Computing in Special Educational Needs & Disabilities Settings: the Current Picture in England

Catherine Elliott

- Sheffield eLearning Service -

The Problem

- Recent surveys* have investigated the picture of computing in primary and secondary schools, but very little research has been conducted into what is happening in special schools and similar settings.

- As of January 2017 14.4% of pupils in England have special educational needs.

- Over a million of these young people are currently educated in mainstream settings.

**Sentance, S. & Csizmadia, A. (2015). Teachers' perspectives on successful strategies for teaching Computing in school / Royal Society (2017). After the reboot: computing education in UK schools*

Aims

To find out:

- What is being taught?
- Teacher confidence and knowledge
- Barriers
- Relevance
- Useful strategies and resources

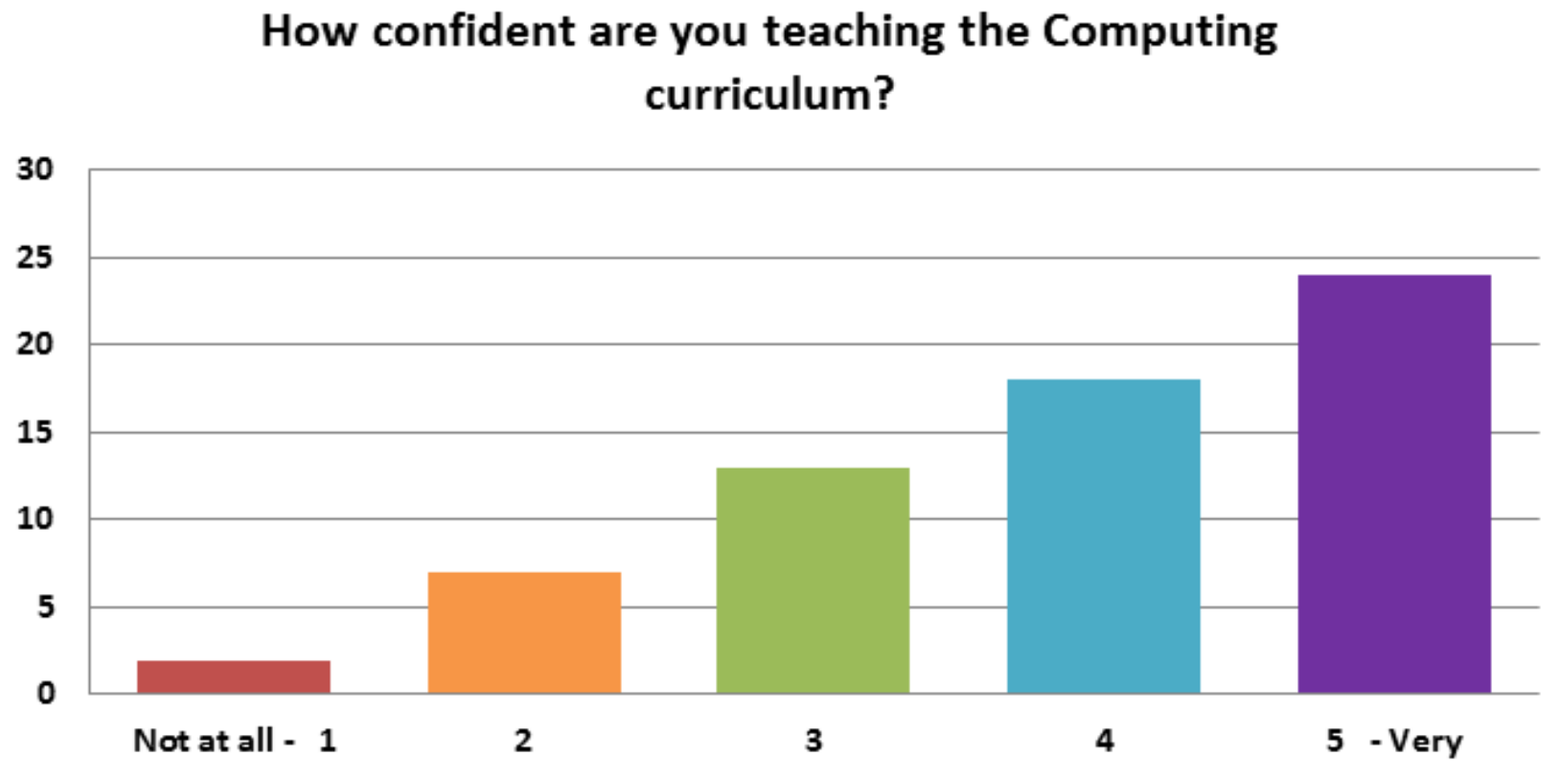
Methodology

A survey was distributed via email, social media and the Computing at School website in Autumn 2017, aimed at computing or ICT teachers in special schools and similar settings in England.

The results were analysed and key theme identified. Follow up interviews were conducted with selected respondents with a range of experience and views.

Participants

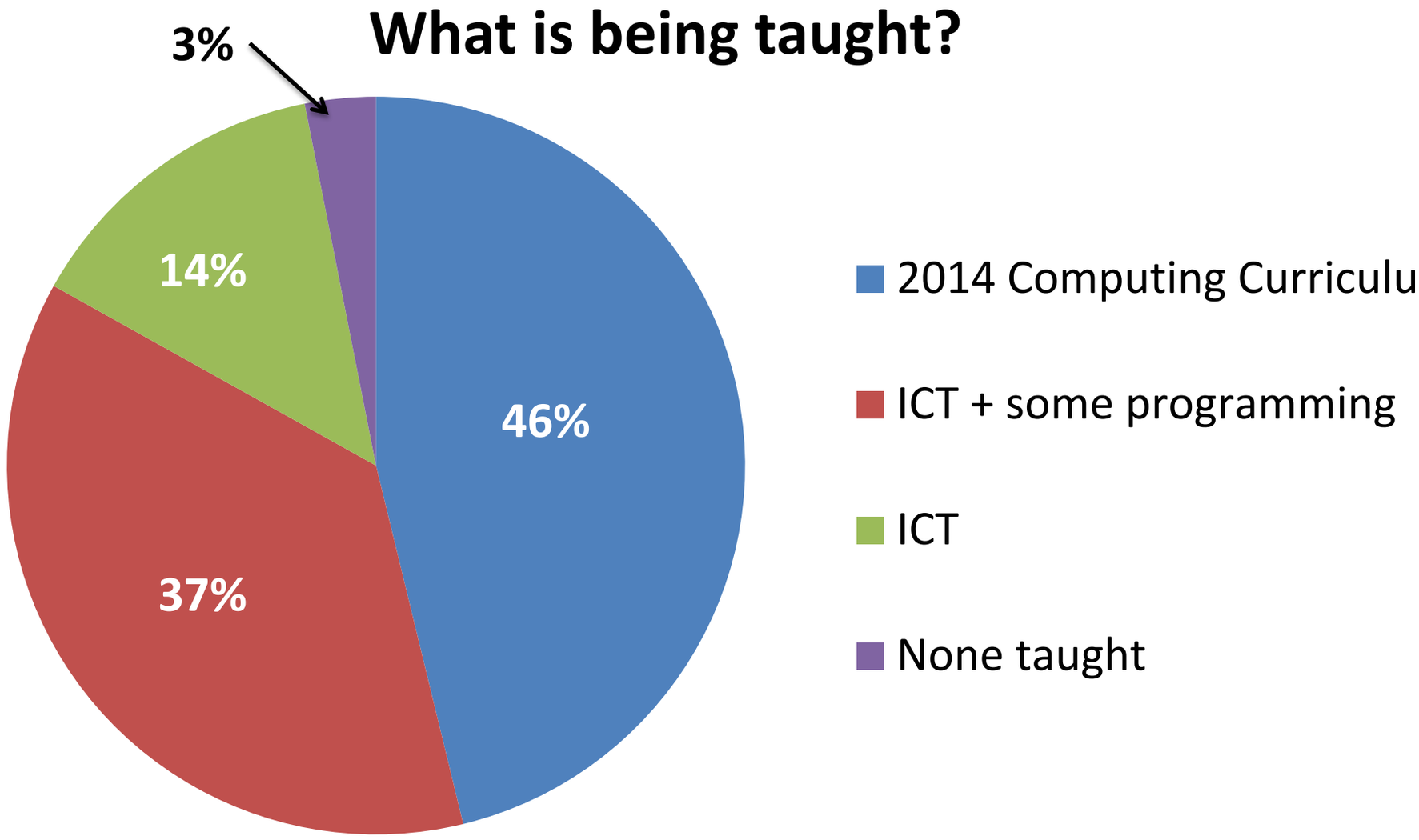
There were 65 respondents to the survey, representing approximately 6% of special schools in the country. Of these 46 were the Computing or ICT lead in the school.



