for primary schools in 2022 and is currently being rolled out on a year-by-year basis for secondary schools. The new curriculum introduces significant changes to the subject of CS, which is part of a new interdisciplinary area of learning and experience. It also reinforces the importance of digital competence, which is a statutory cross-curricular skill alongside literacy and numeracy for all learners aged 3 to 16. Older learners can choose to take qualifications in CS or digital technology. National initiatives such as Technocamps aim to improve teacher confidence and capability to deliver the new CS curriculum.

In Northern Ireland, digital skills are taught to all students from age 3 to 14 as part of Using ICT, one of three statutory cross-curricular skills — alongside mathematics and communication — which must be included in lessons. After age 14, students can choose to take qualifications in computing- or technology-related subjects, such as GCSEs in computing or digital technology.

In the Republic of Ireland, the introduction of a digital technology component as part of the new STEM subject area at primary level is currently in progress, and elective courses have been rolled out at secondary level. Students aged 16 to 18 can choose to take the Computer Science for Leaving Certificate qualification, and those aged 12 to 15 can study an optional short course in coding. A series of continuing professional development (CPD) programmes have been developed through teacher support service Oide to support teachers in delivering the new CS course.

Computing teachers' experiences

We were interested in exploring not only the policy context and the intended curricula in each country, but also computing teachers' own experiences of their teaching.

In 2022, we used a localised and adapted version of the MEasuring TeacheR Enacted Computing Curriculum (METRECC) tool (**helloworld.cc/METRECC**) to survey teachers in the UK and Ireland. METRECC is a validated tool developed by an international working group to create a survey instrument that could be used to monitor CS education in schools around the world. So far, teachers from 14 different countries have taken part in a version of the survey.

The UK and Ireland version of the survey included ten sections, asking questions about many different aspects of computing teachers' experiences, such as qualifications, support and resourcing, classroom practice, teaching approaches, and professional development. Our final data set comprised responses from 512 teachers.

More secondary-school teachers took part in the survey than primary teachers, and on the whole the teachers taking part were experienced, both in general and in their teaching of computing: more than 65 percent had over ten years' experience in teaching (in any subject), and over 30 percent had taught computing for over ten years.

Computing qualifications

We asked the teachers what computing qualifications they held. The most common was a bachelor's degree in computing (held by 37 percent of the teachers), followed by computing professional development (PD) (29 percent). We could see the impact of national policies on teachers' experiences: England and Ireland have both established substantial PD programmes in recent years, and more English and Irish teachers have undertaken PD in computing.

Feeling qualified to teach CS

We compared teachers' responses to the statement 'I feel qualified to teach CS' with the level of qualifications they held, and found that those with computing PD, a post-16 CS qualification, or a diploma or certificate in CS responded similarly to those with no CS qualifications. Those with a bachelor's degree or higher appeared to feel more qualified. The teachers from Scotland, where there is a long-standing requirement for computing teachers to have a degree with a substantial CS component, were most confident about being qualified to teach the subject. Those from Ireland were the least confident, but new requirements have been introduced in Ireland for computing teachers to have studied CS as part of their degree, so perhaps we can expect this to change over the coming years.

Computing self-esteem

The other measure we used to examine teachers' feelings about teaching computing was a validated construct measuring their computing self-esteem. Again, we found that the teachers from Scotland reported the highest CS self-esteem. Teachers in England, where there's been the most substantial policy actions with regards to computing, also reported positive self-esteem.

Classroom teaching time

We asked the teachers how much of their classroom teaching time each week is spent teaching computing. This doesn't follow on directly from national policy — a mandatory computing curriculum doesn't ensure that students have access to a specific number of hours of computing each week, or that specialised computing teachers teach them. In our survey, 93 percent of the secondary teachers in Scotland told us they spent at least half of their teaching time on computing, compared to only 29 percent in Ireland. We were surprised to find that a third of the primary teachers in England spent at least half their time teaching computing, suggesting that primary schools in England may be recruiting computing specialists who teach computing across multiple years.

Computing topics

We also asked teachers which computing topics they taught, and we found that in general, the responses aligned with the intended curricula in the different countries. Nearly all teachers said they taught programming, which was included in all curricula at the time (either implicitly or explicitly) except the Irish primary curriculum. As expected, we found differences between the topics taught by primary and secondary teachers across all countries, such as more primary teachers covering robotics, and more secondary teachers teaching topics such as artificial intelligence and machine learning, ethics, and hardware.

Next steps

If you're interested in exploring our teacher survey data further, the full data set is available open-access (**helloworld.cc/UKICTSresults**). You can also read a previous paper we published on this project (**helloworld.cc/UKICTSstudy**).

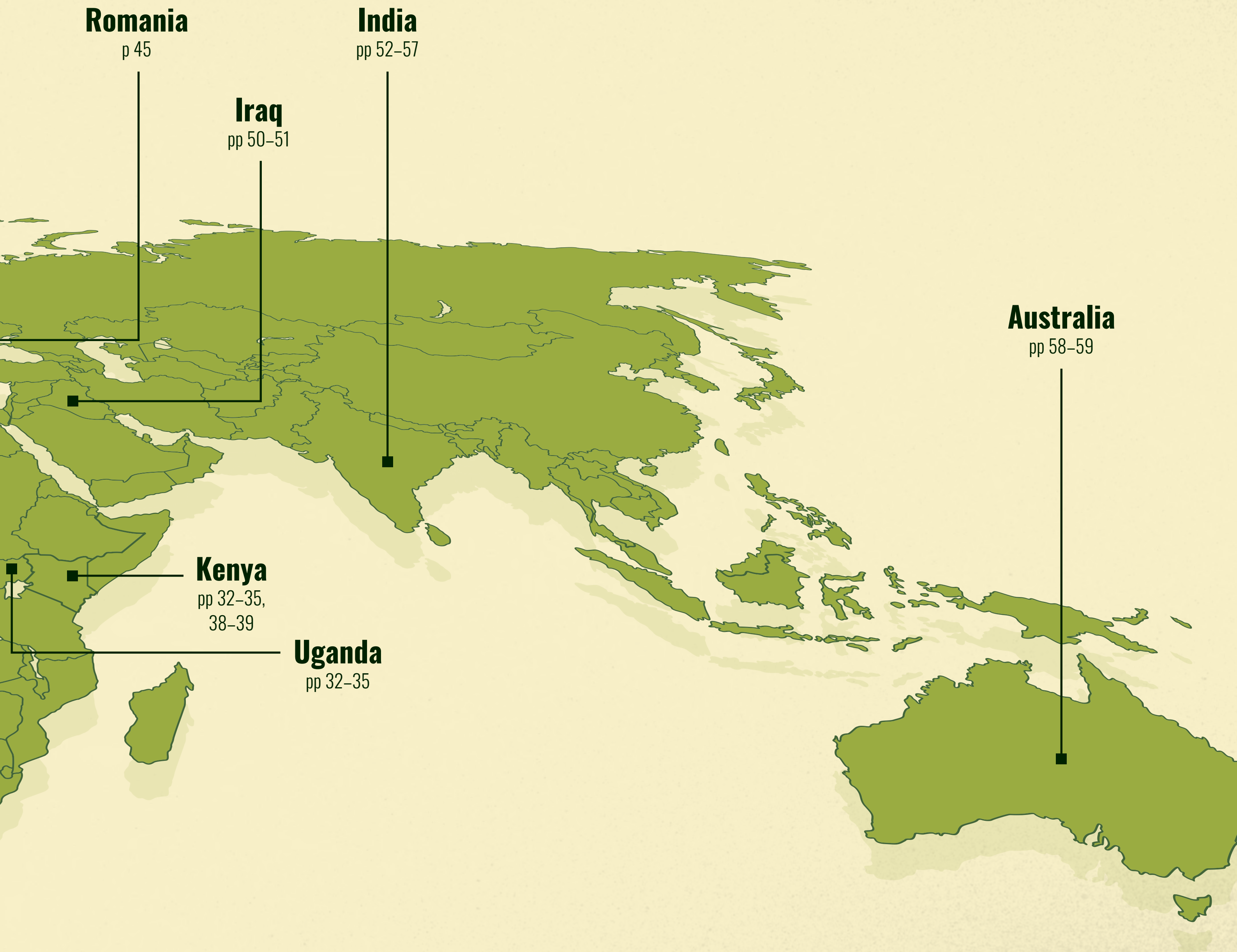
We aim to run a similar teacher survey globally in 2024, to capture a snapshot of how computing is currently being taught and compare the experiences of computing teachers around the world. Look out for more details soon! (HW)

DIANA KIRBY

Diana works in the research team at the Raspberry Pi Foundation. Her work has covered a range of topics, such as AI education, broadening participation in computer science, and computing education in different countries.

FEATURE

GLOBAL EXCHANGE



Romania
p 45
India
pp 52–57
Iraq
pp 50–51
Australia
pp 58–59
Kenya
pp 32–35,
38–39
Uganda
pp 32–35

COMPUTING EDUCATION AROUND THE WORLD

In this article we look at the state of computing education globally and the policy decisions being made by different countries

In the last decade, there has been growing interest at a policy level in the benefits of providing computing as a school subject. Many of the arguments centre around global equity in an increasingly technological society. A 2021 report from Brookings entitled 'Building skills for life' (helloworld.cc/BrookingsCSed) highlighted that educating citizens about computer science can strengthen a country's economic situation and tackle inequality between countries. The report states on page 12 that "global development gaps will only be expected to widen if low-income countries' investments in these [computing-related subjects] falter while high-income countries continue to move ahead."

SUE SENTANCE
Sue is the director of the Raspberry Pi Computing Education Research Centre at the University of Cambridge. Formerly a computing teacher, she leads a number of research projects relating to computer science and AI education, including programming pedagogy and physical computing.

Research in the US has shown that low access to computer science (CS) educational opportunities and resources is a critical driver of STEM participation gaps, which tend to mirror larger socioeconomic inequalities based on race, income, or locale (**helloworld.cc/diversity**). There is also a need for gender equity in computing education around the world. A 2023 UNESCO report on technology in education (**helloworld.cc/techined**) describes an increasing gender gap in digital skills in 50 countries, with just 3.2 percent of females able to write a computer program compared to 6.5 percent of males.

Computing in school — an overview

The 2021 Brookings report surveyed computer science in schools in over 120 countries and states, with ten in-depth case studies. The report notes that countries with mandatory CS education are geographically clustered in Eastern Europe and East Asia, with the US offering state-by-state elective (non-mandatory) computing to a greater or lesser degree. Latin America and Central and Southeast Asia have the most countries that have announced CS education programs or pilot projects. Countries in Africa and the Middle East have integrated the least amount of CS education into school curricula. The report demonstrates that there is a strong relationship between a country's income and its access to computing education: in 2021, 43 percent of students in high-income, 62 percent in upper-middle-income, 5 percent in lower-middle-income, and no students in low-income countries had access to compulsory computer science in primary and/or secondary education. It's now nearly three years since the report was published and more countries have developed computing education since then.

In this article I'll look briefly at several countries around the world where computing education is starting to develop, highlighting lessons learnt by countries that have successfully introduced computing.

North America

In the US, computing education efforts are widespread, with initiatives at federal and state levels. Many states have policies supporting CS education, aiming to integrate it into K–12 schools (ages 5–18). Since 2017, Code.org has published an annual 'State of CS' report monitoring which states have introduced elective computing (**helloworld.cc/stateofcs**). The latest report highlights that 57.5 percent of schools across the US now offer CS locally. The percentage of schools within each state offering CS varies from 28 percent (Minnesota) to 99 percent (Arkansas).

In Canada, several provinces, including British Columbia, New Brunswick, Ontario, and Quebec have implemented computing education in schools, emphasising coding and computational thinking skills.

Latin America

A recent UNESCO report details the development of CS in Latin America in seven different countries (Argentina, Uruguay, Brazil, Paraguay, Chile, Costa Rica, and Cuba; **helloworld.cc/LatinAmerica**). As an example, in Uruguay, informatics (a subject name which translates loosely to computer science or computing in England) became mandatory in the first two years of secondary school in 2012, alongside an extracurricular robotics programme since 2010. A computational thinking programme was developed in 2017 by the Ceibal Foundation that now involves 2500 teachers and impacts 30 percent of 9- to 12-year-olds. In contrast, in Paraguay, computing is offered just as an extracurricular course at a few schools by a nonprofit organisation called Paraguay Educa. In Latin America, countries face a number of challenges in introducing CS — these include defining what CS is, preparing teachers, and scaling up smaller initiatives to make them available around the country.

Africa

Kenya has become the first African country to incorporate coding as a subject in primary and secondary schools under its new competency-based curriculum (**helloworld.cc/Kenya**). The Kenya Institute of Curriculum Development recently approved a coding skills curriculum developed by Kodris Africa, a for-profit company, for 7- to 16-year-olds in the Python programming language that focuses on algorithms, debugging, and logical operators. The Raspberry Pi Foundation is also working with partners in Kenya to support the development of computing education (see pages 38–39). Other African countries, such as Botswana, are working towards the development of a curriculum for schools that includes computing (see page 32).

Southeast and East Asia

Across Southeast and East Asian countries, there are many examples of developments in computing education. In Singapore, computer science is part of the curriculum from primary to pre-university levels, emphasising computational thinking and coding from an early age, and incorporating it into various subjects. Malaysia is also focusing on integrating coding and computational thinking into its national curriculum. Indonesia has seen growing interest in teaching computing in schools, with efforts underway to introduce computing into the curriculum, supported by initiatives from the government and various organisations to train teachers and provide resources. South Korea introduced computer programming as part of its secondary-school curriculum in 2018, making it a mandatory subject. China has been actively promoting computer science education, integrating it into school programmes and offering courses in coding and programming. In Japan, computer science is being introduced in secondary schools and integrated across the curriculum in primary schools.

> RESEARCH SHOWS THAT LOW ACCESS TO COMPUTER SCIENCE EDUCATION IS A CRITICAL DRIVER OF STEM PARTICIPATION GAPS

Europe

A 2022 report by the Eurydice network (**helloworld.cc/Europe**) looked in detail at where informatics is being taught in Europe. The report covers 37 countries, including all 27 EU countries, and the findings are generally very positive. In upper-secondary education (14- to 16-year-olds), almost all countries teach informatics as a distinct discipline, and the vast majority include one or more informatics subjects (compulsory or optional) in at least one grade. In primary education, informatics is taught as a distinct discipline in 23 of the 37 education systems. Around half of them provide for a separate informatics subject that is compulsory for all pupils. Students start learning informatics from the first grade of primary education in almost a third of the education systems studied. What is taught seems to be fairly consistent; for example, in upper-secondary education, more than 30 European education systems explicitly include the areas of algorithms, programming, and safety and security. Some of the countries included have a long-standing history of teaching informatics at school; for example, in Poland, informatics has been taught since the 1990s.

What lessons have been learnt around the world?

Brookings outlined a number of lessons learnt from the case studies they undertook of locations that had successfully introduced computing, including England. I've loosely summarised these here:

- Where ICT already exists in schools, it is easier to develop and introduce computing
- Introducing computing early in the curriculum helps to create demand to study it later in school
- Developing provision for teacher development should be a top priority
- Engaging stakeholders across various areas of society (government, nonprofit organisations, industry, education policymakers) reaps many benefits
- Computing can be taught in an interactive and hands-on way, and build skills for life

To this list, I would add the provision of communities of practice. Our own experience of implementing computing education in England since 2014 has shown the importance of teachers supporting each other, and how various networks, for example Computing at School (**computingatschool.org.uk**), are instrumental in bringing computing teachers together to share knowledge and experiences. With so many countries introducing computing education, and teachers around the world facing similar challenges, maybe we need to extend this to a global teacher network, where teachers and policymakers can share good practice and learn from each other. (HW)

CANADA: TRENDS IN CSED

Matthew Griffin shares the challenges of supporting computer science education in a large country that has two official languages

The great white north. Many things may come to mind when we think of Canada: snow, politeness, hockey, poutine (the culinary delight — or horror — consisting of French fries doused in cheese curds and gravy). But, putting aside our most common clichés, we do indeed contain multitudes — multitudes that, very crucially, are the underpinnings that guide our thinking as educators of digital skills and computer science in our home.

The biggest thing that comes to my mind as it relates to education in general, and computing education in particular, in Canada is just that: the sheer size of the place. It's massive. Like, bigger than Europe massive. But there really aren't that many people, spread across a vast area, and one of the major challenges of delivering equitable computer science and digital skills education across Canada is how we reach them all. This speaks to the higher-level issues of computer science education (CSed) in Canada, in that the 'how' very often supersedes the 'what'.

Our organisation, Digital Moment (**digitalmoment.org**), has a small team and a big mission. For just over a decade, we have aimed to bring computer science and digital skills education to teachers and students all across the country. Now, we are one small piece of a large, interconnected education system, but I would hazard a guess that our challenges are analogous to the larger trends across the country.

MATTHEW GRIFFIN

Matthew Griffin is an educator and artist from Montreal, Canada. He currently works as the senior manager of educational programs at the organisation, Digital Moment, and he has taught some combination of art and technology from the primary to the post-secondary level. He is driven by a passion to build equity in learners through the use of digital skills and creativity.

Urban centres

As in many other countries, our population is concentrated in a few urban centres. Unlike in most other countries, those centres are hundreds or even thousands of kilometres removed from each other. As a consequence, communities not close to one of those centres are often historically underserved, and resources are concentrated in a few places, largely inaccessible to significant portions of the country. Traditionally, this has been a nearly impossible problem to solve. Where do we put our teams but the most populated areas, in order to have the largest (numerical) impact? Does this decision drive or reinforce inequity? What alternatives could lead to genuine change for the better?

In some ways, the massive shifts in education (and education delivery) brought by the Covid-19 pandemic have been extremely useful in increasing equitable access in a place like Canada. Online presentations are now an understood and accepted delivery method, meaning that any class, anywhere in the country with a reliable internet connection, can receive instruction from anywhere else.

The obvious caveat, though, is that not everywhere has a reliable internet connection, particularly in rural and northern communities throughout Canada. The less obvious caveat is that there's still the challenge of awareness. If computer science resources have historically been accessed by (and accessible to) certain areas, those areas are naturally are more inclined to seek out those resources. As a consequence, the focus of organisations like ours is not only to make resources available to as many communities as possible, but also to actively seek out those communities, to help build awareness and collaboration.

Disjointed education systems

Of course, as a charitable organisation, we operate somewhat removed from the formal education system, or rather, systems. Another unique challenge in our country is the fact that in this vast land, there isn't one national education system, but rather 13 systems, one for each of our ten provinces and three territories. And remember, each of these provinces and territories are country-sized, and each has slightly different curricula. This means that for newer subjects

Digital Moment is based in Montreal, Canada

such as computer science education, each of them is at a slightly different point in the adoption of the subject, and not all of them adopt it in the same way (for example, is it a subset of mathematics, or science, or a discrete, new area of study?). This is particularly true in the younger grades, and is an extremely relevant concern for organisations like ours, which aim to bring computer science to schools everywhere. So, when we are asked, 'That was fun, but how do I connect it to my class?', we need to be equipped with not one answer, but many.

Two official languages

We are also a country with two official languages — English and French — and it is very important to consider this as we develop materials and deliver lessons in a way that builds equity. It's not only about the act of translation, but unique histories and cultural touchpoints, in addition to the curricular concerns mentioned above. In fact, building an empathetic understanding of one's audience is fundamental in education, a fact that can sometimes get lost in computer science. So perhaps the fact that our linguistic uniqueness forces us to consistently consider this can be seen as a feature, not a bug, to borrow a well-used computer science phrase.

Adapting to our audience

In our decade working in this space, we've seen profound changes in the landscape of computer science education in Canada. We began as an organisation that specialised in giving students and teachers their first taste of coding. Generally, our audience was quite young — we did the majority of our workshops with students aged ten to twelve. We would typically do our activities in block-coding environments, most often Scratch and MakeCode. The activities were simple — cats chasing balls, or Fortnite dances on the micro:bit. Provincial curricula are slowly building coding expectations into their respective education systems, and the resulting sophistication of young people has been remarkable and gratifying to see. It has allowed organisations like ours to expand our thinking about what computing education can and should be, including, as mentioned, cultural relevance, but also things like placing these skills and these problems on a larger stage, considering computing education in the context of social and climate justice, skills building for careers, and ethical implications. As a result, we now find ourselves using the tools and platforms not only to teach the foundations of coding and computer science, but also to design data collection projects that aim to solve global environmental problems.

Artificial intelligence

One rapidly emerging area of interest for students and educators in Canada (and for everyone everywhere) is artificial intelligence (AI). We began delivering AI workshops four years ago, and at that point, we were mostly reaching early adopters and brave, curious educators. Our organisation is based in Montreal, Quebec, which happens to be one of the global hotbeds of AI. As such, we are extremely fortunate to be surrounded by a brilliant and collaborative community of thinkers, which led us to be quick out of the gate with AI content. However, it's a complex balance. We know this is relevant and interesting information, but once again it comes down to a notion of equity: this isn't written into curricula yet, so how do we build awareness for all communities, and talk about this vital subject in a way that is relevant, interesting, and useful to all of our audience?

This tension in the AI space is exacerbated by the sheer speed with which the field is moving. It's a nearly impossible challenge to keep up with all the new tools, let alone build content to teach about them. And, of course, the technical sophistication of AI systems is far beyond our traditional wheelhouse of introductory or early awareness for generalist educators. As such, our aim is to build conscientious, ethical users and builders of AI technologies, and we are seeing momentum building in a wider adoption of these goals across the country.

The future

I think, fundamentally, that the trends in computer science education in Canada are not unique. Perhaps there are certain things that we focus on more due to the size of the country, but ultimately, our goal is to prepare our youth to build a better future. We do that through digital skills, but however educators teach computer science, we share a drive and a vision with our collaborators globally. It's a privilege to work in this space. There's a genuine excitement surrounding the field, and it's not something we take for granted, nor is it a responsibility we take lightly. We've seen such fundamental change in the past decade, and we're extremely excited to see what comes next. (HW)

USA: TRAINING COMPUTER SCIENCE TEACHERS

Two different approaches to using *The Big Book of Computing Pedagogy*

COREY ROGERS

Corey, a computer science and digital learning consultant at Grant Wood Area Education Agency, supports PK–12 teachers to grow their identity, knowledge, and skills as CS educators. She enjoys reading and playing overcomplicated board games (**@corey_rogers**).

BEN SCHAFER

Ben is a computer science professor at the University of Northern Iowa who focuses on training preservice and in-service computer science teachers. If he could figure out how to do that from a kayak, he would (**@CSEDatUNI**).

In 2020, the state of Iowa in the United States passed a legislative bill requiring computer science (CS) at all levels of K–12 education (ages 5–18). This means that educators from a variety of backgrounds now find themselves teaching computer science, often with little training. We are both CS teachers and have been supporting educators as they grow their own understanding of CS. As we have done so, it has become clear that it's important for educators not only to learn what to teach about CS, but also how to teach it.

A new resource

The Big Book of Computing Pedagogy (BBCP), Hello World's first-ever special edition, was released in 2021. Both of us had printed the BBCP when it first came out and were intrigued by what it had to offer. We were drawn to the organisation — the grouping of related articles under the banner of twelve principles of computing pedagogy. We appreciated that the articles were concise but informative, and the fact that each of the articles was limited to between two and four pages, yet many contained well-curated references for when we wanted to learn more. Finally, we were struck by the tone and style of writing used in the articles, and how this could help build teacher CS identity within these easy entry points into the discipline of CS pedagogy.

The overall potential of the book really took off for us at the Computer Science Teachers Association (CSTA) 2022 conference, where free copies were available to all participants. Knowing that over 100 Iowa educators attended the conference and had picked up a copy, we spent part of our trip home discussing how we could use the book with our respective audiences.

An informal online course

Corey is a digital learning consultant with one of Iowa's area education agencies. She first started exposing teachers to the BBCP by organising an online book discussion group. Corey split the book into two groups of six pedagogical principles each. She then designed two five-week virtual courses that relied heavily on building community by talking about the BBCP articles. Each course had a one-hour kick-off and a one-hour wrap to outline course objectives, build community, and solidify learning. The remaining three sessions required participants to summarise the two identified pedagogical principles, select and read at least two articles from each principle, and record key takeaways, important vocabulary, questions about the articles, and ways they either have used or might use the concepts from the articles in their classroom.

During the virtual sessions, Corey modelled one of the pedagogies and then facilitated breakout rooms where participants could discuss the articles they read using their notes and a thinking routine. Participants from Iowa, Maryland, New York, and Texas, ranging from elementary to high-school classroom teachers, teacher librarians, instructional coaches, and content consultants, reported growth in their understanding of CS pedagogies as a result of participating in the courses. Corey felt that the magic of these courses was the conversations in the breakout rooms. Educators from different states serving in different roles were able to make meaningful connections, share successes, and problem-solve missteps when using the pedagogies.

A formal undergraduate course

Ben is a professor of computer science education and works with both preservice and in-service educators working toward the state of Iowa's computer science endorsement on their teaching licence. One of the courses in this sequence is *Teaching and Learning Programming*, which has historically been a collection of readings about CS-specific pedagogy, lesson design, assessment, and equity, all while participants completed additional programming practice. Ben discovered that most of these topics were in the BBCP, but in a much more organised and informative manner than the collection of readings he had previously created.

> THIS RESOURCE BELONGS ON EVERY TEACHER'S BOOKSHELF

In the spring of 2023, he taught two sections of this course — one on campus with preservice undergraduates, and one online with in-service teachers. Much like Corey's reading groups, Ben divided up the book based on the twelve principles. In a typical week, participants would read the articles about one of the principles, research at least one additional resource from the reference list, write a reflection/response based on their readings, participate in a small-group discussion, and then submit a connection paper discussing how the material fits into their current or previous classroom experiences.

FEEDBACK FROM COREY'S INFORMAL ONLINE COURSE FOR IN-SERVICE TEACHERS

What was something you found valuable about this course?

- *The conversations were really valuable. When there was an article I didn't understand as well, it was good to hear the perspective of someone else.*
- *Collaboration with other teachers and realising we are all in the [same] boat and are dealing with similar issues.*
- *I really like the articles and learnt a lot from them. I liked the structure of the course with the breakout rooms where we could discuss the week's topics and sometimes make more sense out of what we read. I really needed that for the semantic waves. I read the information multiple times before I even began thinking I might understand it. I was pleasantly surprised when others in my breakout group had the same response to that article. It made me feel less incompetent! Also, I understood the concept much better after we discussed it as a group.*
- *Peer interaction related to how others applied the pedagogy/strategies to their content/ grade instruction. Others' views on what the pedagogy/strategies really entailed. I didn't always completely understand how each strategy would be used and what it was exactly.*
- *The conversation about the articles that we read. Being able to hear from a different grade level and different area helped gain even more perspective about the article and how to approach it.*

Which pedagogy or strategy is your top takeaway from the course?

- *'Code tracing' and 'Read before you write'. These two just seem to go together. It's interesting how some things that seem so obvious need to be pointed out. I feel like these are like that. Why wouldn't we trace code? Why wouldn't we teach to read code first? If [students] begin to understand coding first by reading it, coding will be so much easier for [them].*
- *'Code tracing' and 'Read before you write'! It was an 'Aha' moment for me. Like a 'duh!' It really isn't any different than any other content area.*
- *'Cognitive load theory' — CLT has implications in multiple areas of education and I want to share this with teachers!*
- *For some reason I am really drawn to the 'Peer instruction'. It hits engagement, discussion, critical thinking, helps uncover misconceptions, and it is a strategy that can be used in any discipline.*

The course continues to include programming activities to strengthen participants' skills, but these activities are now designed to connect them with a technique or skill discussed in the weekly readings. Additionally, Ben recently gave a talk about education to graduate students at the University of Minnesota, where he gave each participant a copy of the BBCP. He was pleased and amazed at how excited these students were to have a resource to help them think about teaching and learning.

How will you use the BBCP?

In the end, it doesn't matter whether you are a formally trained educator or a novice thinking about teaching your first class or coaching your first club — we all can use advice on how to structure our computer science teaching and how to create positive experiences for our students. We believe that the BBCP (**helloworld.cc/BBCP**) is an invaluable resource that belongs in ready reach on the bookshelf of every computer science teacher. (HW)

FEATURE

MEXICO: VOLUNTEERS INSPIRING THE NEXT TECH LEADERS

Volunteers are at the core of Ciberistas (formerly Axt@Teen), a nationwide programme that inspires young people with technology training

In the heart of Monterrey, Nuevo León, Mexico, a transformative programme was born in 2017: Axt@Teen, now known as Ciberistas, an initiative by CSOFTMTY (Cluster TIC).

This initiative was conceived with the primary goal of increasing the number of children opting for careers within information and communication technology (ICT) and engineering. The primary aim is to have a positive impact on the social and economic landscape of the region, given the well-paid nature of these professions and the good availability of job opportunities within these areas.

ILIANA RAMIREZ

Iliana is a software engineer, teacher, and human capital director of CSOFTMTY. She has shaped and led programmes to motivate and teach young people about IT for more than ten years.

Needs versus reality

There is substantial demand for technology and computing professionals in Mexico. However, a critical imbalance persists, as underscored by the 2017–2018 labour market guide prepared by the consultancy Hays Mexico, cited by Echeverria (**helloworld.cc/Echeverria**). It emphasises, "The rapid evolution of technology and the advent of industrialisation 4.0 are exerting profound effects on the Mexican labour market, resulting in a pronounced scarcity of local talent to meet the burgeoning demand in science, technology, engineering, and mathematics (STEM) fields."

Compounding this challenge is the stark reality that children enrolled in private schools enjoy significantly better access to computing devices, the internet, and technology and computer science education than their counterparts in public schools (which 90 percent of children attend). According to public body INEGI (**helloworld.cc/INEGI**), a concerning 20 percent of Mexican families continue to grapple with digital illiteracy. Furthermore, between 2015 and 2017 there was a decrease from 41 percent to 38 percent in the number of individuals aged 18 and above who were interested in pursuing engineering studies. Additionally, Poy-Solano (**helloworld.cc/PoySolano**) reports that less than 38 percent of public schools have internet connectivity for pedagogical purposes.

Ciberistas

In response to this complex scenario, Ciberistas has emerged, aided by collaboration between universities, businesses, and the state government. The main objective is to democratise access to technological education and increase proficiency in computational thinking, regardless of the origin and type of school.

Since the beginning, volunteers have been the essence of our programme. Their active involvement, motivation, retention, and responsible management significantly influence the programme's success.

Ciberistas initially started as a summer camp with the goal of training fewer than 200 kids aged 9–17 in information technologies through in-person workshops. Fast-forward to 2023, and the initiative has evolved into a powerhouse, with 1114 enrollees spanning 18 states in Mexico and approximately 15 Spanish-speaking countries. The programme has not only grown in numbers, but has also adapted to the digital age, offering free year-round workshops both virtually and in-person.

Inspiring young minds

Ciberistas' primary mission is to inspire young minds to consider careers in engineering and technology, equipping them to navigate the challenges posed by industry 4.0 (the fourth industrial revolution). The programme's enduring success and its expansion over the years can be attributed to its beating heart: these dedicated volunteers.

Diverse volunteers, united purpose

The volunteer landscape at Ciberistas is very different from its early days, when adults — parents, university teachers, and industry professionals — predominantly filled the ranks. This group has grown into one that is much more diverse, with volunteers expanding their roles beyond course preparation and teaching to create a vibrant and supportive community.

In recent years, we've observed a preference among participants for young university students as instructors, fostering a closer connection; yet we maintain a mix of volunteers who play crucial roles:

- **Champions:** devoted individuals committed to helping underprivileged children pursue futures in technology. They leverage Ciberistas' processes, connect with organisations, gather resources, and implement programmes in their respective regions.
- **Volunteer leaders:** guiding groups of volunteers, ensuring a safe environment for participants and trainers, they assist in selecting suitable workshops and recognising diverse skills and experiences.
- **University teachers:** specialised professors in digital topics such as quantum computing, competitive programming, or design thinking. They supervise, create content, and teach.
- **University students:** this is the largest volunteer group, with engineering majors fulfilling social service requirements by teaching courses. Students from various disciplines contribute to content creation, marketing campaigns, social media management, and data analysis.

In particular, we allow these students to shape their courses and incorporate additional activities for icebreaking. For university students, it is crucial for them to understand that they are contributing value, positively impacting lives, and feeling a sense of autonomy in creating their learning environment.

Due to their youth and limited experience in working with children, this group requires more training and supervision. More seasoned volunteers are crucial in ensuring they successfully undergo and understand their safeguarding training. Additionally, these seasoned volunteers assist the younger ones in addressing any challenges.

Learners at Axt@Teen, now known as Ciberistas, a programme where young people participate in technology workshops

Furthermore, we have noticed a preference for in-person workshops among young volunteers, who appreciate the opportunity to interact directly with the children. Additionally, children tend to form stronger connections with their instructors and feel more at ease asking questions, not necessarily confined to technical aspects, when their teachers are younger.

Volunteers fostering gender balance

Our commitment to gender balance involves hosting workshops exclusively for girls, led by female university instructors and women in technology groups. Courses related to art, animation, design thinking, web design, and storytelling have been successful in attracting and retaining girls, contributing to the result that around 50 percent of our participants identify as women. One of our most successful courses with girls has been the one teaching Snap! (**snap.berkeley.edu**).

Volunteer voices

In our quest to attract volunteers, former volunteers play a crucial role in sharing relatable experiences. That is why we ask them to create content sharing their experiences (**helloworld.cc/volunteervoices**).

I encourage you to let your volunteers talk; their feedback and voices invite others to take part. Testimonials such as the ones that follow highlight the profound impact of the programme on their careers, and the joy derived from teaching:

"This project has had a direct impact on my career. It was incredibly gratifying to see the students bring to life the games that existed in their minds, despite their limited experience and time constraints. I am proud of their achievements and grateful for the lessons learnt, which now contribute to my professional development."

"The programme is quite entertaining; it helps you understand programming better, as you try to grasp it in a way you can teach it."

"Teachers learn and significantly enhance our skills thanks to the students themselves. This prepares all of us to face the different stages of life better equipped, more prepared, and more resilient." (HW)

FURTHER READING

Echeverría, M. (2018, February 26). "Hays México presenta su guía laboral 2018". The Logistics World. **helloworld.cc/Echeverria**

INEGI. (2023). *PECyT – Encuesta sobre Percepción Pública de la Ciencia y la Tecnología.* **helloworld.cc/INEGI**

Reyes, E. (2023, June 6). *En México hay un 20% de familias analfabetas digitales.* Expansión. **helloworld.cc/Reyes**

Poy Solano, L. (2022, March 3). "Sólo 37.3% de escuelas tienen Internet con fines pedagógicos: reporte." La Jornada. **helloworld.cc/PoySolano**

FEATURE

PERU: COMPUTING IN AN OFFLINE WORLD ON STILTS

Dana Rensi shares her experience of working with a school to create a Raspberry Pi computer lab in the Peruvian Amazon

In a large corner classroom of a school which stands two stories high on stilts, in the floodplain of an Amazon River tributary, there are 25 Raspberry Pi workstations powered by solar panels on the roof. The San Francisco Rio Itaya School is in Iquitos, the world's largest city in the Peruvian rainforest, and can only be accessed by boat or plane.

School in Iquitos

Surrounding the school are log or slat homes that are either on stilts or are built to float on the river. Children play and swim in contaminated water used for drinking, laundry, and waste disposal. During the rainy seasons, they and their teachers arrive at school in home-made wooden boats.

DANA RENSI

Dana is a former Spanish teacher, educational technology specialist, and librarian. She came to the Peruvian Amazon on a Fulbright Teacher Exchange and now puts her career experience to use as regional director for Latin America at Powering Potential, Inc. (**@DanaRensi**, **danarensi@poweringpotential.org**).

Each school day is split into two sessions: primary grades (ages 6–12) in the morning and secondary grades (ages 13–18) in the afternoon. The school is the pride of the community and the site of year-round holiday festivities. Its academic ranking, however, is low, typical of Peru's remote, underserved areas.

Creating an offline computer lab

The classroom was installed in 2019 by professional staff of Powering Potential, Inc., who volunteered their time. Powering Potential is a US nonprofit with prior experience creating 36 such computer labs in rural Tanzania since 2007 (**poweringpotential.org**). Its mission is to provide technology equipment and training to enhance education in developing countries. The Raspberry Pi workstations in the SPARC+ (Solar-Powered Access to Raspberry Computing) lab in San Francisco Rio Itaya School are networked to a local RACHEL Plus (Remote Area Community

The San Francisco Rio Itaya School is built two stories up on stilts because flooding is an annual event in the lower section of Belén, Peru

Hotspot for Education and Learning) server from World Possible, a nonprofit which curates open educational resources for K–12 education used in more than 40 countries around the world (**worldpossible.org**). Powering Potential provided a RACHEL Plus 1TB, including Wikipedia, Khan Academy videos, UNESCO textbooks, Project Gutenberg titles, apps for learning music and English, medical information, Scratch, and more. Funded by Learning Equality (**learningequality.org**), this project uses its Kolibri structured learning environment.

RACHEL Plus is an appliance that provides the Wi-Fi LAN as well as the educational content, so that the lab functions offline where internet connectivity is costly and/or unreliable. While the school is on the power grid, electricity is not reliable. It's also not available in most of the Powering Potential schools, so we install solar power to model sustainable, eco-friendly solutions. This has been the most expensive component of the installations. But with the low power draw of the Raspberry Pis, coupled with energy-efficient monitors that can run on the 12V DC output, the solar systems have been kept as small as possible.

Reassessing priorities

Anita Gil Avila, director of the San Francisco Rio Itaya School, cited mathematics as the first learning priority, based on the students' poor showing in annual national exams for fourth-year primary (ages 9–10) and second-year secondary grades (ages 14–15). In the fall of 2019, in my role as regional director for Latin America for Powering Potential, I began piloting classes for teachers and students to learn to use Khan Academy videos through the Kolibri platform. Discovering the lack of basic arithmetic skills among the students led me to devise analogue activities with whiteboards, manipulatives, and flashcards. By the time the school year ended in December, the computer classroom was a point of local pride and the director was

Dana Rensi, Powering Potential's regional director for Latin America, is proud to see the SPARC+ lab in action

using its Raspberry Pi with a Pico projector for meetings.

Pandemic setback

Multiple waves of Covid-19 outbreaks left Peru with one of the highest pandemic death rates in the world, and kept the school closed for more than two academic years. National attempts to reach children at home via online, radio, or television programmes did not succeed in this area, where many lack equipment or electricity. The resulting learning loss and achievement gaps are considerable, and the computer classroom is one part of the effort to raise student achievement.

Back to school

When I was able to return to San Francisco Rio Itaya School in October 2022, I first had to repair rain damage and update the software. By November, multiple classes were coming to the lab each week. At the start of the 2023 school year in March, the director asked that Powering Potential provide weekly mathematics reinforcement as mandated by the education ministry. I set up Kolibri classes with Khan Academy maths modules sequenced according to Peru's curriculum, so that students in the fourth grade and up could work at their own pace to learn and practise arithmetic and maths skills in twice-weekly 45-minute sessions. Khan Academy videos are usually five minutes or less in length, and as framed by Learning Equality, there are questions and explanations based on student responses. Students can also replay segments they don't understand, and teachers can see each student's progress and the areas they need to work on.

In order to ensure that the lab is independent and sustainable, I worked with the regional education administration to ensure that it was recertified in 2023. Along with this, two new teachers will be assigned to the school for the 2024 academic year. I plan to be in Iquitos for the month of March, to update the lab and train the 'innovation teachers', as technology teachers are called in Peru. While we look forward to being able to teach programming and web design, we're pleased to work with the local teachers and administrators to model the creative use of educational technology to enhance learning in basic subjects where the students need the most help at present. (HW)

FEATURE

GUYANA: EMPOWERING GIRLS IN ICT THROUGH 21ST CENTURY SKILLS

Penelope DeFreitas describes the Guyanese Girls Code programme

The underrepresentation of women and girls in the field of information and communication technology (ICT) is a growing concern. In 2021, for instance, the European Union reported that only 17 percent of ICT specialists in its member countries were female (helloworld.cc/EU2021), while data from the University of Guyana revealed that from 2009 to 2021, the number of female graduates with computing degrees was consistently lower than the number of male graduates.

The ICT sector is one of the fastest-growing industries globally, and there's a projected increase of 90 percent in jobs requiring digital skills, with nearly 50 percent expected to undergo transformation due to automation by 2030 (**helloworld.cc/WEforum**). Given the imminent shortage of ICT skills, it is increasingly important to fast-track better representation of women in this field. To address this deficit, leaders worldwide are initiating education-driven programmes aligned with the UN's 2030 Agenda for Sustainable Development. A notable example is the Guyanese Girls Code (GGC) programme, designed to introduce young females in grades 7 to 9 (ages 12–15) to the realm of ICT. This initiative places particular emphasis on cultivating essential twenty-first–century skills, including problem-solving and critical thinking.

PENELOPE DEFREITAS

Penelope is a computer science educator at the University of Guyana. She has over a decade of experience contributing to academia and actively participating in community projects focused on fostering the upliftment of youth.

The training model behind Guyanese Girls Code

The MST-tree model, conceived as a metaphor for professional growth in the ICT sector, delineates three crucial components: motivation (M), support (S), and teaching (T). It was crafted as a framework for guiding curriculum design and implementing effective teaching strategies for delivering ICT training, such as the GGC programme. Within the motivation component, elements such as role models, positive teacher perception, parental involvement, and opportunities to explore the social applications of ICT were identified as instrumental in fostering high levels of self-efficacy and interest. The support component, highlighted for its pivotal role in creating a conducive environment and ensuring access to training opportunities, aligns with global calls for collaborative efforts in education. The teaching component, focusing on skills such as creativity, problem-solving, and critical thinking, was seen as essential for preparing students for the twenty-first century. Furthermore, activities involving tools such as Scratch and the BBC micro:bit were found to be particularly effective in exposing young learners to these crucial skills.

The model underwent its initial testing in the face-to-face mode in 2018, unveiling some encouraging outcomes. Notably, female participants exhibited increased interest in delving into ICT, despite perceiving the subject as challenging. Additionally, the participants demonstrated higher levels of self-efficacy, particularly in their engagement with the programme modules they perceived as the most challenging (**helloworld.cc/Layne2020**).

Follow-up evaluations conducted in the virtual setting during the Covid-19 pandemic revealed an enriched dimension to the GGC model. It was revealed that mentorship, and fostering a sense of community among learners, played crucial roles in providing support for the participants. Additionally, game-based activities, live demonstrations, breakout rooms, and projects were identified as effective strategies for delivering the programme in a virtual setting (**helloworld.cc/DeFreitas2023**). (HW)

THE GAMBIA: TEACH THE TEACHER PROGRAMME

Graham Hastings updates us on his experience of developing a computing curriculum with a school in The Gambia

If you have read my article on pages 68–69 of Hello World issue 19 (helloworld.cc/19), you will have learnt about the Teach the Teacher (TtT) programme, The Gambian Education Ministry's programme to create a national web of six teacher training centres across the country. TtT was established to explore the potential of cloud-based computing in elevating education in one of the world's most economically disadvantaged countries.

How we began

The school at the centre of this ambitious project, Mansa-Colley Bojang School, had no computers, and like most schools across The Gambia, it also lacked internet access. All teaching relied on traditional methods, such as paper and pencils. We worked with TtT to provide the school with 60 second-hand Chromebooks, internet access via a Starlink dish, a simple Wi-Fi network, a large touchscreen TV, and a sound system. We also added an uninterruptible power supply to provide backup to cover the frequent power outages. To manage the content, we created accounts for teachers and pupils via Google for Education.

GRAHAM HASTINGS

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

Teaching the teacher

With the technology relatively stable, we turned our attention to teacher training and equipping the pupils with basic computer skills. We initiated a collaboration with the school's two ICT teachers, who had no prior experience with online teaching but were experienced laptop users.

Cultural and social challenges

From the outset, we realised that the more significant challenge would not be technological, but rather cultural and social. We quickly understood the need for extreme tact and sensitivity in introducing the use of computers in the teaching and learning methodologies at Mansa-Colley Bojang School.

One of the greatest challenges for the teachers was accepting that they could no longer be the sole keepers of knowledge in the classroom. Instead, they needed to adapt to a more supportive and guiding role, helping students explore and discover knowledge themselves. They had to embrace the role of a facilitator, encouraging students' curiosity and problem-solving skills.

Grade 9 pupils (ages 15–16) using Chromebooks for the first time

Future goals

After the second full year of operation, we have concluded that the TtT model is an affordable and highly effective approach that helps schools in economically disadvantaged countries to deliver a similar level of education to those in the world's wealthiest economies. Our goal for next year is to train more teachers, secure additional funding, and scale the programme up to establish seven more TtT hubs spread throughout The Gambia.

Acknowledgements for support with the TtT programme go to Mansa-Colley Bojang School staff and pupils; The Gambian Education Ministry; Levi Thomas Foundation; St John's College School, Cambridge; and Rotary Club. (HW)

BOTSWANA, NIGERIA, KENYA, AND UGANDA: THE CAPE FRAMEWORK

Bridging the gap between developed and developing countries in computer science education

As the world embraces K–12 computing education (ages 5–18), concerns about equity and inclusion in these initiatives come to the forefront. Despite extensive research in high-income countries, there remains a significant gap in understanding computing education in low-middle-income nations, particularly in regions such as Eastern, Southern, and Western Africa. I will explore the critical significance of addressing this disparity and discuss efforts to bridge this gap, fostering equitable computing education on a global scale.