"I've been teaching computer science for a number of years, have a degree in the subject and have worked in industry both as a programmer and in network management"

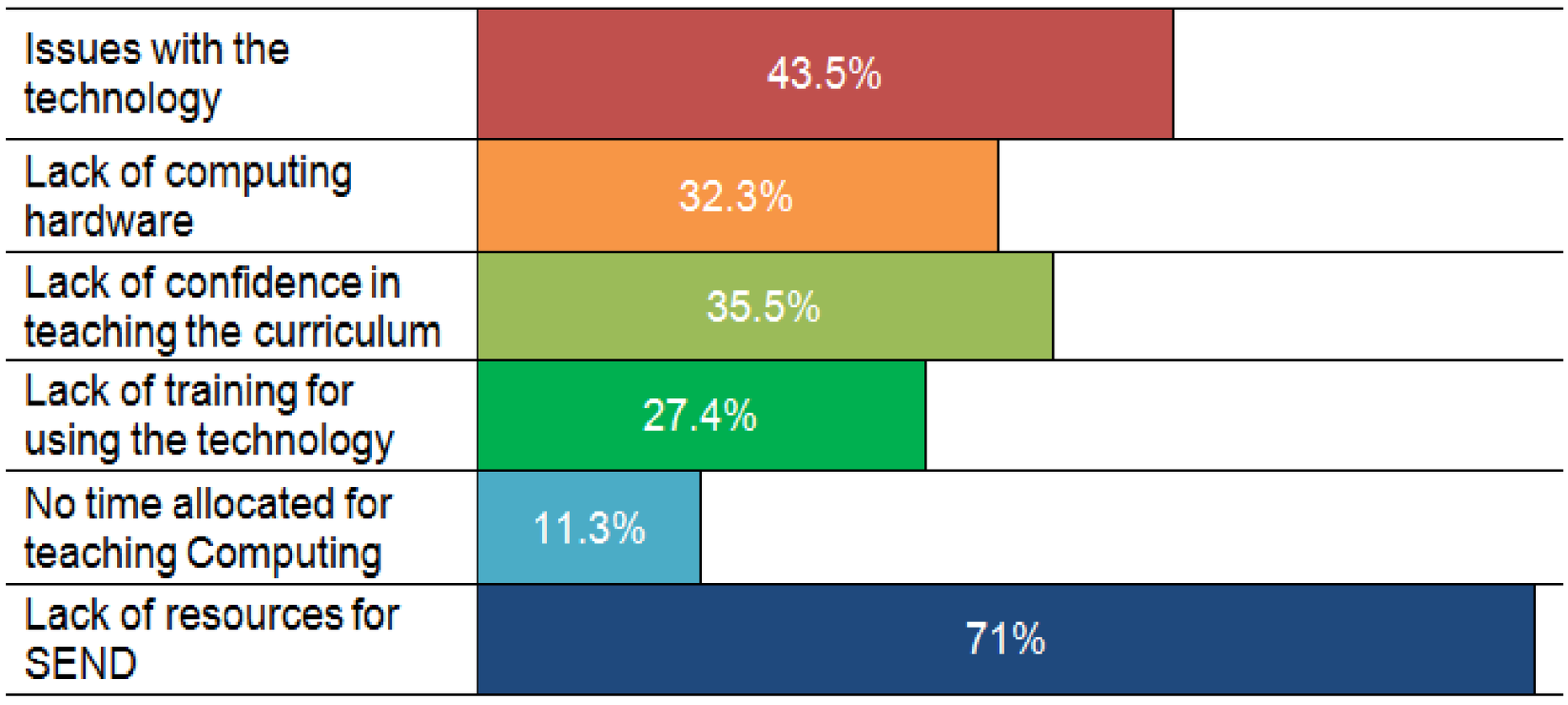
"Not as confident with some of the newer programming"

Curriculum


What is being taught?
3%
14%
46%
37%
2014 Computing Curriculum
ICT + some programming
ICT
None taught

Barriers

Respondents were asked which, if any, of the following they had found was a barrier to teaching Computing in their school:



"Not a lot of resources that meet the needs of SEN pupils. All resources are targeted at mainstream pupils."

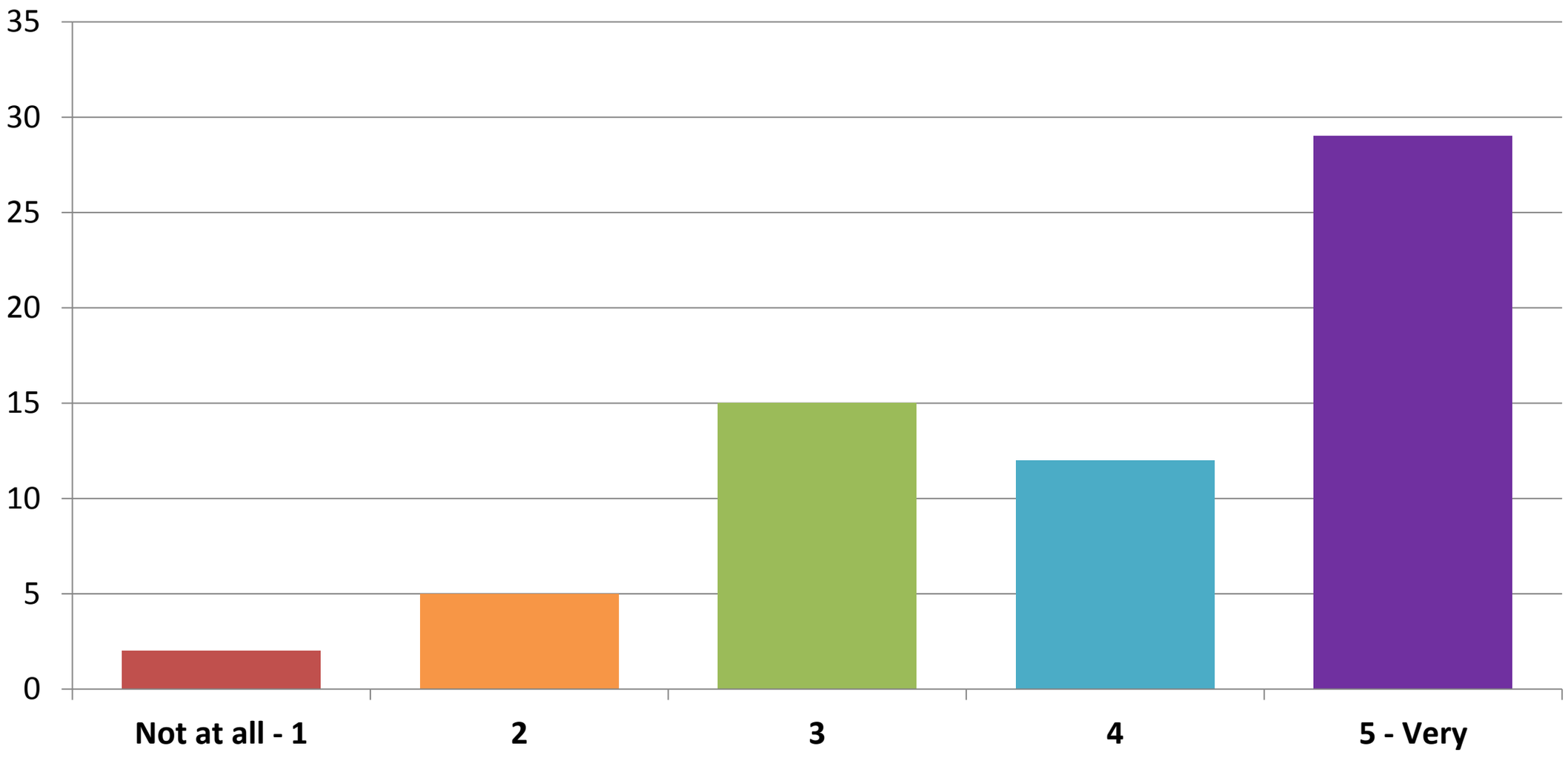
Relevance

“You can imagine… ‘Listen to the computer beeping… Look at the numbers on the screen… Smell that data… taste that algorithm!’”