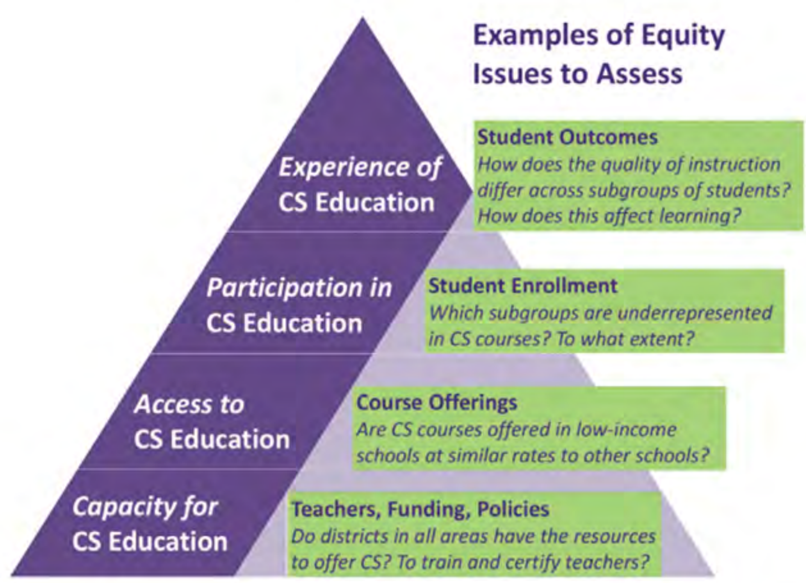
ETHEL TSHUKUDU
Ethel is a computer science education researcher and scholar who is enthusiastic about making CS education accessible to all. (**@EthelTshukudu, linkedin.com/in/dr-ethel-tshukudu-b9965338**)

The CAPE framework

The CAPE framework aims to assess equity throughout the computer science education ecosystem (**helloworld.cc/CAPEframework**). It can help to make sure everyone gets a fair chance to learn about computer science in schools. It focuses on four areas: capacity, access, participation, and experience. These areas are all connected, and they all need each other to work well.

At the top of the framework is Experience. This part is about how students feel and what they learn when they study computers. It's important that everyone feels they belong and can do well in computer classes. To make this happen, teachers need to use teaching methods and lessons that work for everyone, making sure all students have a good time learning and achieve similar results.

Before students can have this good experience, they need to take part in computer science classes. This is Participation. It means that everyone, no matter where they're from or who they are, should have the chance to



The CAPE framework

Country	Botswana	Kenya	Nigeria	Uganda
Population (million)	2.35	54.5	206	44.3
No. of schools	1,112	89,361	20,314	129,734
No. of students	520,110	16,060,000	27,900,000	10,220,172
No. of teachers (FTE)	30,311	496,801	834,613	125,883
African region	South	East	West	East
Income classification (World Bank[1])	Upper-middle	Lower-middle	Lower-middle	Low

Table 1 Characteristics of the selected African countries (2020)

learn about computers. Sometimes, not all students get this chance, perhaps because of their family situation, their gender, or where they live. For example, in some places, boys are more likely to take these classes than girls. To fix this, schools need to encourage everyone to join in.

> FUNDING IS A CRUCIAL BARRIER, AS HIGH-INCOME COUNTRIES ALLOCATE MORE RESOURCES TO COMPUTING EDUCATION

But even before joining in, students need Access to computer science classes. This means they should have the chance to learn about computer science no matter what their socioeconomic status is. Sometimes, schools in different areas offer different opportunities to learn about computers. Schools with more students from families with incomes above the poverty threshold often offer more computer science classes. To make things fair, we need policies to make sure all schools offer enough classes for everyone.

For schools to offer these classes, they need Capacity. This means they should have enough resources, such as trained teachers, funding, and policies that support the effective teaching of computer science classes. Sometimes, schools in different places don't have enough trained teachers for these classes. This can be a bigger problem in rural areas. So, policies need to help schools in these areas access better teacher training and improved resources.

In our research, we adopted the CAPE framework to assess the state of computer science education across four African countries (Botswana, Nigeria, Kenya, and Uganda), each with a very different economic status (**helloworld.cc/Tshukudu**). Our evaluation involved a METRECC (Measuring Teacher Enacted Computing Curriculum) survey administered to 58 teachers. We focused specifically on the capacity level, analysing whether these countries have suitable policies, trained teachers, and ICT infrastructure. Furthermore, we compared our findings from the African countries with those from seven high-income countries: Australia, England, Ireland, Italy, Malta, Scotland, and the United States.

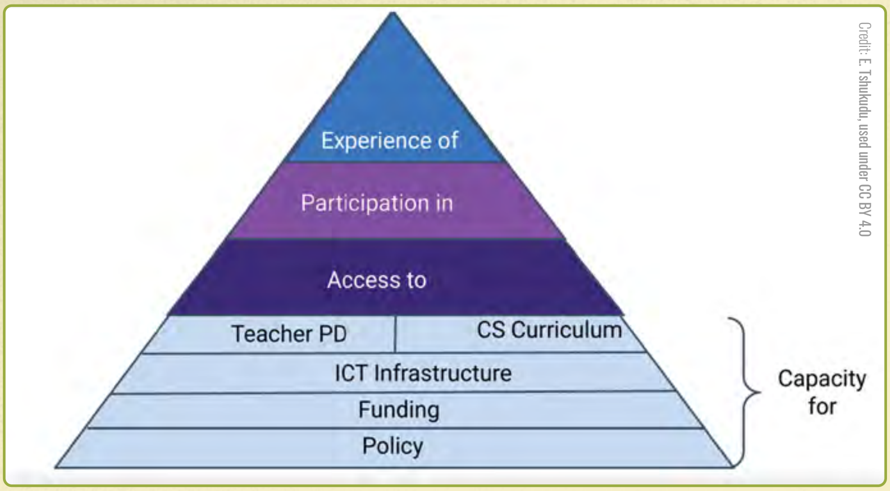
Curriculum focus

The emphasis on teaching programming skills varies widely across different countries. For instance, Uganda, classified as a low-income nation, places minimal emphasis on programming skills, while Kenya, a low-middle-income country, shows a slightly stronger focus. These variations intertwine with factors such as curriculum influence and the availability of infrastructure, subtly shaping the educational landscape.

Comparatively, high-income countries such as England and the US have advanced their computing curricula significantly. They prioritise the teaching of computational thinking, algorithms, and programming skills, even at early educational stages. In contrast, African nations tend to prioritise fundamental ICT skills, with programming being a minor, if present, component in their educational programmes.

ICT infrastructure

The availability of ICT infrastructure varies considerably. High-income countries typically boast better-equipped schools with widespread technology access. In contrast, African countries encounter significant hurdles in providing adequate ICT infrastructure, particularly in low-middle-income areas. For instance, Botswana, categorised as an upper-middle-income country, exhibits extensive access to ICT infrastructure. Notably, it recently provided laptops to every senior secondary-



Extending the CAPE framework for the international context

school student. This stands in contrast to Uganda, where teachers report the lowest access to ICT infrastructure; here, only 33 percent have access to a computer lab. This highlights a correlation between socioeconomic status and the accessibility of resources.

Teacher professional development

Access to professional development varies significantly. High-income countries allocate more resources to train teachers to deliver modern computing education, while African nations face challenges, with limited support and opportunities for relevant training. On average, only 17 percent of African teachers reported receiving time off for professional training, compared to 76 percent in high-income countries. However, a higher percentage of Kenyan teachers (50 percent) reported receiving time off for professional development than in other African countries.

Funding constraints

Funding remains a crucial barrier globally. While high-income countries generally allocate more resources to computing education, disparities do exist within these nations. However, the lack of funding significantly hampers the effective implementation of computing education in African nations. Our findings show that only a small percentage of teachers in the four African countries we looked at (an average of 16 percent) have access to funding for purchasing computing equipment. Specifically, Uganda (11 percent) and Nigeria (7 percent) face the greatest challenges in acquiring funding for such resources. These countries also lack funding for basic electricity, a critical requirement for utilising ICT in schools.

Policies

Some policy frameworks in high-income countries have progressed to incorporating computational thinking, digital education, and even artificial intelligence in their education systems, adapting to rapid advancements in computing. In contrast, African nations predominantly prioritise deploying ICT infrastructure in secondary schools, often lacking policies that cater to the evolving needs of computing education.

Revisiting the CAPE framework

After reviewing the findings of our study, we revisited the CAPE framework, proposing a more detailed model, specifically within the Capacity level. This refined model highlights different stages of implementing computing education in schools. The subcomponents at the Capacity level, based on our study, encompass policies, funding, ICT infrastructure, curriculum, and teacher training. Our research indicates that while African countries have moderately sufficient policies, with commendable recommendations, their implementation

Country/Age	16–18 years	14–16 years	11–14	6–10 years
Botswana	E	C	N	N
Kenya	N	E	N	N
Nigeria	E	C	C	C
Uganda	E	E	N	N
C = Compulsory; E= Elective; N= Not taught				

Table 2 Computer science teaching across the selected African countries

faces hurdles due to limited government funding. This means that until the basic needs of funding ICT infrastructure in schools are met, these countries cannot prioritise other aspects, such as computing curriculum and teacher professional development. Once the Capacity level is met, the low-income countries may be able to focus on other efforts to broaden participation.

Teachers have also highlighted insufficient funding for teacher professional development and ICT infrastructure. Funding has been reported as a critical issue across all African countries. This shortage may explain why these nations, unlike high-income countries, still grapple with resource capacity. Addressing this deficit requires policy recommendations that prioritise funding. Notably, the policies themselves seem adequate, and so deficiencies at this level appear to be linked to the economic status of the countries.

For effective implementation, professional development and curriculum advancement must advance together. This means that teachers need training that is aligned with the curriculum content, while the curriculum's effective execution relies on well-trained educators. Our findings highlight that achieving sufficient teacher professional development and implementing the curriculum might hinge on adequate ICT infrastructure, funding, and appropriate policies in schools, as observed in high-income countries.

Moving forward, it's crucial for countries to collaborate and prioritise funding for ICT infrastructure, acknowledging its pivotal role in computer science education. Additionally, revising policies to meet evolving educational needs, with a focus on fair resource allocation and improved professional development opportunities, is crucial. These collective endeavours aim to fortify computing education and narrow resource disparities among locations with diverse economic statuses. (HW)

COMPUTER SCIENCE EDUCATION BOTSWANA (CSEDBOTSWANA)

This research affirmed my commitment to equity and inclusivity within computer science education in Botswana, one of the countries covered in the study. As a result, I established CSEdBotswana, aligning its goals with the country's digitisation strategy, aiming to offer every student in Botswana access to computer science skills. The CsedBotswana initiative is situated within the University of Botswana and engages diverse stakeholders, including local and international researchers, government bodies, and industry representatives. The initiative focuses on fostering support and awareness, while expanding access and inclusivity in computer science education across Botswana's schools. Collaborating closely with the Ministry of Education, the initiative seeks to do non-formal coding events, and to comprehensively evaluate and advocate for the computer studies curriculum to meet global standards. Moreover, CSEdBotswana actively advocates for teacher professional development in the field of computer science education. Central to the success of CSEdBotswana is its distinguished advisory board, composed of eight experts and leaders in the fields of industry, education, and computer science.

In a year, CSEdBotswana has marked significant milestones:

Coding clubs: introduced 20 code clubs in Botswana's junior and senior schools, including Baylor Clinic. These clubs, formed in collaboration with the Raspberry Pi Foundation, offer hands-on learning experiences in computer science, nurturing students' digital skills for the future.

Hour of Code campaign: engaged 16 schools, 20 teachers, 20 facilitators, and over 1000 students in a one-hour coding event, promoting coding awareness and inclusivity across Botswana's secondary schools. We have partnered with Code.org, and this coming year, we are excited to have another Hour of Code event that will be culturally relevant and engage even more schools.

Summit impact: hosted a computer science education summit uniting 150 local and international experts, ministries, and educators in advancing computer science education. The summit influenced policymakers, advocating for reforms and resulting in two international research papers.

Curriculum advisement: worked with the Ministry of Education's curriculum department to draft a report reviewing the existing curriculum and advocating for an updated computer science curriculum meeting international standards.

Teacher empowerment: provided comprehensive training to teachers, empowering them to conduct non-formal initiatives such as Hour of Code and code clubs.

International impact: our impact on the international community is evident as we welcome an intern, Emma Dodoo, a PhD student from the University of Michigan, supervised by Professor Mark Guzdial. Emma is joining us in 2024 for a three-month internship to assist with our endeavours at CSEdBotswana. Additionally, pending venue approval, the University of Botswana will host the first ACM SIGCSE conference (COMPED) in 2025. This event will benefit not only Botswana, but also other African countries, contributing to our aim of raising awareness about computer science education in the region.

CSEdBotswana's reach and impact are enhanced through strategic partnerships with the Ministry of Education, Skills, and Development, the Ministry of Information and Communication Technologies, the Raspberry Pi Foundation, Code.org, Diamond Trading Company Botswana, Liquid Technologies Botswana, with indirect help from NSF, ECEP, and Sagefox Consulting Group, as well as local and international volunteers (trainers and facilitators). These partnerships support various activities, including facilitating training and organising non-formal interventions such as the Hour of Code and code clubs.

In summary, CSEdBotswana stands as a beacon of change, determined to usher Botswana into a future where computer science education is inclusive, dynamic, and transformative. With a resolute commitment to its aims and a robust network of partners, the initiative embarks on a journey to empower students, teachers, and communities through computer science education (**csedbotswana.org**).

SOUTH AFRICA: UNPLUGGED CODING

Rujeko Moyo interviews **Randal Rousseau** about how and why he began championing unplugged coding in libraries and schools

I met Randal Rousseau, a librarian at Bonteheuwel Library in Cape Town, during Coolest Projects South Africa 2023 in November of that year (helloworld.cc/CoolestSA2023). His passion for introducing coding concepts to young people in practical and tangible ways impressed me and I was intrigued to learn more about his remarkable journey and impact.

Why unplugged coding? What motivated you to bring unplugged activities to your library, and now to over 100 libraries in Cape Town?

RANDAL ROUSSEAU

Randal is a librarian at Bonteheuwel Library in Cape Town, South Africa. If you'd like to learn more about unplugged coding, you can contact Randal directly at **Randal.Rousseau@capetown.gov.za**.

I have always been drawn to technology and information. I recognised this gap in our education systems and curricula long ago, and was looking for something to bridge the gap and stem the fear of technology. Unplugged coding was a natural fit for the library environment, as it assumes a non-threatening experience of learning coding. As the unplugged resources and apps we use are game-based, children don't even realise they are coding because they are having so much fun. Unplugged coding requires only a mobile phone and cardboard tokens, so it's tangible. They can touch the code, manipulate it, and make it their own. In a country like South Africa that is plagued by several challenges, unplugged coding seemed a logical and sustainable means to introduce coding and robotics concepts to our young people.

What are the challenges to teaching coding in Cape Town? And how does unplugged coding address some of these challenges?

The mobile phone displays a photo image which is captured by scanning the corresponding cardboard tokens with a QR code app. Once the app confirms that the photo image matches the physical token sequence, it checks whether the coding sequence is correct or incorrect

There are various challenges. The cost of setting up and maintaining computer labs is astronomical, resulting in a lack of computing infrastructure in many schools. Frequent power cuts do not allow for the efficient operation of computer labs and computers. Schoolteachers are not equipped to teach coding, and are also overwhelmed by curriculum demands. Coding is therefore not part of the curriculum in public schools, only provided as an extracurricular activity in some. This is problematic, as coding helps to nurture vital twenty-first–century skills like computational thinking collaboration, problem-solving, and communication, to name but a few.

Unplugged coding eliminates all of these challenges. What is even more exciting is that literally anyone can learn coding concepts through unplugged coding activities, as long as they have access to a mobile phone and tokens. The apps I use are zero-rated in South Africa, which means you can visit their website without using mobile data (**helloworld.cc/tangiblgames**).

Tell us more about your unplugged sessions. What do they involve? What space or equipment do you need? What are the children learning?

A typical unplugged coding session does not take up much space. You basically need a flat surface to lay out your coding tokens (**helloworld.cc/tangibltokens**), a smartphone that has the apps on it already, and good lighting to capture the code. We make use of the whole library and the outside as well. To consolidate movement concepts, I use a temporary grid (with adhesive tape to tile the floor) in the library. This works well, as the grid patterns simulate the app layout. The first session introduces movement on the TANKS (**helloworld.cc/TANKS**) or RANGERS (**helloworld.cc/RANGERS**) app. Children physically walk out the path they think the tank or ranger needs to follow on a physical grid, and then use their coding tokens to build the relevant code. I allow them to make mistakes, as this is the fastest way to learn. Other sessions can include a game of battleships to introduce shooting concepts. I also include physical obstacle courses and treasure hunt sessions with coding movement instructions such as 'if', 'infinity/loop', or 'repeat'. I also use Lego Six Bricks (**helloworld.cc/sixbricks**) concepts to add some variety to the sessions. It's a lot of fun and the children don't even realise that they are coding!

How did you learn about unplugged coding?

In 2021, our department head of professional services and programmes invited library staff to an introductory session in unplugged coding. I immediately knew that this was my missing link to teach children coding concepts. After the initial session with representatives from Nelson Mandela University (NMU), I contacted Professor Jean Greyling, head of computing sciences at NMU, to help me understand the concepts better and explore how I could teach others (**helloworld.cc/rangerscoding**). Professor Greyling then sent a team that was based in Cape Town to help me consolidate the concepts. I received my first training kit from the University and the Leva Foundation to get me started teaching children.

Randal demonstrates and explains unplugged coding to parents at the Coolest Projects South Africa 2023 event in November 2023

Please share your journey of growing unplugged coding sessions in Capetown.

In May of 2023 I approached school principals and offered to teach unplugged coding to teaching assistants based in their schools. They agreed. I invited the assistants to the library, introduced them to unplugged coding, and then requested they implement what they'd learnt in their schools. I asked them to organise mini tournaments within a class, between classes of the same grade, and eventually among classes of different grades, in each school. The ultimate plan was to yield teams that would represent their schools in a regional Coding4Mandela tournament on Mandela Day across various regions in South Africa (**helloworld.cc/coding4mandela**).

> “ ANYONE WITH A MOBILE PHONE AND TOKENS CAN LEARN CODING CONCEPTS THROUGH UNPLUGGED CODING ACTIVITIES

Simultaneously, I started training City of Cape Town librarians and assistants at various hubs around Cape Town. I also requested that they implement the training in their own libraries. Consequently, we held various semi-final events at libraries around the city, building towards the tournament day.

In June of 2023, 33 libraries took part in a regional Mandela Day tournament in the City Hall and Central Library in Cape Town. This was a pinnacle event in the city. The winners then went on to partake in a national competition in October. Three libraries from the national competition went on to represent South Africa against 55 countries in the Coding4Mandela World Champs event on 5 December 2023 (**helloworld.cc/unpluggedworldchamp**). All three libraries finished in the top ten, taking third, sixth, and eighth positions.

What's next on your unplugged coding journey?

The vision for unplugged coding is to introduce coding concepts to children and adults in a way that is easily understood. Once this foundation has been laid, we would like them to move on from the 'unplugged environment' to software coding.

Our partnership with CoderLevelUp (**coderlevelup.org**), who are partners of the Raspberry Pi Foundation, will enable us to introduce CoderDojos (**coderdojo.com**) in our libraries, which will provide our young people and volunteer network with free access to various coding projects and languages and online learning courses.

Libraries are safe spaces that are conducive to learning and exploring, and CoderDojos require exactly that: a safe space. We hope to realise this partnership by appointing Dojo champions in every library in Cape Town. These champions will lead vibrant coding hubs and transform the hashtag Ilearntocode@mylibrary into a tangible reality.

Looking further to the future, the big dream is to have these coding and digital making experiences translate into real jobs that will change the economic circumstances of many people in South Africa. (HW)

KENYA: CULTURALLY RELEVANT PEDAGOGY IN PRACTICE

Ben Hall discusses the process of adapting computing content for Kakuma refugee camp

The problem of underrepresentation in computer science education has been widely acknowledged. To address this problem, there is a growing body of research advocating culturally relevant pedagogy (CRP) as a potential solution (see pages 14–15). So far, most of this research has been carried out in schools or colleges and has been focused on direct-to-learner activities. While schools and colleges celebrate a wide range of diversity, there are, by their nature, some things that many students there have in common. They are likely to be the same or similar ages, come from a relatively narrow geographical range, and have broadly similar educational backgrounds.

A new setting

To broaden our knowledge of the application of CRP, we analysed a vocational training course that we developed for students from Kakuma refugee camp in Kenya. The content we created was a mixture of adapted curriculum content and new resources. Students taking the course did so voluntarily, were aged between 16 and 25, and came from very different locations and backgrounds.

We also sought direct feedback from both students and facilitators. This enabled us to review the content we created, the experience of students, and the pedagogy we used.

In recent times, we have incorporated elements of CRP into many of our projects. In particular we have equipped educators with the knowledge and skills needed to review their own practice through the lens of CRP.

Over six months, we worked in partnership with Amala Education to deliver the course (**amalaeducation.org**). The project involved developing 100 hours of computing content to be delivered over a ten-week period to 16–25-year-olds. We designed the course to be vocational and to comprise 60 percent classroom learning and 40 percent independent learning.

The classroom-based learning was led by facilitators. Both learners and facilitators spoke English (the language of the course materials) as an additional language. The facilitators had a higher level of English than the learners. Throughout the initial stages of the project, we held regular meetings with the facilitators (weekly for cohort 1, monthly for cohort 2) and also held pre- and post-course focus groups with learners.

> AS OUR LEARNERS HAD LIMITED EXPERIENCE WITH DIGITAL DEVICES, WE NEEDED TO DEVELOP THEIR DIGITAL LITERACY SKILLS

Adaptations to make the content culturally relevant

The key adaptations we made were to:

- Make content more relevant to a vocational course designed for older students
- Make the language accessible to students and facilitators who spoke English as an additional language
- Use contexts and examples that were relevant to the students and the setting
- Make changes to suit the learning environment, which had no central screen and unreliable internet access
- Change the content from teacher-led to student-facing

Listening to our audience

To assess the success of the project, we conducted surveys and focus groups with both students and facilitators.

Our content development team reported that the most important considerations when making adaptations or writing new material were context and accessibility. This was, in part, informed by our experience of delivering previous projects in a variety of different settings. However, the main driving force behind this was the detailed feedback we got from students and facilitators prior to the delivery phase of the project. Both groups reported that community development and entrepreneurship were motivations for students. As a result, the content we developed was primarily focused on improving digital skills, with the ultimate goal of developing a website.

In terms of accessibility, two themes emerged: making the content accessible within the technical infrastructure at Kakuma; and ensuring that the language

Learners at Kakuma refugee camp in Kenya

and vocabulary used were appropriate for both students and facilitators.