(Ian Bean, What do algorithms taste like?)

Relevance

Respondents were asked how relevant they considered the subject of Computing/ICT as a whole is to their students:



Relevance

Computing/ICT:

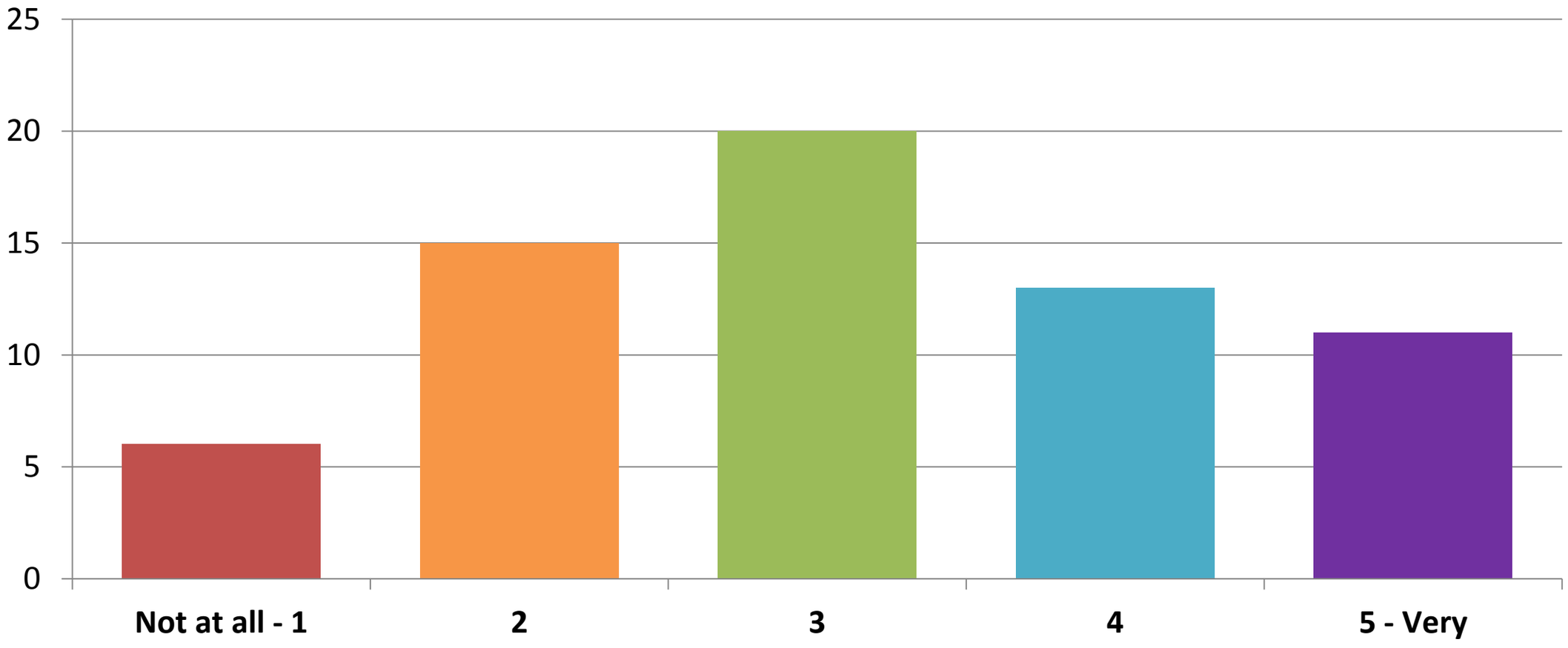
- Technology is **all around us** and enables pupils to become more **independent** and improve **problem-solving** skills (20)
- ICT/Computing can help improve **literacy** and aid **communication** (10)
- Technology enables **access to the rest of the curriculum** (4)
- It is engaging and an area where pupils can **excel** (6)
- ICT skills can help pupils in their **future careers** (6)
- The subject can help students **stay safe online** and be more **digitally literate** (3)

"Many of our students are very engaged by ICT. It can be incredibly motivational for ASD and SEMH learners. It empowers students who struggle with social interaction to present and share their work widely."