Other themes whose importance was highlighted by the focus groups were student agency, materials, and content. In response, as content creators we aimed to make sure that, where possible, students had choice — whether that be the freedom to decide on the themes of their projects, or what role they would assume in group-based projects. The knowledge we had gained about the learning environment ensured that the course materials were appropriate for the setting they were to be used in. Finally, we sought to add extra cultural relevance to the content of the course by choosing contexts which were relevant to the learners and facilitators.

Multiple iterations

We found applying the areas of opportunity framework useful in reflecting on how principles of CRP were employed as we developed the course (**helloworld.cc/AO**). Multiple cycles of iterations were required to make the course content more culturally relevant. Using CRP enabled us to see where we hadn't made adaptations to the course content or delivery. For instance, adapting for teachers was less important to curriculum developers, due to the fact that a significant portion of the content (40 percent) was designed for self-directed learning (learning without a facilitator).

Surveying the intended participants during the early stages of the project meant that we could better adapt the content to be more accessible, particularly as the content was originally developed for an entirely different context (young learners in England). As our learners had limited prior experience with digital devices, we needed to tailor the learning to develop their digital literacy skills; this also meant we could ensure that enough time was allocated to the prerequisite skills.

The results

The adapted course was received positively by multiple cohorts, with future courses planned. Course facilitators detailed examples of learners applying their skills to create digital artefacts which help them practically on a daily basis. The vocational course is currently in its third iteration, with previous learners now both serving as course facilitators and having secured jobs in the technology industry.

Final thoughts

Using CRP in our work provided many benefits. By applying the principles to the adaptation and delivery of a vocational computing course, we have demonstrated the application of CRP in a non-formal education setting. We believe that employing CRP can promote positive outcomes for learners, but caution that these applications do require a careful consideration of the cultural contexts where they are to be applied. (HW)

BEN HALL

Ben is a learning manager at the Raspberry Pi Foundation. He is a CAS Master Teacher and a Raspberry Pi Certified Educator (**@hengehall**).

GHANA: AI EDUCATION IN HIGH SCHOOLS

How Ghana is responding to rapid advances in the world of technology

The new curriculum emphasises STEM subjects such as computing

SALOMEY AFUA ADDO

Salomey is a second-year PhD student under the supervision of Dr Sue Sentance at the department of computer science and technology at the University of Cambridge. Her research focuses on investigating how to prepare teachers to teach about AI in K–12 settings (**@SalomeyAddo1** or **linkedin.com/in/salomey-addo**).

As nations prepare for a future shaped by AI, many developing countries such as Ghana are working towards making sure they are not left out. One way in which Ghana is responding to the fast-advancing world of technology is through curriculum reform.

Ghana transitioned from an objectives-based curriculum to a new standard-based curriculum for primary and junior high schools in 2020. This new curriculum presents the cumulative body of knowledge and set of competencies that all pupils should know and be able to do, but does not dictate pedagogy.

The curriculum emphasises the study of STEM subjects, particularly computing, which has changed from being ICT-heavy to being a combination of ICT and computational thinking content. The curriculum aims to train young learners to become global citizens who are not only consumers of knowledge, but responsible creators of knowledge.

Computational thinking in the new curriculum has four strands: introduction to programming; algorithms; robotics; and AI. Across all stages in junior high school (ages 12–15), learners are expected to know the difference between artificial intelligence and human intelligence; compare the various models for training data — for example, machine learning models and deep learning models; discuss the importance of the use of AI in society; and discuss ethical issues around the use of AI technologies. While the content standards for AI in the new curriculum focus mainly on the applications and social and ethical issues around AI, we hope that teachers may be able to teach the engine and modelling aspects of

> LEARNERS WILL BE TRAINED TO BECOME GLOBAL CITIZENS

AI in schools, as enshrined in the SEAME framework (**helloworld.cc/AI_SEAME**).

One gap in the existing research is around teachers' motivation and preparedness to teach about AI as guided by the new curriculum in Ghana. Therefore, I aspire to investigate these as part of my PhD. I may also create teacher aids and resources to augment their professional development in AI, to facilitate the effective teaching and learning of AI in junior high schools across the country. (HW)

IRELAND: RECRUITING FOR CODERS

Darren Bayliss describes his experience as an IT recruitment specialist and now as Code Club programme coordinator for Ireland

You may or may not recognise me from Code Club online workshops. My role now encompasses supporting active Code Clubs across Ireland. But before I joined the Raspberry Pi Foundation, I was an IT recruitment specialist.

The problem

My time in Ireland's recruitment industry spanned more than two decades. Back then, the ongoing demand for skilled professionals in the IT sector outpaced the availability of qualified individuals to fill these roles. Despite significant efforts by major employers to promote careers in IT, a critical gap persisted: the absence of a formal computer science education programme in secondary schools throughout Ireland.

Still a long way to go

Unfortunately, it wasn't until September 2018 that the Department of Education took a step forward by introducing a new Leaving Certificate for Computer Science course in 40 schools. The first opportunity for students to take this exam was in 2020.

However, even with this formal introduction in 2018, a report from the University of Galway in March 2023 revealed that only 16 percent of schools in Ireland were offering computer science. The same report highlighted that of the 140 teachers offering computer science, only 34 were accredited by the Teaching Council. This points to a need for more confidence among educators to enhance their skills to teach the subject effectively. This accreditation barrier also hindered more schools from incorporating computer science, and is a disincentive to establishing Code Clubs within primary schools.

Unique challenges

The introduction of Code Clubs in primary and post-primary schools across Ireland faces additional challenges due to the limited after-school culture, centred around the Gaelic Athletic Association, soccer, and rugby. For the few schools offering after-school coding activities, inclusivity can be a challenge, as many students rely on public transport for commuting to and from school.

The interest is there

My Code Club journey began by participating in another youth programme from Raspberry Pi — CoderDojo. In 2013, I was managing my own business when a connection from my professional network approached me about starting a Dojo in Portlaoise, where I lived. After some consideration, I arranged an open meeting to gauge local interest. I vividly remember that first public gathering held in a Portlaoise hotel. I was expecting a modest turnout of around a dozen individuals. By the night's end, we had connected with over 140 enthusiastic parents and children. And so, CoderDojo Portlaoise was born.

Still recruiting

Now, in my six-year tenure as the Code Club programme coordinator for Ireland, the community built from that night in Portlaoise is something that I draw on every day. What I love about our community is each club's unique identity, built on a common fundamental foundation of informality and fun.

In recent years, there has been more and more interest in introducing STEM-related activities in schools across Ireland. One of the most rewarding parts of my job is interacting with educators, showing them what Code Club can offer and how easy it is to implement within their school, regardless of their comfort level or expertise, and seeing those clubs go from strength to strength. You might say I'm still an IT recruitment specialist.

If you're interested in starting a Code Club, visit **codeclub.org**. I'll see you at a Code Club online workshop soon. (HW)

DARREN BAYLISS
Darren is Code Club programme coordinator for Ireland. He loves rugby and everything around digital making, and a slice of seriously good lemon cake goes a long way with him.

FEATURE

ENGLAND'S COMPUTING CURRICULUM: WHERE ARE WE NOW?

Primary-school teachers **Sway Grantham** and **Nic Hughes** reflect on the implementation of England's computing curriculum almost ten years on

Imagine waking up one day and discovering that the subject you had been teaching for many years would no longer exist. This was the reality for many ICT teachers in England when the first draft of the new 'computing' curriculum was shared in 2013. There had been rumours and suggestions that this change was coming, but it was then that we learnt that from September 2014, it would be mandatory for 5–16-year-old children attending state schools in England to be taught computing. It seemed extreme to some and not enough to others, but ten years on, we're still here!

> RECOGNISE KNOWLEDGE GAPS AND WORK ON THEM GRADUALLY; YOU DON'T NEED TO ADDRESS THEM ALL AT ONCE

Looking back now, a lot has changed. We've learnt so much, but there are still a lot of unknowns. It is true that adoption was not automatic, and that some schools dived in, while others dipped a toe. However, we both agree that it is now more common for schools to talk about how and what they teach in computing, rather than looking the other way and avoiding eye contact. As we look back, other countries are looking forward, and beginning to make their own journey into this field. For you and for anyone who has been on this journey with us, here are some reflections, lessons learnt, and perhaps an insight into what your future might hold.

Acknowledge the change, but focus on what you already know

When the curriculum was first introduced, there was lots of talk in the media and schools about everything that had changed. This is the natural reaction when people are nervous and panicking, but when we calmed down a little, we realised there was actually a lot of content that was the same. Scratch was already being taught in many schools, as were 'control' units (the name of programming-style activities in the old curriculum). While the curriculum seemed completely new, there were lots of aspects that we already knew how to do and could keep on doing.

Similarly, vocabulary was responsible for a lot of nervousness. Algorithms, for five-year-olds! But again, when this was unpicked and what it really meant was discussed, it took some of the pressure off teachers. There would be some learning to be done — change always requires some upskilling — but the words alone were not

HELPFUL RESOURCES

These books contain a wealth of knowedge that we have gathered over the last five years on what to teach and how to teach it. They're free to download in PDF:

- Big Book of Pedagogy (**helloworld.cc/BBCP**)
- Big Book of Content (**helloworld.cc/BBCC**)

a reason to panic. Having other teachers to talk to, not just about the changes, but also about the similarities, gave a sense of security and allowed you to reflect on what could be reused.

Those early days involved many communities, such as Computing at School (**helloworld.cc/cas**), a grassroots organisation that helped computing teachers reach out to others. Many of these early meetings focused on adapting the curriculum you already had, concentrating on the similarities, and then making space for the new parts.

Gradual change

When looking for training materials and support, it's tempting to try and learn everything at once. But even if you have time for that, your brain may not have the capacity! Once you have established what you already have that can be reused and tweaked, iterate from there. For example, we took the old ICT control units and added more focus on conceptual development and the vocabulary to explain processes. We found that many of the digital literacy elements were just as important in the new curriculum as they were in the old one.

Where you do have knowledge gaps, aim to fill them one unit or subtopic at a time. Identify the gaps, know they're there, but don't be desperate to plug them all right away. We've been delivering this content for the past ten years and are still learning new things, and believe that the idea you can know everything you need to will only set you up for failure.

In contrast to this gradual change, do take a moment to reflect holistically about when the change stops. In our experience, and when working with other teachers, we found that after the first year or two, we'd spent so much time upskilling and learning how to teach the 'new' content, that we spent too much curriculum time on it. Suddenly students had so much knowledge about programming, networks, and how technology worked, but they could no longer search the internet for information or create a poster for a history project. It's very easy to go too far, and balance is key.

Have a long-term plan rather than a short-term reaction

Being ten years into a computing curriculum, we have the benefit of hindsight. This allows us to reflect on the times we did something and then had to spend time undoing it (see the point above about 'too much' change!). In the beginning we didn't spend much time thinking about the 'how'; it was all about the 'what'. We've had to learn a lot about how we teach — the pedagogy of computing. In hindsight, a more structured approach to introducing programming, considering cognitive load, and considering how to introduce key vocabulary, would have made teaching the 'what' so much easier!

Another short-term solution we saw was specialist teachers. While they are commonplace in a lot of secondary schools, with children 11–18, this was less common in primary schools (ages 5–11). However, utilising this approach in primary schools meant only upskilling one or two teachers rather than 14 or more. This works in the short term, but over time, as teachers move on, the school is once again in the situation of having no one with the skills to deliver a vital part of the curriculum. They get stuck over and over again.

If you are in this position, we implore you to consider how you will gain institutional knowledge that you can build on year after year. One technique could be curriculum teams, where several staff members share the responsibility for a subject. Therefore, when one moves on, there is still a team of people remaining to continue the progress that you have made.

It has been quite a journey so far. There have been ups and downs, steps we'd rather jump over and pretend weren't there, and others that helped us become the teachers we are today. The quality of research that now exists into new pedagogies, the gender gap, and cultural relevance, have us excited to move forwards into a more equitable computing landscape. Wherever you are in this journey, you are not alone; share what you know and be part of helping the next teacher come on board. (HW)

SWAY GRANTHAM

Sway is a senior learning manager at the Raspberry Pi Foundation, where she leads the educator development team developing resources and training for computing teachers around the world (**@SwayGrantham**).

NIC HUGHES

Nic is a specialist teacher for computing and digital learning at The Raglan Schools and an independent educational consultant. He is a massive fan of robotics and physical computing and is eager for more teachers to try these out in class (**@duck_star**).

BELGIUM: TEN YEARS OF CODERDOJO

Cindy Smits shares her experience as a volunteer coach and now as the general manager of CoderDojo Belgium

CoderDojo Belgium started over ten years ago — I think we were one of the early adopters of the programme here in Belgium. Our mission is to give each child the opportunity to discover and develop their digital and social skills for free, close to home.

Multilingual

Belgium is a country made up of three language areas (Dutch, French, and German), with the largest being the Dutch- and French-speaking areas. Computing education is only a small part of the Flemish, Dutch-speaking school curriculum. When we began CoderDojo Belgium, parents and children were already interested. The Dojo where I volunteered had a waiting list of 100 kids! So we recruited more coaches, and they opened Dojos in their local libraries and communities.

I began volunteering in the city of Mechelen. I originally went there with my son, but started volunteering because the rule was that if you were a volunteer, there was always a spot for your kid.

CINDY SMITS

Cindy is the general manager at CoderDojo Belgium and Digitale Wolven. Both nonprofits focus on digital skills for youngsters. Cindy is passionate about IT and education, and supporting girls in IT. (**linkedin.com/in/cindy-smits-43508410**)

Our geography

To support access to CoderDojos, some clubs run in Dutch, and some in French. Being a small country means that we can have language-specific clubs. We also organise events that are for the whole country: Coolest Projects, CoderDojo4Girlz, DojoCon, and a 'Thank you' day to thank our volunteers for their commitment. Our volunteers also meet regularly to acquire new coding skills and to build a community. Our geography makes it possible to do these events for the whole community, and it is really amazing to bring everyone together.

Inclusivity

We want to focus on inclusion, and we do a lot of initiatives to connect with other organisations. I am particularly passionate about getting girls involved in CoderDojo. In the early days, you'd see parents dropping off their sons but not their daughters, and I would say, "Is your daughter coming too?" The parents had already made up their minds that CoderDojo was for boys. Some of them would say, "My daughter doesn't feel comfortable in the environment." But we know that once the kids attend the Dojo, they like it, and the barrier is gone. That's why we have CoderDojo4Girlz once a year in Dutch and French — it gives girls a safe environment to try out coding.

The next generation

In Belgium, we don't really have an after-school club culture, so Dojos are usually once a month for two to three hours on a weekend. We get great volunteers, because most of them don't work on Saturdays and Sundays and they want to help in their free time. A lot, but not all, of our coaches are actual developers who are happy to give their knowledge to the next generation.

We believe that there will always be a need for CoderDojos for kids who really want to pursue computer science. At school, you get music lessons or you might sing in the choir, but if you really want to play an instrument or learn how to sing properly, you do it outside of school. That's also our vision. In school, you might learn a bit of coding, but if you really want to become a programmer, you go to CoderDojo. (HW)

BEYOND CODE AND PROGRAMMING IN **ROMANIA**

Empowering students to solve problems with product development

In Romania, computer science education has for a long time been primarily about algorithmics and programming, particularly C/C++. Since its introduction in around 1987 as a stand-alone compulsory subject called 'informatics' in high schools with STEM tracks, computer science has grown exponentially in importance in the curricula. Nowadays, the high-school (ages 15–19) informatics track is one of the most sought after, as IT companies in Romania have grown to contribute approximately 6.2 percent to GDP (gross domestic product). High-school students in the informatics track can sometimes study as many as seven hours per week of informatics under the current curricular model.

Since the 1990s and ever more quickly since the turn of the century, Romania has also introduced a compulsory class of computer and productivity tools usage in all high-school tracks under the name of 'information and communications technology', which is to be upgraded in the following years to reflect current technology and the curricular changes in middle school.

In 2017, Romania introduced 'informatics and information and communication technology', the local name that usually designates computer science and digital skills studies, as a stand-alone compulsory subject in middle school (ages 11–15), and was among the first and one of few European countries to do so.

Starting with the fifth grade (around eleven years old), students have a one-hour lesson each week in which they learn about computers and digital technology. They learn about how a computer is built, algorithmics, programming using visual tools, and how to use productivity tools such as presentation software. The lessons also include elements of basic digital skills, media literacy, digital rights, and cybersecurity. This is an ambitious take on teaching basic and advanced digital competencies, and is an enormous responsibility for the teacher, who must master a variety of content related to computer hardware, digital technology services, and programming. They need to teach these abstract concepts to young students (ages 11–15) with diverse social and economic backgrounds and unequal access to technology.

More than ever, students need to leap from programming to solving social problems with programming, and from using a digital tool to understanding how to build a better one — and this perspective change is something that can go unaddressed in the current curricular set-up. This is why we at Asociația Techsoup recommend teaching informatics from a product development perspective. We suggest including at least one experience of building a digital technology product for middle-school students. Ideally, this would be in the seventh grade (ages 13–14). We have developed Accelerator, a computer science teaching method aligned with product development methodologies used by technology companies. Our pedagogy team designed it with product development input from IT experts. Accelerator can be done in 9–11 lessons and includes lesson plans, activity sheets, materials, and teaching content. It has been piloted in more than ten Romanian schools.

Students build their product teams, brainstorm digital product ideas that would solve problems in their community, research similar or competitive products, and validate their usage hypothesis with potential real users. They also build and test wireframes. Then they build a clickable wireframe, and sometimes even code a minimal viable product. This mainly happens in high school, where students have already mastered coding. They prepare a public presentation for the product and present it to the class. In an update to be piloted later this year, students must also thoroughly consider the ethical and social consequences of the technology they build.

Students report learning more from these lessons than from regular ones, and enjoying working in teams more than the usual individual activity in informatics. Teachers are happier, too, because the method helps them to deliver all the knowledge and skills students need in computer science in one holistic experience. (HW)

ELENA COMAN

Elena, director of development, has held various executive positions with Asociația Techsoup for over twelve years. She has an academic background in pedagogy and philosophy (**elena.coman@asociatiatechsoup.ro**; **@elenacoman** on LinkedIn) .

FEATURE

FINLAND: INNOVATING AI LEARNING

Juho Kahila provides a glimpse into Finland's Generation AI project

Imagine a classroom in which fourth and seventh graders learn about artificial intelligence (AI) by designing and creating AI applications based on their own ideas. That's exactly what happened in Finland in the spring of 2023, when more than 200 Finnish primary- and secondary-school students explored AI basics and applied their knowledge to build their own AI applications. Finland is known for its high-tech expertise and has often been a leader in educational innovation. Now, in today's rapidly evolving digital landscape, the joint Generation AI project (helloworld.cc/GenerationAI) of the University of Eastern Finland, the University of Oulu, and the University of Helsinki has taken a pioneering step by embarking on an exciting journey to integrate AI education into classroom settings. This move is not just about keeping pace with technology, but about educating future citizens who are informed, creative, and ethical in their digital interactions. This article examines this journey, in which young students are taking the front seat in navigating the AI world.

Exploring AI: from basics to ethics in interactive workshops

Last spring, students from twelve schools in Eastern Finland explored AI through three hands-on workshops. In the first workshop, students were introduced to AI basics. They learnt about AI, its uses, and how it affects their everyday life, through real-life examples. They also had the opportunity to use an interactive tool called the GenAI Teachable Machine (**helloworld.cc/GenAITeachableMachine**). This intuitive tool is designed for beginners, helping them to get to grips with key concepts and workflows in machine learning.

In the second workshop, students stepped up as active creators. They got hands-on designing and implementing their own ideas and AI applications using the GenAI Teachable Machine. This phase was important in enhancing the students' understanding of machine learning's core concepts and workflows. This part of the project underscored the idea that the best way to understand something is by doing it — by experimenting, collaborating with peers, and applying ideas in real situations.

The final part of the journey was perhaps the most critical, shifting gears towards the big picture: the societal and ethical aspects of AI. The student groups showcased their AI apps and explored algorithmic bias by creating images using generative AI. They also discussed AI's societal effects based on thei r explorations. This phase was key in developing not only informed AI users, but also responsible digital citizens who understand AI's power and its impact on society.

Fourth graders (aged 10–11) explore the GenAI Teachable Machine tool using physical materials

Engagement and creativity: reflecting on the AI learning journey

The overall experiences of the project were very positive. Teachers participated in the workshop planning process and provided positive and enthusiastic feedback afterwards. Both students and teachers were highly interested and excited.

During the visits, we connected the AI topics to the children's daily lives and positioned the children as designers and creators of knowledge and tools. Connecting learning topics to their daily experiences helped them to see the relevance of AI in their own lives. We also gave open-ended tasks, which meant that they could freely design their own AI applications based on their interests. This allowed them to set personal goals and use their own ideas, making learning more engaging and meaningful for them. Additionally, it was crucial for participants to first understand AI basics through practical exercises before starting discussions about its societal and ethical impacts.

Educational technology played a significant role in the visits. The GenAI Teachable Machine was an excellent, beginner-friendly tool that offered students the freedom to complete open-ended tasks. However, the tool alone does not teach much about AI, and the role of the teachers was pivotal. They facilitated the visits, guiding students through the learning process, connecting concepts to real-world applications, and providing support for complex tasks. Moreover, teaching materials, such as worksheets, provided essential guidelines for the tasks. All these elements were needed to ensure effective, student-centred learning.

Seventh graders (aged 13–14) working with a text-to-image generative AI tool to explore algorithmic bias

> WE NEED TO EDUCATE OUR FUTURE CITIZENS TO BE INFORMED, CREATIVE, AND ETHICAL IN THEIR DIGITAL INTERACTIONS

Why this matters: shaping informed and ethical digital citizens for tomorrow

Understanding AI is essential in a future filled with digital technologies. The workshops described in this article help students learn about AI in an engaging and meaningful way, focusing not only on how AI works technically and how one can create one's own AI-based systems, but also on the impact of AI on society and ethics. Moreover, this kind of education teaches students to critically evaluate AI technologies that are already a big part of their daily lives. At the heart of our approach is the concept of a 'data agency'. In this context, it means equipping students with the will and the skills to make informed decisions and take meaningful actions in a data-driven world.

Our experience from the workshops highlights the importance of integrating technology into the curriculum in a way that is meaningful, engaging, and ethical. As AI technology becomes increasingly central to our daily lives, the need for comprehensive and ethical AI education grows. Our approach provides one blueprint for preparing the next generation to confidently navigate and shape this future.

However, our journey is just beginning, and we continue to collaborate with schools, teachers, and students. We have already begun to design next year's visits with the teachers, and have developed new educational technology for them. In our spring 2024 workshops, we will delve deeper into AI's mechanisms and its social impacts, with a special focus on social media. We're also excited to share our upcoming experiences from these school visits with Hello World readers! (HW)

JUHO KAHILA

Juho is a postdoctoral researcher at the University of Eastern Finland. Coming from a background as a youth worker and classroom teacher, he is currently involved in the Generation AI project, focusing his research on how to teach children about AI. (**fi.linkedin.com/in/juho-kahila-1a7112161**)

EUROPE: BARRIERS TO COMPUTER SCIENCE EDUCATION FOR GIRLS

New research from Google shows that the majority of girls have never studied computer science in school

Computer science is considered a fundamental skill for all young people, alongside reading, writing, and mathematics (helloworld.cc/EU). But it is only taught as a required primary-school subject in 12 of the 37 countries that are part of Eurydice, the network that explains how education systems in Europe are organised and how they work (helloworld.cc/InformaticsEU).