"Computing should be about accessing technology not about gaining computer or programming skills"

Relevance

Respondents were asked how relevant they considered the Computer Science (e.g. the programming and computational thinking) elements of the curriculum are:



62.5% of the teachers who rated themselves as very confident the computer science elements to be relevant (4 or 5).

Relevance

Computational Thinking & Programming:

- Computational thinking and programming can help improve **problem-solving, logical thinking** and link to **life skills** (15)
- Meaningful links can be made to other curriculum areas, including **maths and literacy** (8)
- Computational thinking and programming is **engaging** and an area where some SEND pupils can **excel** (6)
- This part of the curriculum provide opportunities for **creativity** (2)

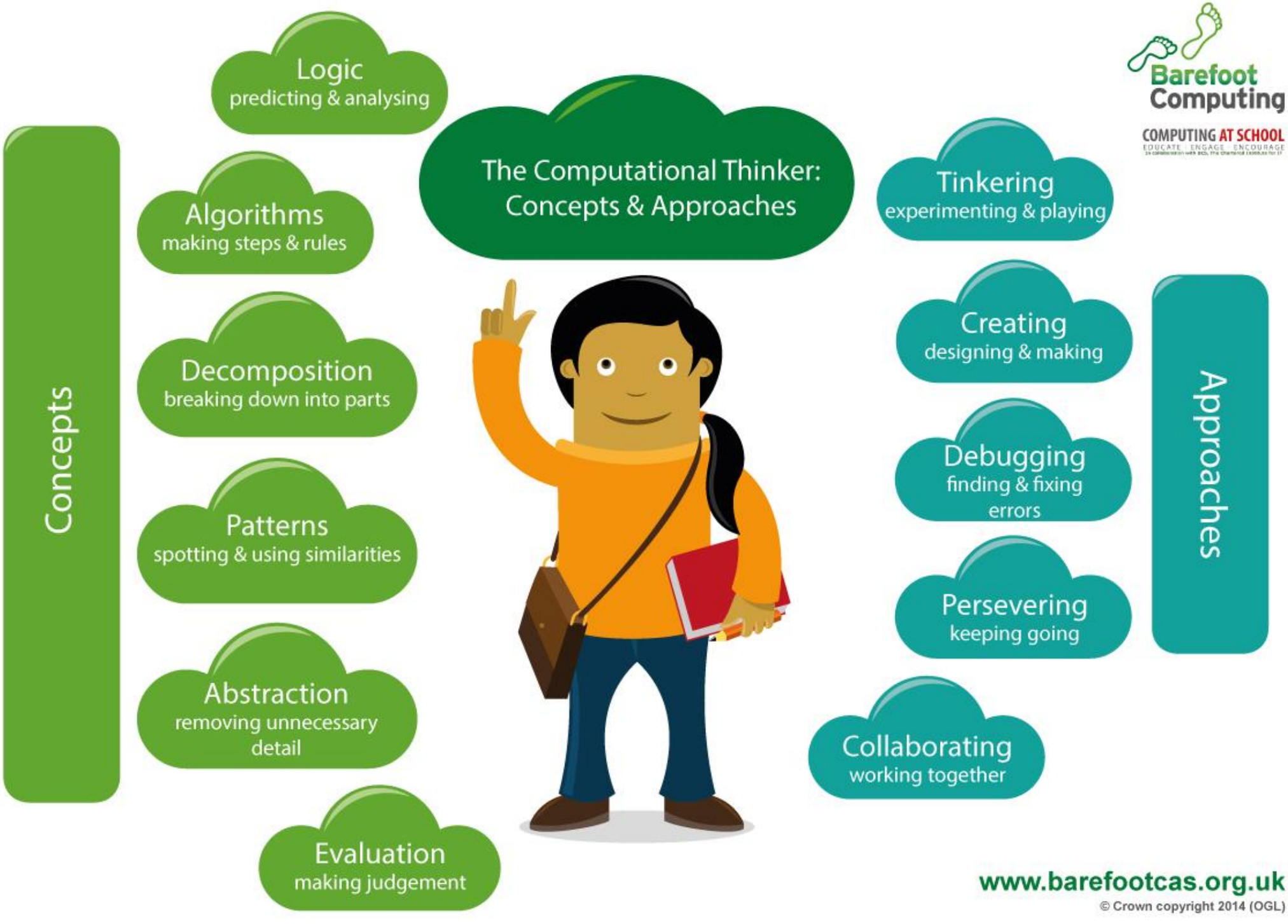
"Students are generally incredibly motivated by computing science. It allows them to apply literacy and numeracy skills in creative ways. They are proud of their achievements."