In 2014, Google released a report on the factors that influence young women's decisions to pursue degrees in computer science in the US, in which we identified encouragement and exposure as the leading factors influencing their choices (**helloworld.cc/GoogleUS**). In 2023, we commissioned a new study to build an updated and culturally nuanced understanding of the issues that hinder girls in Europe from pursuing computer science education.

What prevents girls in Europe from studying computer science?

Six key barriers

Based on interviews and surveys with more than 3000 students and education leaders across Europe, our report identified six key barriers impacting the participation of girls in computer science studies:

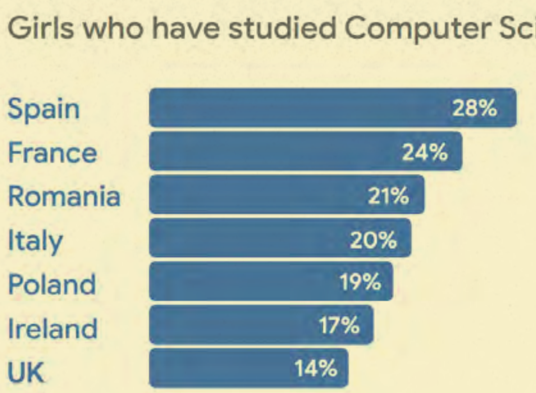
1. Computer science is often perceived as an isolated subject, rather than a skill to develop. When it's integrated with other subjects, though, 41 percent of girls are interested in studying computer science.

2. Role models are important, but it's not just about having more of them. Girls and young women need to be exposed to a fair and varied representation of relatable identities (i.e. age, gender, sexual orientation, race, attainment, and ability) throughout the education pipeline.

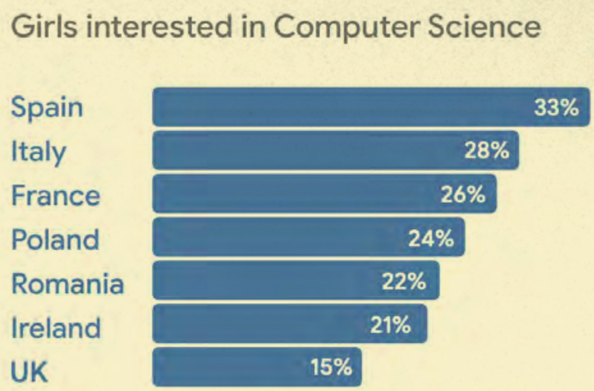
3. Teachers aren't getting the support they need to engage students. Educators often aren't equipped with the depth of knowledge, resources, or time needed to teach computer science effectively. Around two-thirds of European education systems at lower-secondary level employ teachers with specialties outside of computer science to teach the subject.

4. There's a disconnect between what students learn in school and what computer science looks like in everyday life. Students aren't aware of how computer science can be useful across other subjects and their own interests.

5. Parents, who are extremely influential when it comes to their children's success,



Participation in computer science is low across Europe, with the UK having the lowest percentage, and Spain the highest



The biggest hurdle in computer science is generating interest, as the level of participation across markets correlates with the level of interest in the subject

often struggle to support them. Many parents do not understand what computer science is and don't feel confident enough to talk with their children about it.

6. Peer networks in computer science are limited. For girls in particular, peer support is crucial in building the confidence they need in their learning. Of the students we surveyed in France and Romania, 20 percent expressed that they would feel more confident in their ability to learn computer science if more of their friends did it.

How can we address these barriers?

Address misguided perceptions: girls and young women are often unaware that computer science can be a valuable skill. Finding ways to connect technical curricula to more tangible real-world problems across other interests has the potential to engage and broaden students' understanding of computer science.

Adapt learning environments: girls' and young women's interest in computer science is significantly influenced by the learner–educator relationship. Educators providing content at the intersection of computer science, art, and the humanities can help appeal to students who are less engaged with the technical aspects of computer science.

> GIRLS AND YOUNG WOMEN NEED TO BE EXPOSED TO A FAIR AND VARIED REPRESENTATION OF RELATABLE IDENTITIES

Create computer science capital: 'bottom-up' support systems are not strong enough — parents struggle to understand the subject, and girls often lack empowerment from peers. By providing ways for students to support each other through collaborative projects, peer networks can be strengthened, while gendered perceptions of computer science can begin to be dismantled.

Google is committed to giving all students the opportunity to pursue computer science and supporting young women in becoming the tech creators and leaders of the future. We hope that the findings and recommendations in this report will raise awareness and inspire action across the education ecosystem — from teachers, parents, and caregivers to nonprofit leaders, academics, policymakers, and all others doing critical work in this space. Read the full report here (**helloworld.cc/CSBarriers**). (HW)

DR SHANIKA HOPE

The Director of Education for Social Impact at Google, Shanika Hope, is responsible for promoting equal opportunities in computing to ensure everyone benefits from its creative and economic potential. As a former educator with over 25 years of experience, Shanika understands the power of education and aims to empower a new generation of tech innovators.

FEATURE

IRAQ: ONE WOMAN'S MISSION TO BRING CODING TO CHILDREN IN IRAQ

Dr Nadia Al-Aboody describes how she is supporting the next generation of digital makers

Multiple crises, including continuous fighting, protracted displacement, economic disaster, and social and political tensions, have hampered Iraq's education sector for decades. The economic ramifications of Covid-19 coincided with a sharp decline in oil prices. Iraq is an oil-dependent economy, with oil accounting for 56 percent of GDP, 99 percent of export revenues, and 90 percent of taxes. These new challenges are putting strain on an already weakened educational system, limiting access to education and leaving 1.2 million people in need in Iraq. Many children have lost years of schooling due to conflict and severe shortages of schools and teachers. At the end of 2019, approximately 345,000 children were out of school (helloworld.cc/Iraqhno), with school dropouts continuing to be an issue.

A new approach is needed to empower the young generation with skills to have better economic independence. After Covid, the world realised the importance of learning digital skills, and more importantly, how to teach digital making and information and communication technology (ICT).

ICT education in Iraq

Until recently, computer science in Iraq was limited to universities, but now it begins at Year 1 of secondary school (ages 11–12). Public primary schools in Iraq don't offer an ICT curriculum, and although private schools do, their teaching methods are based on memorising information instead of creative thinking development, which prevents them from developing the skills they need for the workplace. Most public schools don't have computers, and others share buildings, which means one school might run in the morning from 8am to 1pm and the second school will start in the afternoon. One of the UN's recommendations is to offer alternate teacher training pathways for women from rural backgrounds. This initiative aims to boost the number of female teachers in remote areas and provide role models for girls in the labour market (**helloworld.cc/IraqEdGirls**). Moreover, it is crucial to provide young people with digital skills as youth unemployment continues to rise globally, especially in Iraqi communities. Although the technology industry is growing rapidly, there is a shortage of professional coders.

My own story

I was lucky to have parents who encouraged my interest in computing. I pursued a PhD in electrical engineering at Brunel University in England. As an international student, I wanted to find a way to be part of the local community and make the most of my time

Code Club Iraq in Maysan

Code Club is a safe space where girls and boys can grow in confidence

abroad. Through my university's volunteer department, I was introduced to Code Club and began supporting club sessions for children in the local library. I saw that the skills young people learnt weren't just technical, but also included team building and communication. That's when I realised I needed to take Code Club with me when I moved back home. Volunteering as a mentor enhanced my training skills and taught me how to guide children to learn through fun and being creative, instead of spoon-feeding them and setting exams. Engagement, creativity, and peer learning are some of the factors that made me decide to start a club in Iraq when I came back in 2018.

Bringing coding home

My home is in Maysan province (400 km southeast of Baghdad), where most families have low income or no income, so private schools are unaffordable. Maysan was one of the provinces affected by the demonstrations that took place in 2019, and was also hit severely by Covid. The majority of children of Maysan are less likely to attend school at all levels, and during the pandemic, students who lived in rural areas were without access to technology and were left behind with no proper education. Teachers did not receive training about using technology or how to conduct online classes. Almost all schools depended on using Telegram, a cloud-based instant messaging service, as a teaching platform, to the detriment of both safeguarding and online teaching.

Conquering Iraq

Before Covid, we started our first Code Club at the Amara Technical Institute in Maysan, and trained 15 female trainers to run their own clubs. During the pandemic, we were able to support our community by providing online sessions for children and club leaders. We also provided training sessions to university faculties, in partnership with the Institute of Electrical and Electronics Engineers (IEEE) youth professional Iraq branch and the Code for Iraq initiative, on how to teach online effectively. We also provided training to teachers in some schools in Maysan. This online collaboration helped us reach more teachers and volunteers who showed interest in running a Code Club in their community. After the pandemic, we lost the space where we used to do training, and had to keep moving between venues. The struggle drove us to use our family savings and take out loans to build our own centre in Maysan. Currently, all our Maysan sessions are held in our venue where we provide support to youth, children, and educators.

Female role models

We have more than 20 volunteers from different provinces, and most are female. Most of them are either from computer science or engineering backgrounds. However, we also have trainers from different backgrounds. We tailor our training to the needs of educators; for example, if they come from a background other than computer science, we offer them training in the basics of programming and how to adopt programming in their field of teaching.

When I was a little girl, I didn't find a mentor to help me with electronics; I only heard voices saying I wouldn't make it. This is why I support the growth of Code Club leaders, and is why I mentor girls and female trainers. I want to give them a chance to grow their confidence and believe in themselves. We currently support three clubs in Najaf (two of them inside the Technical Engineering College), and all-female schools in Babylon, Baghdad, Diwaniya, Wasit, Duhok, and Dhi Qar — and the next targets are Basra and Karbala. My goal is to open a club in every single school in Iraq.

*To hear more from Nadia, you can see her community story (**helloworld.cc/Nadia**) or listen to her on the Hello World podcast (**helloworld.cc/Podcast5.3**).* (HW)

NADIA AL-ABOODY

Nadia is the head of the Department of Computer Systems Technologies at the Southern Technical University in Maysan, Iraq. She is the founder of Code Club Iraq and also works with a team of volunteers to translate the free Code Club resources into Arabic (**linkedin.com/in/nadia-alaboody**).

INDIA: BROADENING ACCESS AND PARTICIPATION IN COMPUTER SCIENCE EDUCATION

Divya Joseph and **Bobby Whyte** describe RPF initiatives in India

Over the past decade, the provision of computer science (CS) education in formal K–12 educational settings has expanded massively across the globe. As computing, AI, and other digital technologies permeate our everyday lives, many have argued that access to CS education should be expanded to equip young people with the vital lifelong skills needed to thrive in the modern economy. To this end, CS has been increasingly introduced through school curricula in many countries. While some countries (for example Poland, Germany, and Israel) have included CS in their curricula for many years, others (such as England and Sweden) have more recently included CS in state curricula, or amended existing ICT (information and communication technologies) curricula to incorporate CS (for example, New Zealand). Likewise, informal CS activities have become increasingly popular in after-school clubs and libraries, through initiatives such as Code Club and CoderDojo.

Large challenges remain in access to and participation in CS education, including a shortage of qualified teachers, a lack of understanding of what should be taught and how, and a need to foster interest in CS education. In many states across India, socioeconomic challenges and a lack of technological infrastructure have prevented young people from accessing CS through formal education.

In this article, we highlight three recent initiatives to broaden access from the Raspberry Pi Foundation, and look at the importance for young people across India of participation in CS education.

Students in Udaipur learning to code

Bridging the digital divide in under-resourced communities

India's digital divide is made worse by a lack of access to electricity, the internet, and digital devices. In 2017–18, only 47 percent of Indian households received electricity for more than twelve hours a day. Only 24 percent of households had internet access, with the figure dropping as low as 15 percent in rural regions. While it is not possible for an organisation to solve these access issues single-handedly, we at the Raspberry Pi Foundation are committed to moving the needle for young people who need access to digital literacy skills the most.

Through our partnership with the Pratham Education Foundation, we have adapted the Code Club programme for learners from remote, underserved communities in Uttar Pradesh, Maharashtra, Karnataka, and Rajasthan to facilitate the learning of coding and computing in their communities. Code Club is one of the Raspberry Pi Foundation's flagship programmes, with the aim of introducing coding and computing to learners in a fun, safe environment. The informal 'club' nature of the programme allows young learners, volunteers, and educators to overcome the fear of coding and experiment using technologies.

The Raspberry Pi Foundation team based in India

Pratham provides local communities with infrastructure that includes tablets, Raspberry Pi computers connected to monitors or screens, and a Wi-Fi connection. Learners aged 13–18 use tablets and Raspberry Pis to learn to code. We support the local staff at Pratham through training that helps them to develop their skills in the basics of coding and programming pedagogy, so that they can then facilitate sessions with the young learners. During our training with the staff (young adults aged 18–25), we found that they were committed to organising Code Clubs in their communities, and they wanted to use our training as an opportunity to upskill themselves as well. As a result of their feedback, we are now adapting our training to include additional programming languages such as HTML and the basics of Python, which the Pratham staff felt would improve their chances of employability.

> THERE IS A PRESSING NEED TO PROVIDE YOUNG PEOPLE IN INDIA WITH THE OPPORTUNITY TO LEARN CODING SKILLS

Supporting non-formal CS education through partnerships

With many education systems around the world struggling to recruit and train adequate numbers of CS teachers, one challenge we would like to address is that of existing teachers' confidence and understanding in teaching CS.

Since 2019, we have partnered with Mo School Abhiyan, an initiative of the Government of Odisha, to support non-specialist high-school teachers in government schools to run Code Clubs. As many were non-specialists, we supported them through online and in-person training sessions on the basics of Scratch, a programming language suitable for novice learners; programming pedagogy; and how to operate a Code Club. We also ran live coding sessions every Friday. This allowed interested teachers to clarify their doubts. The online sessions were supplemented by the creation of WhatsApp groups for regular support. By April 2023, over 950 Code Clubs had been registered, and four hundred and forty-three (443) teachers reported that they had run Code Club sessions. From this, we estimated that 32,000 young people had taken part in at least one Code Club session.

We were able to provide tailored support to teachers to develop their confidence and basic coding skills. However, we found many other systemic challenges during the programme's implementation. Since Code Club is an informal club programme, teachers had to work with their head teachers to allocate time in their timetables. This proved challenging, as some requirements of the timetable are dictated to the teachers by the state education department. Teachers have many non-teaching responsibilities as well, and this made it difficult for them to make time.

In addition, teachers across the state continued to struggle with infrastructure issues. While the state government has mandated ten computers for each school, most of the equipment is dated and unsuitable for learning coding. This is a challenge for both students and teachers. The problem is further aggravated by intermittent electricity and internet access, making it difficult to access applications and resources online. Teachers also reported that having just one session of 40 minutes a week was found to be insufficient for students. Teachers seemed to be spending considerable time in revisions, as students were not getting enough time to practise as a result of the infrastructural issues at their schools.

To achieve a robust CS programme in schools, some of these issues must be tackled at the state level. The scale of the problems makes it quite challenging for a non-governmental entity to solve.

Adopting a school to pilot a computing curriculum

The insights from the community-led informal clubs programme at Pratham and the educator-led clubs programme at the Mo School Abhiyan helped us design our formal computing curriculum for Coding Academy, a government-led initiative in Telangana, in 2023. Our vision is to create a comprehensive computing curriculum that can be used in schools across the country.

Coding Academy was established by the Telangana Social Welfare Residential Educational Institutions Society department (TSWREIS) to provide opportunities for children from economically and socially underrepresented communities to learn coding and computing. Our partnership with TSWREIS involves creating and piloting a computing curriculum adapted to the needs of young learners in Telangana. The curriculum is delivered to the students by the Raspberry Pi Foundation's instructors, who come with degrees and backgrounds in computer science and computing.

Over the course of the five-year partnership, we aim to identify non-specialist teachers at the Coding Academy. The instructors will train and mentor them to deliver the computing curriculum. Moreover, we will work with the State Council of Educational Research and Training (SCERT) and other state departments, to introduce the computing curriculum to government schools across the state. (To find out more about TSWREIS, read the next article on pages 40–41.)

> “ OUR VISION IS TO CREATE A COMPREHENSIVE COMPUTING CURRICULUM THAT CAN BE USED ACROSS THE COUNTRY

The current and future state of CS education in India

Over 2.6 million students in India choose STEM subjects at the graduate level every year. STEM-related subjects and courses across the country have advanced and been refined over the years. However, they still leave much to be desired, especially at the school education level.

The National Curriculum Framework of 2005 (NCF 2005) highlighted the importance of learning coding and computing. It spoke of the problem-solving strategies arising from computing and emphasised the place occupied by computers in the modern world. It acknowledged that bridging the digital divide in the country had to be prioritised before young learners around the country could be encouraged to learn coding and computing.

The pandemic further highlighted the need to bridge this divide, and this has accelerated efforts in this direction by both government and private organisations. States such as Delhi, Punjab, and Pondicherry now have computers and internet in more than 90 percent of their

Through our partnership with the Pratham Education Foundation, we have adapted the Code Club programme for learners from remote, underserved communities

The Mo School Abhiyan programme supports non-specialist high-school teachers in government schools to operate Code Clubs

schools. This has created the potential to introduce a comprehensive coding and computing curriculum when ready.

NCF 2005 was also thoughtful in its approach, in that it explicitly called out the difference between ICT and CS, while also stating that both had their place in school education. To use an analogy, while ICT is learning how to drive a car, CS is understanding how a car is built. In our experience of working with education departments and teachers across the country, we found that there is a tendency to use ICT, computer literacy, coding, programming, and computing interchangeably. Most curricula across states seem to emphasise computer literacy, and some programming, for higher grades. Unfortunately, there is very little focus on other aspects of computing and computer science.

The latest NCF (2023) does not seem to build on the thoughtful vision laid by NCF 2005. Instead, it promotes technology only as a means to increase access to online resources, create localised content and potentially track learning achievements. The focus, therefore, seems more on students being competent but passive consumers of digital technologies, rather than active, innovative creators.

Developing an understanding of CS and using technologies to solve real-world problems is key to economic and social change. There are many options for those who can afford expensive private courses on coding and programming. However, for the underserved, low-resourced communities whose children study in government schools across the country, the options are more limited.

The importance of learning computing is increasingly apparent. Even the renowned physicist Stephen Hawking recognised this when he said, “Whether you want to uncover the secrets of the universe, or you just want to pursue a career in the twenty-first century, basic computer programming is an essential skill to learn.”

Apparent apathy from policymakers, and their failure to think deeply about the skills that will be required by children in the future, have resulted in a lack of opportunities for children to learn these critical skills. As demand grows for graduates qualified in computer science, AI, and related fields (for example, data science), there exists a current and pressing need to provide young people in India with opportunities to develop coding skills and to harness computing to tackle the problems of the future. Through our partnership work across India, we hope to advance this mission. (HW)

DIVYA JOSEPH

Divya heads the Formal Education programmes at the Raspberry Pi Foundation, India. She has over a decade of experience, including implementing programmes in education and developing solutions for tech companies. Divya holds a master’s degree from the University of Oxford in education and technology.

BOBBY WHYTE

Bobby is a computing education researcher and former primary-school teacher. He currently works at the Raspberry Pi Foundation as a research scientist. He holds a PhD in education from the University of Nottingham.

INDIA: CODING FUTURES IN TELANGANA

Partnering with TSWREIS to bring computing education to Telangana

At the Raspberry Pi Foundation in India, our goal is to work closely with schools, ensuring our offerings are tailored to their specific needs. We strive to design and evaluate unique learning experiences by incorporating content from our diverse range of high-quality educational products. Through these efforts, we aim to drive significant advancements in education and technology, benefitting both students and education systems across the world.

A shared mission

The Telangana Social Welfare Residential Educational Institutions Society (TSWREIS) department manages 268 residential educational institutions in Telangana. Its primary objective is to provide quality education to under-resourced young people, particularly children from India's most disadvantaged socioeconomic groups (**helloworld.cc/TSWREIS**). One of these institutions is the Coding Academy school, located in Moinabad. It operates as a fully residential co-ed school for grades 6 to 12 (ages 11–18) and accommodates around 800 students. Additionally, TSWREIS oversees another centre of excellence, the Coding Academy degree college in Shamirpet, which provides undergraduate education to 600 female students.

We joined forces with TSWREIS to form a collaborative partnership with their Coding Academy units in both the high school and college. We're committed to sharing our expertise in computing and coding curriculum with students at the school and across all courses at the college.

The curriculum aims to provide quality education to children from India's most disadvantaged socioeconomic groups

Creating a curriculum

The Computing Curriculum of the Raspberry Pi Foundation encompasses computer science, information technology, and digital literacy. These materials have been extensively researched and tested in the UK (**helloworld.cc/TCC**). Our curriculum is based on twelve pedagogical principles (**helloworld.cc/BBCP**), which ensures that we provide a holistic, project-based approach to learning. We also plan to provide national and international ways in which the Coding Academy students can showcase their learnings. For instance, we host Coolest Projects, the world's leading global technology showcase for young creators, every year.

We took on the challenge of directly delivering a comprehensive curriculum at the Coding Academy school and college through our own educators, exclusively hired and trained for this project. This was an exciting new approach for us, as we had never before delivered a curriculum directly anywhere in the world. However, we know that we have created a world-class computing curriculum for educators in both formal and informal settings, and we have many years' experience in training educators. So, we are well prepared to tackle this project head-on and make it a success.

To begin the project, our team members based in India conducted a thorough study of the Coding Academy students' interests and learning levels. Based on this, our teams in the UK and India customised and localised the curriculum content accordingly. We plan to observe the delivery of the curriculum in classrooms and collect students' feedback. Based on this data, we will refine the localised curriculum further.

Throughout the lifespan of the project, we will measure the effectiveness of our curriculum and the impact it has on students' learning. To do this, we will collect data from classroom observations, periodic assessments, and focused group discussions with students and educators.

During the second year of the project, we will work to develop the system's capacity. We will work together with TSWREIS to identify capable and interested teachers from within the organisation and provide them with the training they need. Our goal is to make TSWREIS self-sufficient in delivering computing education to students at the Coding Academy and other institutions under its jurisdiction by the fifth year of the project.

The promise of this project for our work in India

We began delivering lessons at the Coding Academy college and school in July, and it's been a rollercoaster ride so far. We've been working closely with the TSWREIS team to equip both the academic units with the resources needed for the seamless implementation of the project. Our India-based team has been hard at work to ensure continuity in the project and make sure everything is just right, and is working tirelessly to ensure this big, challenging, and exciting project blossoms and succeeds. The students' energy and enthusiasm, and the sparkle they have in their eyes for their learning, are unmatched, and everyone feels proud of their achievements so far.

This work with TSWREIS is immensely important for us, representing our dedication to shaping a brighter educational landscape, especially for young people from under-resourced communities. We hope to replicate similar initiatives across various regions in India, enabling widespread access to quality education. Eventually, we aspire to reproduce our initiatives at scale across the entirety of India. (HW)

The Coding Academy will offer national and international opportunities for students to showcase their skills

> **THE STUDENTS' ENERGY AND ENTHUSIASM, AND THE SPARKLE THEY HAVE IN THEIR EYES FOR THEIR LEARNING, ARE UNMATCHED**

MAMTA MANAKTALA

Mamta is a senior learning manager in our India team. Having been a computer science teacher for 15 years, she now creates learning resources on computing and coding for school students.

AUSTRALIA: WHAT COMPUTING LOOKS LIKE

Kaye North describes computing education in Australia, from new curriculum to confident teachers and students

Over the last ten years, there has been a dramatic change in the way that computing is taught in schools in Australia. In 2014, the Technologies curriculum was introduced to the national curriculum, the first new curriculum to be introduced since 1960. The curriculum was written so that regardless of the current technology available, the ten key concepts underpinning students' learning would remain current and relevant.

The key concepts focus on computational thinking skills and building a deep knowledge of how technology works. In Australia, the Technologies curriculum is a mandatory subject taught from Foundation to Year 8 (ages 5–14). An achievement standard (the set expectations for a student to achieve at the end of a learning year) is banded across two years of learning to ensure that a deep understanding can be built. For Years 9–12 (ages 14–18), it becomes an elective area of study.

A change in mindset

The introduction of the new curriculum brought about a change in mindset and pedagogy for teachers. It shifted the teaching from being focused on skill sets and looking at how to use a computer, to an inquiry approach that focused on using a design cycle, in which technology was used to solve a problem in a collaborative manner that may include coding, robotics, and/or data.

This change in approach to teaching computing in Australia has seen myriad responses and approaches to the curriculum. At one end of the spectrum there are teachers who have embraced and celebrated the change, delving into integrated STEM units of work with technology being a key component. Students build collaborative learning environments in Minecraft, creating digital solutions with Scratch, building prototypes with micro:bits and Arduinos, and sharing their solutions through video presentations using green screens and augmented reality.

Then there were educators who suddenly felt lost. They saw themselves as unskilled in the teaching area, and overwhelmed by new learning in a rapidly changing environment.

Building confidence

The Department of Education has worked to address these concerns and put a variety of supports in place. Teachers were offered professional development to upskill, particularly in coding, and links were drawn to the computational thinking skills that translate from maths and science to the Technologies curriculum. In the state of Queensland, an initiative was developed to support a lead teacher in each school to delve deeply into the Technologies curriculum, to then be the mentor and critical friend at their school, and to develop lessons and resources that were relevant for their context. These ambassadors for technology then formed collaborative networks to extend their own learning and continue building capacity.

This model of building networks of teachers who shared the same passions and skill set is something that has shown great benefit across Australia. Some of these teachers were employed in their school as the technology specialist teacher, so having a network of peers to collaborate with meant there was a support network for them. This model has also found success in other specialist teaching areas.

Developing students

From these collaborative networks, educators across Australia have been able to develop and implement some amazing technology units of work for students, with flexibility to adapt them to their own unique situation. An example of this is a common digital technology unit of work in which students need to demonstrate coding concepts of iteration (repetition), branching, and user input. It is quite common across Australian schools to find students developing a simple quiz in Scratch that can be adapted to suit any context, and will uniquely showcase their computational thinking skills.

For example, students can create a quiz that tests your knowledge of solids, liquids, and gases, aligning students' digital technology learning with their current science task. This activity has also been adapted in other classrooms to create a true-or-false quiz about their current book study. I have also seen it adapted to be a game to practise times tables, and a revision tool about the healthy-eating food pyramid.

Favourite resources

As Australian schools move towards being more comfortable with teaching digital technologies in the classroom, more teachers are jumping into coding for the first time and seeing it as a teaching tool across all subject areas. There are many favourite resources that teachers can access for this, and some of the most popular are coding projects by Code Club Australia and resources from the Digital Technologies Hub. Code Club Australia has been able to develop coding project tutorials that match the Australian Curriculum, and the Tasmania Quiz project is an example of this. The project tutorial can be used as a lesson plan to teach iteration, branching, and user input to students, and can also be adapted to any topic of choice.

Physical computing is also a highly popular learning opportunity within classrooms, blending digital and design technology together to build a solution to a problem. In Year 5 and 6 classrooms (ages 10–12), you can often find students using micro:bits, LEGO® MINDSTORMS® EV3s, Edison robots, or Makey Makeys to prototype a product, coded to a specific task. If we select a topic that is relevant to the students, and place their learning in context, the digital solutions that children can develop are amazing to witness.

Code Club Australia

> **EDUCATORS ACROSS AUSTRALIA HAVE DEVELOPED AND IMPLEMENTED SOME AMAZING TECHNOLOGY UNITS OF WORK**

Uniquely Australian projects

In schools that experience visits from snakes, students have used micro:bits to develop snake alarms for their kitchen gardens and playgrounds. Students have used micro:bits and Makey Makeys to develop inventions that deter crows from stealing lunchboxes from school bags before lunch break. Micro:bits have also been used to record data for analysis to make informed suggestions about changes in schools. Students may look at moisture in the soil, temperatures in playgrounds, or climate or noise levels in classrooms, and by tracking data, they can make informed proposals. These could look like shade sails in the playgrounds, timed watering systems, or devices that alert classrooms when it is too loud. I've also seen the development of alarms for pencil cases, personalised pencil holders with fans, and a scoring system for tracking noise in a classroom.

There are many great things happening in classrooms with digital technology, and we continue to ensure that digital literacy is being taught across every subject. Our young people's ability to use and engage with technology in a safe manner is imperative to their becoming active citizens and lifelong learners. (HW)

KAYE NORTH

Kaye is a passionate STEM educator based in Queensland, Australia. She currently serves as the community and engagement manager for Code Club Australia. She has taught in both primary and high schools, and has spent many years working with school leaders to map out their whole-school STEM journey. Kaye actively works to build teacher capability and support schools in building PLCs (professional learning communities) to establish a network of colleagues on similar STEM journeys. (**@KayeNorth1, linkedin.com/in/kaye-north**)

TRANSLATING EDUCATIONAL CONTENT: GUIDING PRINCIPLES

Vicente Forcada outlines the four principles that drive localisation efforts at the Raspberry Pi Foundation

English is seen by many as the language of computing, and in many countries it's also either the language of education or a language that young people aspire to learn. However, English is, in some instances, a barrier to learning: young people in many communities don't have enough knowledge of English to use it to learn about IT, or even if they do, the language of communication with other students, teachers, or volunteers may not be English.

As we are an organisation with global reach, translation and localisation have been part of the Raspberry Pi Foundation's activities from the start. Code Clubs, CoderDojos, and educational partners all over the world are helping young people learn about computing in their own language. We've already published over 1700 translated learning resources in 32 languages, thanks to the work of our talented localisation team and our amazing community of volunteer translators.

We are surrounded by learning materials in a variety of languages. In a world where browsers can instantly translate web pages and large language models can power seemingly perfect conversations in virtually any language, it's easy to assume that translation just happens, and that somehow, technology takes care of it. Unfortunately, that's not the case. Technology is certainly crucial to translation, but there's much more to it than that. In this article, I will discuss our approach to translation. It involves considering design, process, and people, to ensure that localised materials truly help young people with their learning journey.

So, how do we handle translation?

At the Raspberry Pi Foundation, we see translation as an enabler. It enables volunteers to reach learners; it enables learners to succeed in their educational goals and it enables the Foundation to achieve its mission all over the world.

With limited resources and potentially

Our 'Space Talk' project in Latin American Spanish infinite translation needs, these are the principles we follow at the Raspberry Pi Foundation to maximise the impact and reach of our translated materials.

1. Create with localisation in mind

Regardless of whether learning materials are intended for English-speaking or global audiences, it's important to create and design them with localisation in mind. That way they can be used in a variety of places, and any piece of content (text, graphics, or illustrations) can be modified to meet the needs of the target audience. We're talking here about things such as allowing enough space for text expansion, being mindful of any text embedded in graphic elements, and even making sure the context is understandable for a variety of audiences.

Making a piece of content localisable at the creation stage is virtually cost-free. Modifying fully built assets to translate them or to use them in other markets can be expensive and extremely time-consuming!

2. Always have user needs and priorities upfront

Before investing in localising or translating any materials, we seek to understand the needs and priorities of our users. In many countries where English is not the usual language of communication, materials in English are a barrier, even if some of the users have a working knowledge of English. Making materials available in local languages directly results in additional reach and enhanced learning outcomes.

> MAKING CONTENT LOCALISABLE DURING CREATION IS VIRTUALLY FREE, BUT MODIFYING ASSETS LATER CAN BE COSTLY

In other communities, where English has a certain status, a more selective approach may be more appropriate. A full translation may not be expected, but translating or adapting elements within them, such as introductions, videos, infographics, or glossaries, can help engage new learners.

3. Maximise the use of technology

While it's possible to translate with pen and paper, translation is only scalable with the pervasive use of technology. Computer-assisted translation tools, translation memories, terminology databases, machine translation, large language models, and so on are all technologies that play their part in making the translation process more efficient and scalable.

At the Foundation, we make use of a variety of translation technologies and also, crucially, work very closely with our content and development teams to integrate their tools and processes into the overall localisation workflow.

4. Take great care of the people

Even with the best technology and the smoothest integrations, there is a human element that is absolutely essential. Our amazing community of volunteers and partners work very closely with learners in their communities. They understand the needs of those learners and have a wealth of information and insights. We work with them to prioritise, translate, review and test the learning materials. They are key to ensuring that our learning materials help our users reach their learning goals.

In summary

Thinking localisation from the start, knowing what the end goals are, maximising the use of technology, and taking good care of our people and partners are the key principles that drive our translation effort.

If you'd like to find out more about translation at the Raspberry Pi Foundation or would like to contribute to the translation of our learning materials, feel free to contact us at **translation@raspberrypi.org**. (HW)

LOCALISATION OR TRANSLATION?

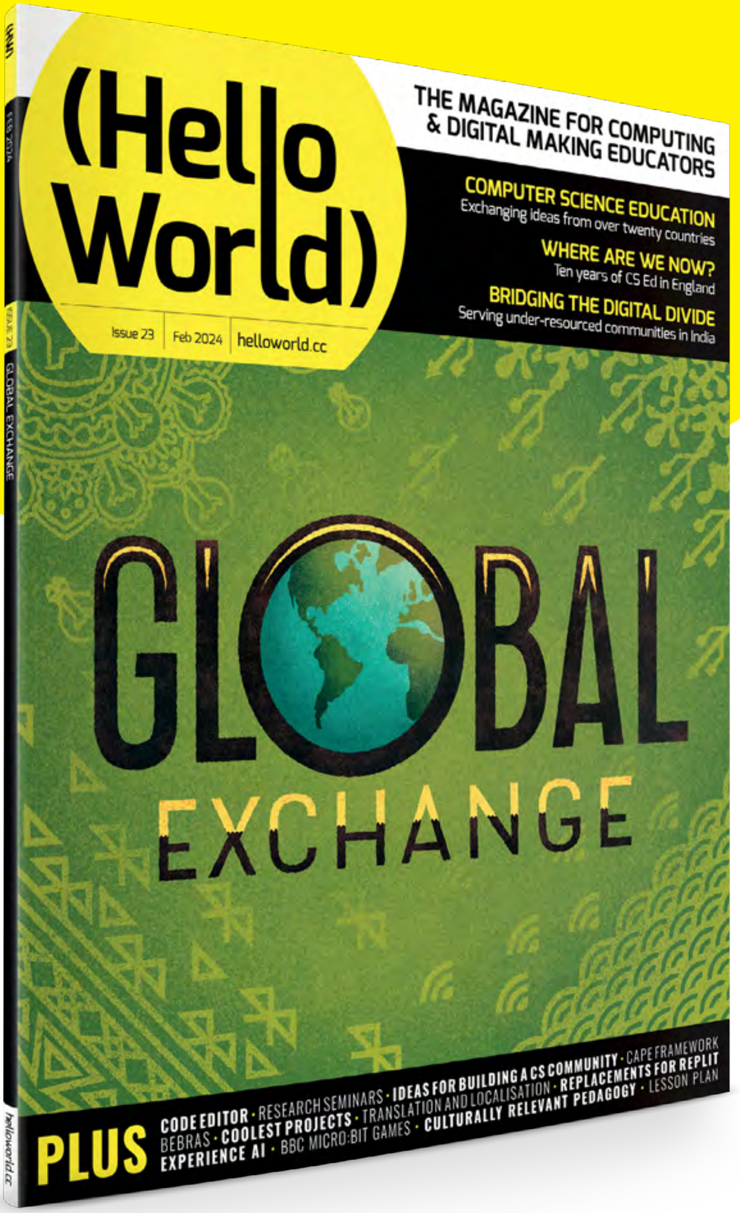
Localisation and translation are similar terms that are often used interchangeably. Localisation normally refers to adapting a product to suit a local market, whereas translation is a subset of localisation that involves changing the language of the text. For instance, localisation includes currencies, measurements, formatting dates and numbers, and contextual references. Meanwhile, translation involves only changing the language of the text, for example from English to French.

VICENTE FORCADA

Vicente is the translation manager at the Raspberry Pi Foundation. He has been in the world of localisation for over 20 years, working as an engineer, project manager, tooling specialist, and operations manager.

(HW)

SUBSCRIBE TODAY



FREE IN PRINT for UK-based educators

Why subscribe?

- Teaching resources and ideas used by over 90 percent of our readers
- Exclusive news, research findings, and in-depth features
- Delivered to your door three times a year

THE BEBRAS PUZZLE PAGE

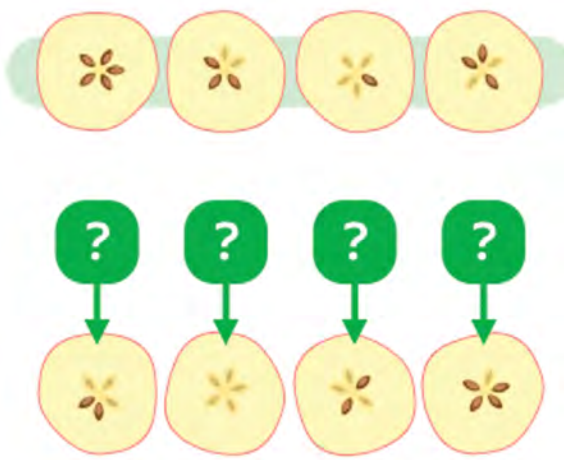
Each issue, **Andrew Csizmadia** and **Chris Roffey** will share a computational thinking problem for your students based on the work produced by the International Bebras Community

ABOUT BEBRAS

Bebras is organised in over 55 countries and aims to get students excited about computing and computational thinking. Last November, over 408,000 students participated in the UK annual challenge. Our archived questions let you create your own automarking quizzes at any time during the year. To find out more and to register your school, head to **bebras.uk**.

THE PROBLEM: **SLICED APPLES**

Pattern recognition



DOMAIN

Data structures and representation

SKILLS

Pattern recognition, abstraction, evaluation

AGE

6–12 years

DIFFICULTY RATING

Ages 6–8 hard;
ages 8–10 medium;
ages 10–12 easy

When an apple is cut in half, a star pattern is seen on both halves. This is where the seeds were. The seeds do not get cut in half. Each seed ends up in only one of the apple's two halves. Vladislava cut 4 apples in this way and put both halves on the table. However, the apple halves in the top row got mixed up.

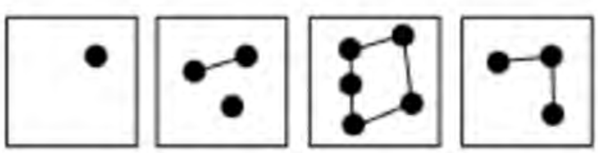
Decide which of the top apple halves match the bottom apple halves as indicated, so that the two halves of each apple are back together.

Further information

Some objects in real life must fit together; for example, pairs of A–T and C–G nucleobases in DNA. In computer science, for each password-protected file (lock) there is a password to open it (key). In programming, some functions require a set number of input parameters. It is the compiler's task to check whether the provided number of parameters in a function call and their type match the expected values by the function definition.

To solve this task, it is necessary to find out how the halves of one apple fit together. Finding a supplement and topological pattern recognition are the main skills necessary.

We can create a graph for each half of the apples in the first row. A node means a full seed bed and an edge means neighbourhoods of two full seed beds. We can also create a similar graph for the second row of halves, but this time with the empty seed beds. Then we get the same graphs for halves belonging to each other.



This Bebras puzzle was originally written by the Bebras team from Czechia, the Czech Republic. Solution on page 81.

COMPUTING KEYWORD SPOTLIGHT: **GRAPH THEORY**

Defining words and phrases in computer science

In computer science and mathematics, a group of objects and the relationships between them can be stored in a graph. These are not like line graphs or bar charts, but much more like what we might usually describe as a map.

A lot of problems presented in Bebras are simplifications of different types of graphs. This is because graph theory can be used to model a huge number of situations we find in life. Consequently, computer scientists and mathematicians have studied them extensively and come up with many algorithms for solving problems that can be described with graphs, such as finding the shortest route, a route that navigates every node, and so on. (HW)

THE BBC MICRO:BIT: ENGAGING ALL LEARNERS BY MAKING GAMES

Chris Lovell explains how making games on the micro:bit sparks interest in coding and creativity for all learners

At my school, I have found that making games is a great way to engage all learners with the subject of computing. All children are familiar with playing games in the physical world, and many have experienced games in the digital world. Learners enjoy making games their own. Visual programming tools such as Scratch enable children to be creative when designing their sprites and levels, and coding events in their games. Children add their own experiences and interests from a template into their designs. Learners naturally reflect on their games by playing and then refining them, or by handing their game to a peer to play and receiving feedback before making further changes.

CHRIS LOVELL
Chris is the head of computing at Ashfold School, UK (**@ashfoldcomp**) and a micro:bit Champion.

Making games on the micro:bit is a natural next step after Scratch. However, unlike Scratch, pupils will create a game that they can play in their hands and with others, and I have found that children love this!

micro:bit Champions

In 2023, I applied to the Micro:bit Educational Foundation to become a micro:bit Champion (**microbit.org/champions**). Micro:bit Champions get to work with educators from all over the world to share micro:bit teaching ideas and activities and to hear the latest updates from the Micro:bit Educational Foundation. Attending these global meetups inspired me to use the micro:bit to create games. As a micro:bit Champion, I wished to extend my knowledge of the micro:bit to teach coding, with Microsoft MakeCode and the Raspberry Pi Foundation's Python editor. Through my engagement with other educators on the program, I learnt that the micro:bit has a whole ecosystem of software extensions that support a wide variety of applications, such as the game blocks available in MakeCode. There is a community of makers who have created games on the micro:bit, but the games I found tended to be too complex for my target audience of learners. I decided to make games on the micro:bit with a focus on learners aged from eight to eleven, and to share them with educators like me.

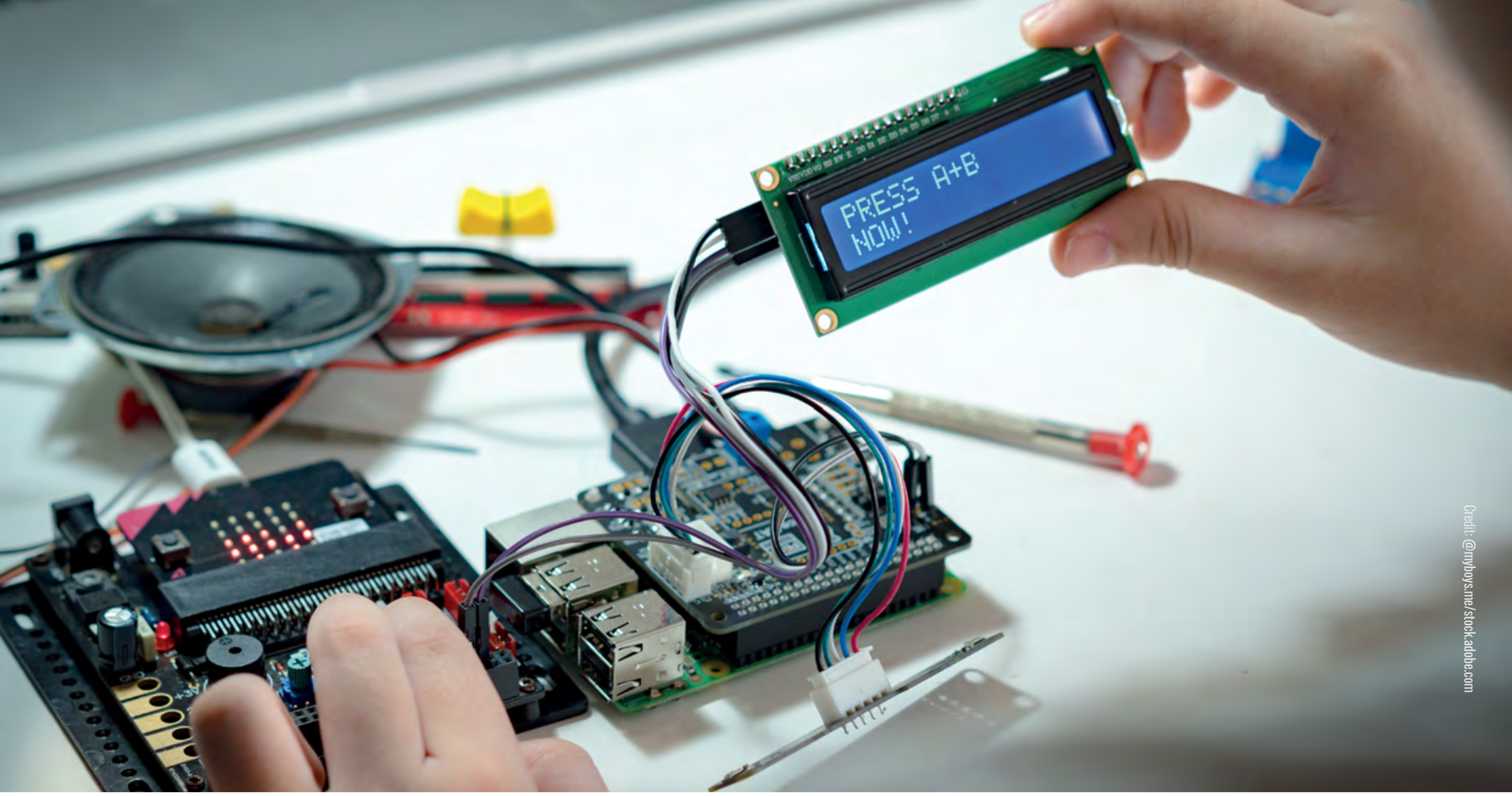
Why make games on the micro:bit?

The micro:bit is a surprisingly versatile gaming device. Its LED screen, buttons, accelerometer, sensors, speaker, and microphone make it a self-contained gaming system. The MakeCode editor game extension enables the rapid creation and manipulation of sprites through a few blocks of code. Learners can quickly realise their ideas and make a game their own through creating sound effects, intro screens, levels, and more.

I have created nearly 40 games on the micro:bit, suitable for learners aged from eight to eleven, which I use in lessons at my school. I aim to recreate classic games that children will be familiar with, such as Bat and Ball, where the micro:bit's A and B buttons control a bat sprite that moves left and right; F1, a game in which the player boosts and races ahead of competitors by shaking the micro:bit; Just Dance, in which the microbit's accelerometer and radio are used to broadcast dance moves; Flappy Bird, in which the player's bird will dodge pipes and obstacles on the micro:bit; and a rugby game in which the player has to pass a ball to a teammate before being tackled. Pupils create a game in a one-hour lesson with me. We have a lot of fun creating these games!

micro:bit in the classroom

The Micro:bit Educational Foundation has created an intuitive tool, micro:bit classroom, that enables scaffolded code to be pushed to groups of learners or individuals. In micro:bit classroom, pupils can check that their code works as expected with the inbuilt simulation and debugging tools. Teachers can save pupil work, and pupils can pick up where they left off in



Credit: @myboys.me/stock.adobe.com

next week's lesson through its classroom save feature. Teachers can also lock pupil screens during guided instruction. I find the classroom tools very useful in my efforts to ensure all learners make rapid progress with game-making in my lessons.

Teaching and learning approaches

In my classroom, learners are introduced to the micro:bit when they are seven years old, and children are quite confident with the MakeCode coding tools. When making games on the micro:bit, I explain how to make a game based on a template. For example, for Bat and Ball, I show learners how to control the bat sprite left and right; how to make the sprite move using the **loop** blocks; and how to make the ball bounce on the bat and on the wall using **selection** blocks. Once learners are confident with this template, they then add a scoring system, a game-over mode if the player misses the ball, and sound effects. I have found that once this template is understood, learners can then apply what they have learnt across a variety of games — for example, in a catching game, instead of the ball sprite bouncing on a bat, the ball sprite disappears and is regenerated at the top of the screen to be caught again. In the catching game, the **loop** blocks for a falling sprite are the same as the bat and ball game; I have found that more able learners notice this, and at this point, they begin making their own games — they may begin by adapting the sprite control user interface, before adding more challenging levels. When I see children adapting code and fixing logic errors or bugs, I am reminded of Papert's 'Hard Fun' in his *Mindstorms* book.

> THE MICRO:BIT HAS A WHOLE ECOSYSTEM OF SOFTWARE EXTENSIONS THAT SUPPORT A VARIETY OF APPLICATIONS

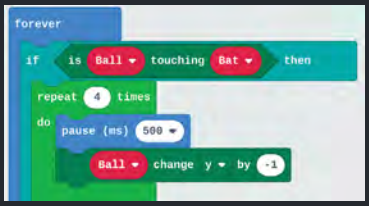
Accessing my games and guides

For many of the games that I have made, I have created teaching guides with learning objectives, step-by-step activities, code examples, and challenges for pupils. Feel free to check back every now and again, as I regularly update the folder at the following link with new games and activities (**helloworld.cc/Lovell**). (HW)

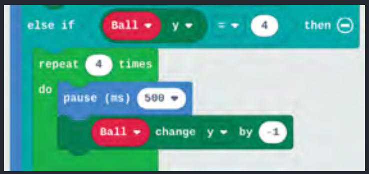
A MICRO:BIT GAME-MAKING TEMPLATE

- Control a sprite to move left and right

- Make one sprite bounce off another



- Make a sprite move from the top to the bottom of the screen



CHOOSING AN IDE THAT INSTANTLY ASSESSES STUDENT CODE

James Abela shares his search for code editing and assessment options after Replit, his preferred integrated development environment (IDE), left the market

With Replit having recently decided to leave the education market, we were forced to scramble to find a new coding editor. We had chosen Replit because it was online, allowed us to set assignments easily and, most importantly, to assess those pieces of code instantly. Their exit in the middle of an academic year created a panic internationally as teachers realised they would have to completely rethink the way they delivered coding lessons. Thankfully, other options are available, but none are simply drop-in replacements for what Replit offered.

There are many options for simply coding online. If all you need is a web-based online editor, Raspberry Pi has an excellent online Code Editor that will enable you to code in Python, HTML, simple JavaScript, and CSS (**helloworld.cc/CodeEditor**). Students under the age of 13 will need to download their code, because online saving requires a Raspberry Pi account, which is currently limited to students over 13.

For younger students, there is also Scratch (**scratch.mit.edu**), which works very well on Raspberry Pi, Linux, Windows, macOS, and Chromebooks. It is rather more limited on an iPad without the keyboard accessory. You can find lesson ideas for Scratch in Hello World issue 21 (**helloworld.cc/21**).

EduBlocks

The other main option for younger students is EduBlocks (**edublocks.org**), which was originally just known for being a simple Python-based block editor. It has now expanded to include text-based editing and supports Python 3, HTML, micro:bit, and CircuitPython.

It has a dedicated Raspberry Pi mode too, which supports a number of Raspberry Pi specialist modules, including its own version of Minecraft (a special creative edition); Sonic Pi, which enables you to make music programmatically; the gpiozero library, which is great for electronics; and Turtle.

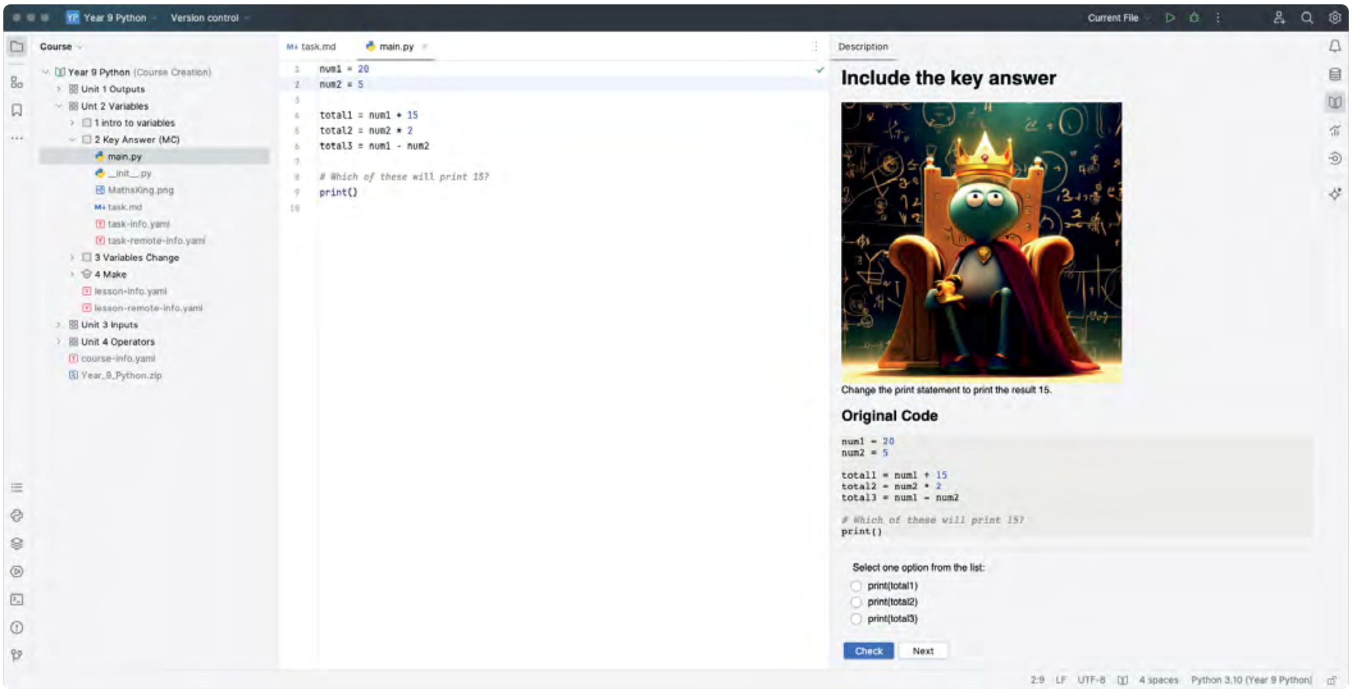
Even in standard Python 3 mode, the blocks have been chosen to include a number of very useful features from selected libraries, including Turtle, Random, Graphs, and the requests library.

The option to have block modes and text-based Python makes the transition from blocks to Python completely seamless. You can even make the initial code in blocks and then copy and paste them into a text-based project.

Just like Scratch, EduBlocks has a classroom and a very simple Google Classroom-style method for students to join it using a short code that you can show on screen or email to students. Students can then sign in with their Google or Office365 school-issued IDs, saving them from having to remember yet another password (though login by email and password is also available).

EduBlocks now also has the ability to run automated input/output tests, enabling students to gain instant feedback about whether their program runs correctly. The most tolerant mode is 'includes', which means that if students add their own spaces or words surrounding the correct output, it will still pass. Then there is 'exact match', which requires precisely the correct output. Finally, there is 'regex', which enables you to write a far greater range of validation tests, including ones with multiple correct answers. These tools make EduBlocks an excellent

The Raspberry Pi Code Editor



A multiple-choice exercise in PyCharm

option for teaching younger students, if you would like them to have a more guided coding experience rather than waiting for teacher feedback.

PyCharm

For older students, PyCharm (**helloworld.cc/PyCharm**) provides a sophisticated experience. The developer, JetBrains, specialises in making coding editors, and although this article focuses on PyCharm, the same facilities are also available in their Java editor, IntelliJ IDEA. PyCharm is one of the most popular Python editors in the world, and JetBrains offers a community edition that is quite sufficient for most users. For individual use, teachers and students alike can apply for a free licence to upgrade to the professional edition (**helloworld.cc/JetBrains**).

PyCharm runs reasonably well on the latest Raspberry Pis and is available on Linux, Windows, and macOS. Unlike EduBlocks, PyCharm processes everything on the local computer, which, once installed, means that you don't need any further internet access.

As well as being a highly competent editor, PyCharm has an entire 'Learn' area. Although these tutorials are intended for professionals, some of them are quite suitable for more able students and for teacher training. They are more verbose than those made specifically for young people, but they are useful to have.

Nor does PyCharm limit you to simply taking their courses. You can also make your own! And for those who are coming from Replit, all of the code and most of the markdown works without further alterations (see the video tutorial at **helloworld.cc/replit_pycharm**). PyCharm allows you to do a full suite of tests, input/output testing, and multiple-choice tests. This will enable your students to assess themselves comprehensively before completing more creative options.

The main limitation with PyCharm is that you will need to individually check each student's code to see the progress they are making. A workaround to this is that you can have an online work portfolio, for example in GitHub or Google Docs. Education and corporate Google accounts have an 'insert code' function that keeps all of the formatting intact.

Although Replit will be missed, both EduBlocks and PyCharm enable teachers to continue to provide a student-paced assessment with 1:1 instant feedback, and include a few new tools that will, in some ways, improve the student experience. (HW)

> EDUBLOCKS IS A GREAT OPTION FOR YOUNGER STUDENTS

JAMES ABELA

James is the head of computing at Garden International School in Kuala Lumpur, Malaysia. He is a Raspberry Pi Certified Educator, founder of South East Asian Computer Science Teachers Association, and author of *The Gamified Classroom* (**@eslweb**).

LESSON PLAN

The following lesson plan is taken from the 'Computing systems and networks — Connecting computers' unit of The Computing Curriculum (TCC), written by the Raspberry Pi Foundation. It is aimed at learners aged seven to eight, and is designed to help students understand the differences and benefits of using digital and non-digital tools.

HOW DO DIGITAL DEVICES HELP US?

Compare digital and non-digital devices, and explore the benefits of each

AGE RANGE

7–8 years

OBJECTIVES

- Explain how to use digital devices
- Recognise similarities between digital devices and non-digital tools
- Suggest differences between using digital devices and using non-digital tools

This is the third lesson in The Computing Curriculum's 'Computing systems and networks – Connecting computers' unit for learners aged seven to eight. Learners will create two pieces of work with the same focus, using digital devices to produce one piece of work, and non-digital tools to create the other. Learners will then compare and contrast the two approaches. (HW)

ACTIVITY 1: WHAT DO DIGITAL DEVICES DO? 10 MINUTES

Remind learners that digital devices are all forms of information technology and that their purpose is to help us to complete certain tasks. Use the pictures from **Figure 1** to ask learners what tasks could be completed using these devices. When they have suggested a task (e.g. playing a game), ask them if the task they have suggested could be completed on any of the other devices.

Ask learners to complete the A1 activity sheet or use **Figure 2** and ask learners to share their answers. Some tasks that can be completed on a digital device have been included, and additional space has been provided on the activity sheet for learners to add their own ideas of tasks. You can download the activity sheet here: **helloworld.cc/digitaldevices**.

Figure 1


Activity 1

Do devices have one purpose?

	Digital camera	Laptop computer	Games console	Smartphone
Take pictures				
Write a story				
Play a game				
Make a phone call				
Send a message				

Figure 2

ABOUT THE COMPUTING CURRICULUM

Raspberry Pi Foundation

The Computing Curriculum is the Raspberry Pi Foundation's bank of free lesson plans and other resources that offer educators everything they need to teach learners aged 5–16. It covers the full breadth of computing, including computing systems, programming, creating media, data and information, and societal impacts of digital technology.

Every unit of work 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**.

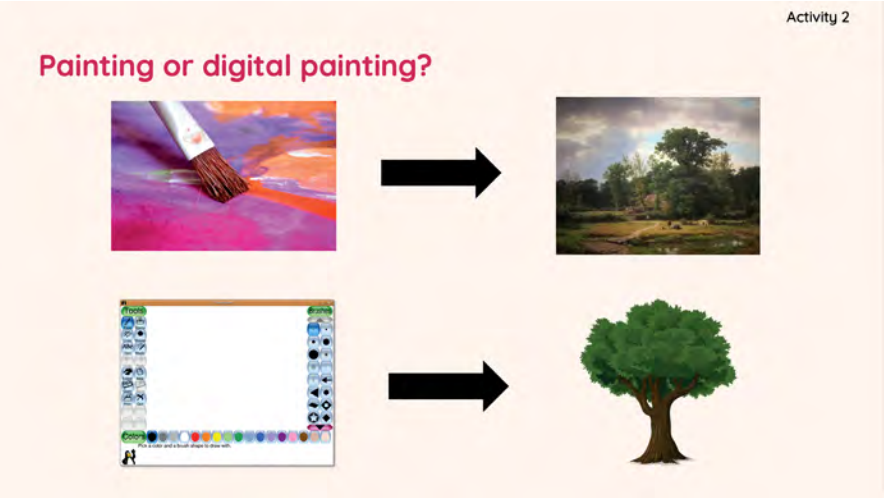


Figure 3

ACTIVITY 2: USING DIGITAL DEVICES AND NON-DIGITAL TOOLS 15 MINUTES

Explain to learners that they will be producing the same piece of art using physical materials and using a digital device. Show learners the image of a digitally drawn tree in **Figure 3**. You could encourage them to use this image as inspiration for the activity, or you could ask them to create images based on a topic that they are currently studying in class.

Non-digital activity: Explain to learners that while producing their pictures, they should consider the process of creating the artwork, e.g. how they change the colour of their brush, or how they rectify mistakes.

Digital activity: Remind learners of the main tools within the graphics program (for this activity, the 'brush', 'fill', 'undo', and 'redo' tools will be sufficient). Depending on learners' prior knowledge, you may need to give additional information about art techniques when using digital tools.

Note: *If it is not practical to undertake both approaches, learners could create a piece of work digitally and relate the experience to prior non-digital experiences, such as painting a picture. Ideally, this activity will comprise an art lesson and a computing lesson.*

ACTIVITY 3: COMPARING BOTH WORKS 10 MINUTES

Once learners have completed both pieces of work, ask them to present their images to the class. Discuss with them the processes involved in creating the images. Some example questions are:

- How did you feel about doing the activities?
- Which method did you prefer?
- Which parts of each activity did you find easier or harder?
- How did you fix mistakes in each activity?
- What input devices did you use?
- Do you think all artists would use a computer if they had access to one?
- Do you need to be good at using a computer in order to produce good art?

PLENARY ACTIVITY WRAPPING UP

5 MINUTES

Show learners the following sentences: 'I used a **paint program** on my **digital device** to **create** a picture that I can **display** in the classroom. This is different to using **paints and paintbrushes** because I can **edit my picture** by changing colours or using the undo function.'

Ask learners to produce an example sentence for a word processing program, applying the knowledge and skills they developed in the painting task.

Ask learners whether there are times when it is better to use a non-digital tool rather than a digital device. Learners should have an open discussion, for example examining how long it takes to begin or complete a task, or how easy it is to use a tool or device (fitness for purpose).

RELEVANT LINKS

TCC 'How do digital devices help us?' lesson:
helloworld.cc/digitaldevices

HOSTING A STEM EVENT

In this Insider's Guide, **Alan O'Donohoe** shares ideas on how to host a school STEM event

Picture a crisp, English morning in February 2012, the kind of weather that sends shivers down your spine before the school gates even creak open. Although it's the first day of the school holiday, at one Lancashire school, a different kind of buzz electrifies the air.

Four hundred eager faces — students, teachers, tech wizards — converge for the first Hack to the Future STEM event. What unites them? A shared spark, an infectious enthusiasm for technology's creative potential. Guest talks unveil tales of ethical hacking and exciting STEM careers, while hands-on workshops empower participants to wield soldering irons, build computers, and conjure 3D worlds with code. For one girl, the magic goes beyond the workshops. "I just like the experience," she beams. "It's a once-in-a-lifetime chance. Meeting people from industry, seeing their passion … that's what makes it different." In that moment, it's clear that Hack to the Future isn't just about gadgets and code; it's about opening doors, inspiring young people, and engaging their interest in the potential of STEM to change the world.

Fast-forward to a decade later and the impact of that first Hack to the Future continues to ripple. While we probably can't attribute any individual career choices solely to a single event, the stories of former attendees now thriving in STEM fields are heart-warming. Who knows, maybe attending the event helped encourage them to pursue future interests in STEM. Yet, the question remained: how could we bring this experience to more young minds, more often, with far less stress? Enter the one-day STEM event.

How can we share our enthusiasm for technology's creative potential with learners?

The one-day STEM event

This yearning for wider impact inspired a model for a one-day STEM event — a powerful dose of excitement, exploration and learning, distilled into a manageable school day. At Exa Networks, we're currently delivering these across specific regions as part of a contract funded by a STEM industry partner. The core mission is to demystify STEM, showcase its career and study potential, and provide hands-on experiences that inspire genuine curiosity.

If you're considering organising one, my first piece of advice is to consider the change you want to make. Be clearly focused on your event's goals from the outset. Consider the event as an intervention, shaping perceptions during and after the experience. We measure that impact through short pre- and post-event surveys completed by the young participants, to help evaluate the effectiveness of our efforts.

Are you considering organising a STEM event in your school or college? If so, the rest of this article will unpack the three main ingredients of a successful event: assemblies, STEM Ambassador engagement, and workshops. It also offers practical tips to make your event buzz with inspiration, along with some recommended resources likely to prove popular with busy teachers.

Some context

The one-day STEM events that Exa Networks has been hosting generally occur within a full school day, from 9 am to 3 pm with breaks. We've been hosting these with students aged 8–18 in UK schools and colleges, including special schools. Class sizes average around 30 students. As we typically visit unfamiliar schools, we collaborate in advance with school staff via phone calls to tailor the programme to the available resources and class sizes. Due to technology variances across schools and unreliability risks, most of our activities are unplugged or use low-tech resources.

A one-day STEM event is a powerful and immersive learning experience

Target audience

It's likely you've already selected the participants for your one-day STEM event. While you may have a specific year group or classes in mind, if the mission is to spark curiosity and open doors for all young minds, you might wish to consider including individuals outside this group who might not have as many opportunities to engage with STEM activities due to socioeconomic background or a lack of prior exposure. Throughout your event programme, you might have one group taking part in all activities, or multiple groups taking part in the same activities. We seek to include larger group assembly presentations when feasible, to reach as many young people as possible.

> YOU CAN INSPIRE STUDENTS TO SEE THEMSELVES AS FUTURE CHANGEMAKERS IN STEM

Assembly: inspiration in a presentation

It's quite a challenge to capture the imaginations of an entire school assembly in 15 minutes, especially with the logistics of getting everyone settled and dismissed.

One strategy I've found particularly effective is to highlight an example of an invention or notable person in STEM. Then I start with some simple questions to find out what they already know. The questions I use are: 'Who invented the World Wide Web?', 'Why did they invent it?', and 'What is the language of the web?' I then ask the audience to think and whisper their answers to a neighbour. This interaction helps break the ice and gets everyone engaged.

Next, I present a problem from history. Before the web, the internet existed, but remained largely inaccessible to many. Tim Berners-Lee, a shy individual, proposed a solution — the World Wide Web. To bring Tim's story to life, I perform a song I've written, set to the tune of YMCA. The entire assembly often joins in with the singing and dancing, which helps them learn about Berners-Lee's creation!

I wrap up by asking the initial questions again. This time, you can hear the answers ripple through the crowd. This reinforces the key points and demonstrates how a STEM career like Tim Berners-Lee's can solve real-world problems.

But the story doesn't end there. I highlight Berners-Lee's selfless, shy personality, emphasising that not all STEM figures shout about their achievements. This subtle message broadens the scope of what someone in STEM can be, potentially resonating with students who might not typically see themselves in those fields.

Remember, even a short assembly can pack a punch. By starting with interactive questions, weaving storytelling and music, and showcasing diverse role models, you can plant the seeds of curiosity and empower students to see themselves as future changemakers in the world of STEM.

STEM marketplace: connect young minds with STEM role models

Many young people, particularly those from underrepresented groups, rarely encounter individuals working in STEM fields. This lack of exposure to positive role models can create a perception of

PROBLEMS AND SOLUTIONS

1. How can I organise a meaningful event without any money? Explore free online resources — even low-cost activities can ignite curiosity and spark a love for STEM.

2. Where can I find the time amidst my teaching responsibilities? Start small and strategic! Focus on a one-hour assembly or a single workshop instead of aiming for a large-scale event.

3. Where can I find engaging STEM professionals? Connect with universities, science museums, or local businesses related to STEM. Consider inviting school alumni, recent graduates, or young professionals who can share their career journeys.

5. How do I convince school leaders of the value of this event? Highlight the learning objectives — showcase how the event aligns with curriculum goals and emphasises essential skills like critical thinking, communication, and problem-solving. Refer to the UK Gatsby benchmarks for objective ideas (**helloworld.cc/Gatsby**).

14. How do we plan the event without neglecting other subjects? Integrate STEM themes into existing lesson topics whenever possible. Seek cross-curricular opportunities where science, technology, engineering, and maths concepts can enrich other subjects. A well-planned STEM event can reinforce key learning objectives across the curriculum.

FEEDBACK FROM A TEACHER

"The STEM assembly was really engaging and got the day off to an exciting start! We liked the link to Tim Berners-Lee as an inspirational STEM personality and great that it highlighted he is quite a quiet and shy person to inspire more of our quiet pupils to have self belief.
The song about Tim Berners-Lee was a great way to remember facts, and this has definitely been embedded into my memory! The children were hugely engaged in the workshop sessions. They really tested their listening skills, collaboration skills (these were, I thought, particularly well-organised and communicated), as well as exciting their curiosity."

RESOURCES

- Inspiration for a school hack day: what, where, why, and how (**helloworld.cc/schoolhackday**)
- STEM Ambassadors (**helloworld.cc/ambassadors**)

STEM as inaccessible or irrelevant. The STEM marketplace activity seeks to bridge this gap by connecting young minds with real-world STEM professionals: the STEM Ambassadors.

The learning space is reimagined as a bustling marketplace. Each 'stall' showcases a passionate STEM Ambassador eager to share their expertise and ignite young minds. The young people are arranged into groups and directed around the stalls at timed intervals, to meet the STEM Ambassadors, engage in stimulating activities, and have conversations that challenge stereotypes and nurture a genuine interest in STEM careers.

Features of stall visits include:

- **'What's my line?' icebreaker:** each Ambassador uses intriguing props as clues, engaging students in a playful guessing game to unveil their STEM specialisation. Entertainment and curiosity are encouraged for deeper interaction.
- **Interactive engagement:** following the icebreaker, the real connection begins. Through open dialogue, Ambassador delve into young people's perceptions and interests, debunking myths and painting a vivid picture of the real-world applications and opportunities within their STEM fields.
- **Diversity takes centre stage:** highlighting diverse backgrounds and experiences among Ambassadors allows students to see themselves reflected in these fields, fostering inclusivity, and breaking down barriers.

The UK has an established STEM Ambassador network connecting educators with volunteers, and similar initiatives exist in other countries. To find dedicated professionals eager to support your STEM event, explore organisations within your country focused on science, technology, engineering, and mathematics education. These groups often maintain databases of volunteers willing to share their expertise and ignite young minds. Remember, early engagement is key to securing the perfect STEM ambassador for your event.

Workshops, where curiosity takes flight

After igniting imaginations in the assembly and sparking connections in the STEM Marketplace, workshops provide the perfect platform for young minds to truly delve deeper into the world of STEM. Workshops provide opportunities for young people to dive into hands-on, interactive activities, transforming theory into tangible experiences that solidify understanding and fuel a passion for exploration.

Workshops offer hands-on activities that turn theory into experience

> WORKSHOPS OFFER AN INTIMATE SPACE FOR DEEP ENGAGEMENT

Unlike assemblies and the marketplace, workshops offer an intimate space for deep engagement. This is where the magic of ‘learning by doing’ unfolds. Exa has delivered varied workshops. The young people might find themselves immersed in a developing cybersecurity simulation, their minds racing as they look for clues and thwart virtual attacks. Another workshop requires the students to use brute-force hacking techniques on combination padlocks, developing critical thinking and problem-solving skills as they unlock the secrets within. In a BBC micro:bit activity, linked to the story of Ada Lovelace and Charles Babbage, young people imagine themselves as Ada discovering the potential of a computer and consider how it could be applied to real-world problems.

Workshops not only reinforce classroom learning but also cultivate essential skills such as teamwork, communication, and creativity. However, managing multiple workshops with smooth transitions and ensuring inclusivity requires careful planning. Here are some key pointers:

- **Variety is key:** offer a diverse selection of workshops catering to different interests and ability levels.
- **Clear instructions:** provide concise instructions and ensure activities are well-structured to avoid confusion. In the cybersecurity simulation, young people need to be reminded of the task’s aim and to use logical problem-solving skills to help them.
- **Collaborative spirit:** encourage teamwork and peer support within workshops to foster inclusivity and shared learning. Our micro:bit activity has clearly defined roles for group members.
- **Differentiation:** offer alternative tasks or scaffolding within activities to cater to individual needs, interests, and backgrounds.

For more workshop suggestions and practical tips, delve into the archive of Hello World Insider Guide articles. Issue 20’s *Preparing a backup plan* on pages 84–87 offers unplugged activity suggestions (**helloworld.cc/20**), while pages 76–79 in issue 21 suggest activities for *Engaging cybersecurity lessons* (**helloworld.cc/21**). (HW)

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**).

THE ROLE OF COMMUNITY IN ADVANCING COMPUTER SCIENCE EDUCATION

Can building community unlock the potential of computer science education?

It's a fascinating time for students and educators to dive deep into the world of computer science. From brainstorming together with unplugged activities to getting hands-on with physical computing, the possibilities are endless. However, ensuring equitable access and integration of these innovative concepts and resources into educational systems is not without its challenges.

A common obstacle to embedding computer science in the classroom can be reluctance on the part of educators. This hesitation can stem from various factors, including a lack of familiarity with computer science concepts, apprehension toward new teaching methodologies, or the perceived additional workload on top of their already packed schedules. Such resistance not only impedes the incorporation of computer science into the curriculum, but also creates an environment in which educators who are willing and motivated about computer science education (CSed) may feel unsupported and isolated. This sense of isolation can be challenging for educators, who are eager to collaborate and share ideas, and the lack of connection can significantly hinder the progress of CSed as a whole. Educators inherently seek connection and belonging, especially when exploring and learning new ideas and concepts. Finding a supportive community becomes essential for those looking to drive change in CSed.

TONYA COATS
Tonya is a technology curriculum coordinator in the USA. Her role involves planning and developing the delivery of technology-based staff development. She fosters a love of technology and innovation in students. Tonya believes that embracing failure as a learning opportunity can significantly contribute to a student's growth and understanding in the field of computing. (**@teachercoats, linkedin.com/in/tonyacoats)**

Building a CSed community

Building a community of passionate computer science educators and professionals can have a transformative impact, fostering a sense of confidence and belonging in computer science education. This journey involves actively seeking out professional development opportunities, both in-person and online, as well as engaging with social networks to connect across geographical boundaries. These communities provide a platform for connection, growth, collaboration, and recognition.

In these environments, educators are not passive recipients of information; they become active contributors, sharing their unique insights and experiences. This enables all members to learn from one another. This mutual exchange can lead to peer support, access to the latest resources, and a broadened global perspective, all of which are integral to the continuous evolution of computer science education.

How community enhances CSed

Communities play a vital role in ensuring educators stay informed about the latest trends and technologies. This ensures that students receive an education that is relevant and up to date with the ever-changing world of technology. By being part of a community, educators can amplify their voices and make more significant contributions to educational discussions and policymaking. Additionally, diverse communities can aid in the development of inclusive curricula and teaching methods, catering to a wider range

CSed communities can act as a catalyst for change

of student needs and backgrounds. This creates a more equitable learning environment for all students. A united community can effectively advocate for the importance of CSed, influencing policy, resource allocation, and support at the highest levels. Within these communities, educators can also find a platform where they can share their successes and challenges, fostering a culture of continuous improvement and innovation in teaching practices. This collaborative mindset helps break down barriers, debunk myths about computer science, and inspire a new generation of educators and students alike. Ultimately, these communities act as a catalyst for change, driving forward the mission of making computer science education accessible, engaging, and inspiring for every student.

Ways to connect

Here are several ways to explore building and engaging with computer science communities:

- **Professional associations and organisations:** join educator organisations like the Computer Science Teachers Association (CSTA) or the International Society for Technology in Education (ISTE). These organisations often have local chapters and offer networking events, conferences, and workshops.

- **Online forums and social media groups:** search for groups dedicated to computer science education on LinkedIn, Facebook, Reddit, or X (formerly Twitter). Follow and engage with relevant hashtags and leaders in the field.

- **Educational conferences and workshops:** attend CSed conferences or local education technology conferences, and participate in workshops and seminars focused on CSed.

- **Online professional development courses and webinars:** enrol in online classes or webinars about CSed. These platforms often have forums or networking opportunities for participants.

- **Educational technology companies:** companies that produce educational content or tools in computer science often host community events or offer forums where educators can connect with one another.

- **Blogging and content creation:** start a blog or a YouTube channel about your experiences and insights in CSed. This can attract a community of like-minded educators and create opportunities for collaboration.

- **Mentorship programmes:** participate in or organise mentorship programmes where experienced computer science educators can guide newcomers to the field.

Together, educators can break down the walls of isolation, bridge the gaps, and empower each other and students to unlock the potential of computer science education. This collaboration fosters an inclusive environment where learning is a shared journey for anyone who is curious and eager to learn. (HW)

FEATURE

WIPSCE INSPIRATION: BELIEF SYSTEMS AS GATEKEEPERS TO COMPUTER SCIENCE EDUCATION

Matthew Wimpenny-Smith looks at whether teachers' own biases around equity in computing can be unwittingly restricting access to the subject

In September 2023, the annual Workshop in Primary and Secondary Computing Education (WiPSCE) was held in Cambridge (helloworld.cc/22, pages 67–69). At the event, hosted by the University of Cambridge Department of Computer Science and Technology and the Raspberry Pi Computing Education Research Centre, researchers and teachers were invited to discuss in a collaborative environment research in computer science education.

Inspired by a conference keynote

Joanna Goode presented research on the influential role of teachers in designing, facilitating, and advocating for equitable learning experiences in computing classrooms (**helloworld.cc/Goode**). Joanna noted that despite national efforts to develop meaningful opportunities for engagement in computer science classrooms, disparities that relate to systemic challenges still exist. She also explained the notion that computing isn't open to certain communities, and that bias belief systems act as gatekeepers to the subject. This has created many of the attitudes and inequalities that exist within the teaching and learning of computing.

Joanna advocated that teacher leadership and agency can help play a pivotal role in promoting equity and inclusion in computing education. Now that I am working as a senior lecturer in Primary Initial Teacher Education at a university in the UK, this keynote resonated with me, as I reflected as to whether I myself was one of these gatekeepers. My role is to directly support the training and development of pre- and early-career teachers, and as such, I have much influence in promoting equity and challenging belief systems, but I also risk embedding my own bias in what I pass on.

Are we training gatekeepers?

The primary national curriculum in England requires teachers to offer a balanced and broad-based education (**helloworld.cc/Englandcurriculum**), and this includes the mandatory teaching of computing from Key Stage 1 to Key Stage 4 (ages 5–16).

Quite often, the trainees on primary initial teacher training (ITT) courses are not subject specialists in computing or computer science. Many of the trainee teachers I work with haven't had computing or information and communication technology lessons since they were in secondary school. This has an impact on their attitudes towards the value of the computing subject, and on their underpinning beliefs around equity and who computing is for. The bias of the trainees may then manifest in their classroom practice as they act as gatekeepers to the subject, reflecting their lack of knowledge of the subject and their anxious feelings about teaching computing. I am also aware that many trainees will soon become early-career teachers and are likely to enter schools where their colleagues may feel the same, and hence maintain a status quo of reduced provision of computer science.

Finding out about my trainee teachers

Stimulated by my reflections following the keynote, I decided to find out more about my current trainees. I did this by administering a survey about computing attitudes, knowledge, and skills. The questions I used in the survey came from the Institute for Advancing Computing Education website (**helloworld.cc/csedresearch**), which detailed the work of Wanzer et al., including an 'attitudes towards computing' scale (**helloworld.cc/Wanzer**). I adapted this and combined the 'knowledge and skills' audit questions from the Computing at School QuickStart Computing toolkit (**helloworld.cc/CASquickstart**). The subject knowledge questions were to inform teaching of the cohort.

Within Joanna's keynote it was proposed that beliefs, values, and attributes are important characteristics for teachers' agency, and that they can work as advocates to promote equity, inclusion, and justice across the ecosystem of computing education. Therefore it's essential that I try to raise teachers' self-confidence as well as their subject knowledge, and in doing so, I hope to increase their belonging.

Reflecting on my findings

Looking at the results, it is clear that many trainees are starting from a very low point in terms of subject knowledge, and perhaps this low subject knowledge might be related to belonging. This would seem to echo the issues raised by Joanna's keynote around equity and who computing is for. However, more investigation is needed here to establish a link between the two.

Interestingly, many of the trainees indicated that they are motivated to teach the children computing. The keynote highlighted the importance of teachers serving as champions, to promote and lead computing in the school curriculum. It suggested that when teachers are given agency and leadership for championing and promoting computing in their schools, it can help advance the equity and inclusion of the subject. Perhaps once some of these novice trainee teachers are given agency, they will become advocates for computing.

I found the responses about belonging interesting, so I wanted to explore my trainee teachers' experiences of computing when they were at school. Speaking with them in my computing sessions, they often remarked on how they found computing difficult at school, and felt the teaching they had received wasn't effective or didn't resonate with them. Those who had taken computing at GCSE level found it unengaging. Conversely, despite having negative experiences, they did recognise the importance of having computing skills and knowledge.

Beliefs about who computing is for can create gatekeepers to the subject

> WHEN TEACHERS ARE GIVEN AGENCY TO PROMOTE COMPUTING, IT HELPS TO ADVANCE EQUITY AND INCLUSION

Next steps

I now plan to resurvey and interview a small group of these new early-career teachers in a focus group. I want to discuss how their motivations and attitudes towards computing may or may not have changed, and how they are acting as the next gatekeepers of the subject. Thinking about this issue has indicated to me that there is a need for awareness of the issues raised in Joanna's keynote around engagement, disparities, and bias belief systems in computing education. These are potentially present in our cohorts of primary initial teacher trainees, and once they become early-career teachers, acting as the next gatekeepers to the subject, the issues may continue to permeate into our classrooms. Without careful development of our ITT training programmes to help develop novice teachers' agency in computing, we might find that this continues. Further work is needed in this area. (HW)

MATTHEW WIMPENNY-SMITH

Matt has been teaching computing since 2004 and is currently a senior lecturer in Initial Teacher Education and Computing at Oxford Brookes University. He is a CAS Master Teacher, Oxford Primary CAS Community Leader, BCS Certified Computer Science Teacher, Raspberry Pi Certified Educator, and STEM Lead PDL (**@MWimpennyS**).

REVISITING COMPUTATIONAL THINKING

Michael Conterio and **Tracy Gardner** discuss how a return to the human element in computational thinking is needed

One of the most influential ideas in computing to emerge in the last 20 years is the concept of computational thinking, as described by Jeannette M. Wing in 2006, following on from Seymour Papert's use of the term (helloworld.cc/Wing). Thanks to this, educators have promoted the idea of key skills such as decomposition, abstraction, and algorithm design, and their uses beyond computer programming. However, as time goes on since the publication of the paper, the concept of computational thinking has often been reduced down to just these specific practices, rather than the broader "attitudes and skill set" that the paper described.

Wing's paper strongly links computational thinking to the skills that humans need for human endeavours. Beyond just thinking about a problem to be solved, computational thinking includes interrogating and investigating the types of possible solution — their advantages or disadvantages, the resources they require, and even how simple and elegant they are. The latter helps us to consider how the solutions could be duplicated, improved, or maintained by other humans who didn't build them.

Computational thinking encourages us to ask about how we can represent or model a problem in a different way in order to work with it more easily — a fundamentally human skill that asks us to use our understanding of a system in order to be able to describe that system clearly, so that we can influence it in the ways that we intend. While in the past we may have then moved on to develop a suitable algorithm or model, nowadays it may be possible to write a useful prompt or series of prompts for a large language model or other AI system. Wing says, "Computational thinking is reformulating a seemingly difficult problem into one we know how to solve, perhaps by reduction, embedding, transformation, or simulation." Large language models (such as ChatGPT or Bard) have added 'writing a well-defined prompt' to this toolbox.

THE HUMAN ELEMENT

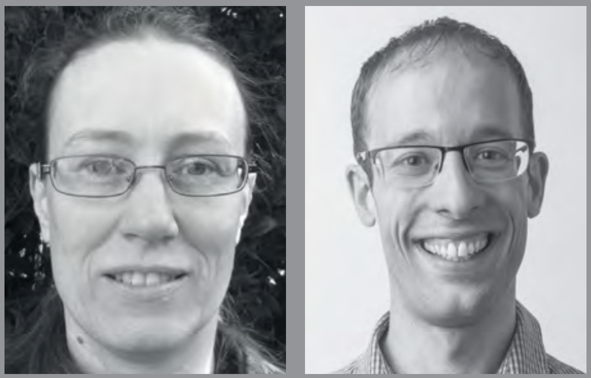
"We humans make computers exciting. Equipped with computing devices, we use our cleverness to tackle problems we would not dare take on before the age of computing and build systems with functionality limited only by our imaginations."

– Jeannette M. Wing

Humans also have a key role in making sure that we are training the machine learning system in the right way for the solutions that we want, whether that means carefully picking and planning our training data, or considering the algorithms that train these machine learning models. It's all part of computational thinking.

Computational thinking also involves thinking about how humans are involved in solving the problems, working in conjunction with computers. Although AI systems may be used in the future to control many other systems, there's an obvious argument for keeping the involvement of humans not just at the start, but also during and at the end of AI-driven processes.

Wing says, "Computational methods and models give us the courage to solve problems and design systems that no one of us would be capable of tackling alone." This is even more relevant today, and as educators, we need to make sure we are teaching the breadth of computational thinking in a way that empowers our learners. (HW)



TRACY GARDNER & MICHAEL CONTERIO
Tracy is a computer scientist, tech industry professional, and technology educator, and a co-founder of Flip Computing. Michael is a former physicist and now works as an online course production manager at the Raspberry Pi Foundation.

ATLAS OF AI

A new book examines the hidden costs of artificial intelligence — from natural resources and labour to privacy, equality, and freedom

INFO **BY** Kate Crawford | **PUBLISHER** Yale University Press | **PRICE** $19.95 | **ISBN** 9780300264630 | **URL** helloworld.cc/atlasofai

Divya Joseph

Kate Crawford's new book urges us to question the artificial nature, supposed intelligence, ubiquity, and promiscuity of AI — specifically, its tendency to be able to do everything, while seeming to have come from nothing.

Where did AI come from?

AI is anything but new, but right now, it feels as though everything around us is about AI. It is important that we examine it with a critical lens and understand how this technology impacts our lives and will continue to impact them in the future.

There have been a lot of conversations around potential job losses because of AI, but very little has been said about how it came to be, how it is being shaped, by whom, and what could become of it tomorrow as a result of who holds the power to shape it.

Kate Crawford's book presents a collection of maps on various topics that may seem unconnected at first, but help the reader to develop a comprehensive view of AI. The main argument the author makes is that AI is neither artificial nor intelligent. It is built using natural (the opposite of artificial), social, and political resources, and it has been given structures and frameworks to process in a way that seems deceptively human-like. As a result, AI trivialises the magnificence and sophistication of human intelligence.

The black box of AI

In chapter after chapter, the author attacks the notion that AI is disembodied intelligence. In Chapter 1, she talks about the extractive nature of AI and how the process of creating models for AI is paid for in minerals, oil, forests, wildlife, and other naturally existing resources. The true cost is never borne by the tech industry that propagates AI. In Chapter 2, the author demonstrates how AI is actually assembled by cheap human labour, and not by machines. Chapters 3 and 4 look at data — whose data is being used, from where, and by whom — to collect and classify what it means to look and be human, and the implications of those decisions. Finally, in the last chapters, the author states that at the end of the day, AI is a tool of state power.

The author argues that it is by design that AI has an aura of being too complex and too difficult to be understood by the hoi polloi. It is either seen as the solution to every problem, or something that is to be feared. However, the author urges that it is because of this very opposing view that one should aim to interrogate what's hidden inside the black box of AI. (HW)

BEBRAS PUZZLE SOLUTION (PAGE 65)

The number of all the seeds in one apple must be five, but counting them is not enough.

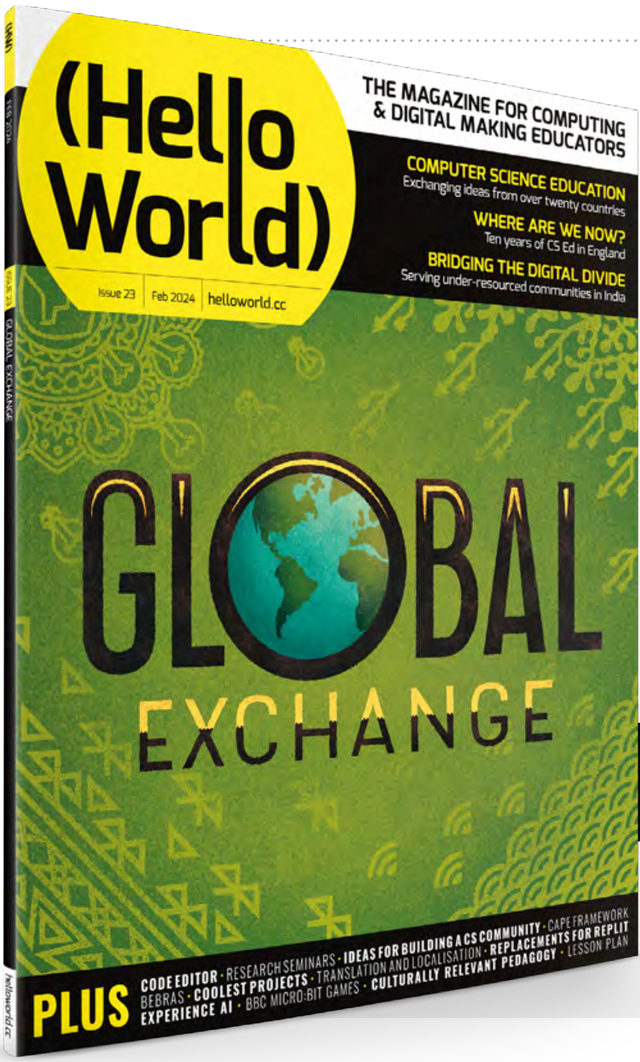
Two apples with an equal number of seeds in one half can be distinguished according to the pattern of placement of full and empty seed beds. The halves on the left in the picture have all full seed beds placed together, without an empty bed between them. The halves on the right have one full seed bed isolated, with empty beds on both sides.

"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 Hello World is the official magazine of the Raspberry Pi Foundation.

Q WHY DO 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 magazine is available three times per year. Check out our podcast (**helloworld.cc/podcast**) and monthly newsletter (**helloworld.cc/newsletter**) to get more great content between issues.

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

fb.com/HelloWorldEduMag

NEXT ISSUE OUT
JUNE 2024

THEME: **IMPACT OF TECHNOLOGY**

(Hello World)

helloworld.cc

Raspberry Pi Foundation