Computational Thinking

"Computational thinking is a fundamental skill for everyone, not just for computer scientists. To reading, writing, and arithmetic, we should add computational thinking to every child's analytical ability."

Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33.

Computational Thinking



www.barefootcas.org.uk

Barefoot would like to acknowledge the work of Julia Briggs and the eLIM team at Somerset County Council for their contribution to this poster.

Computational Thinking

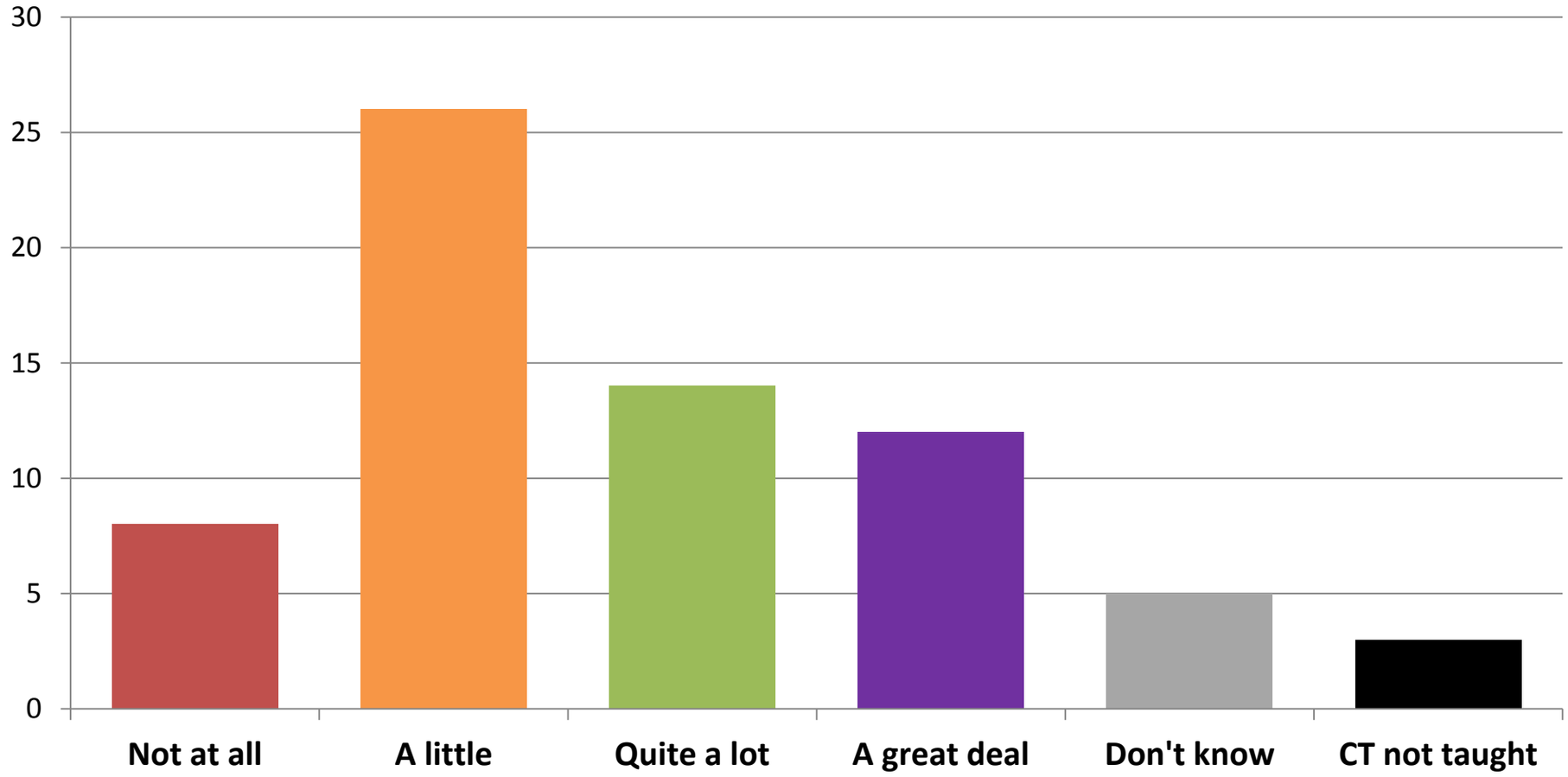
Bubble Sort:

- Meets many of the priorities of students with SEND, e.g. communication, numeracy, life skills, problem solving

- *But is this really computational thinking?*

Computational Thinking

- To what extent do you feel developing computational thinking skills has supported learning in other areas of the curriculum for pupils?



"Computational thinking has been using in a variety of contexts such as DT, cooking, geography, maths and English. Through the use of algorithms it allows pupils to grasp concepts or processes from these subjects. It allows them to physically follow the process from start to finish."

Computational Thinking

"Transfer of learning across contexts does not happen automatically."

Grover & Pea, in Computer Science Education, ed. Sentence et al, 2018

Requires explicit links to be made between what is taught in computing in other subjects and use of specific language of computer science

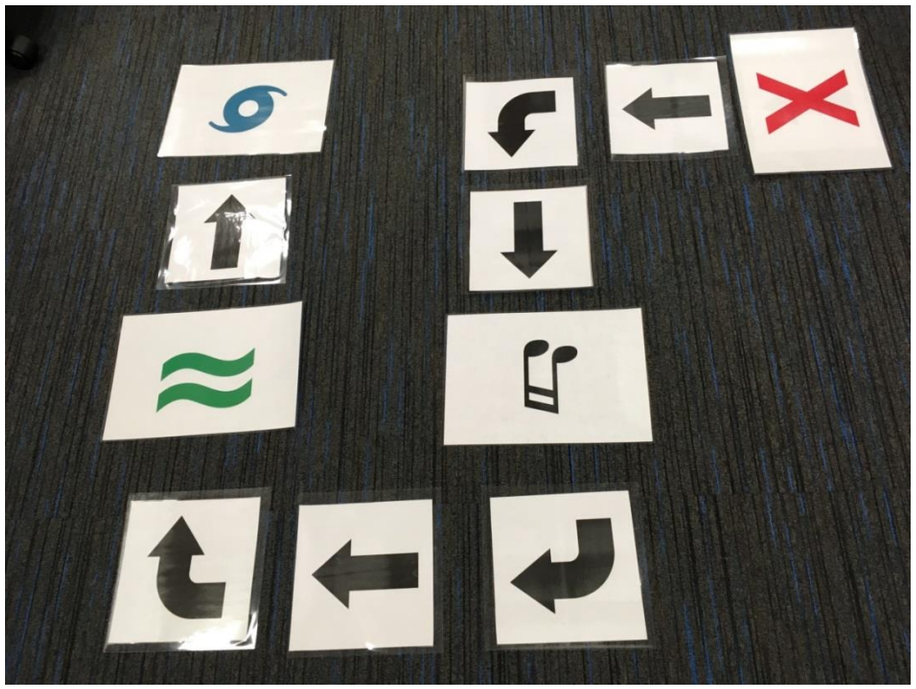
Computational Thinking

"We did a whole topic before the summer based on algorithms, so I had several different classes making algorithms. The pupils then took that into maths and started to create their own algorithms for doing angles, for example, in triangles, and area in squares. They created their own algorithms, sequenced them, passed them to another pupil, debugged them then took it back and amended it. And it really did help in maths…"

Ryan Hayes, ICT Lead Teacher - Wales

Unplugged Activities

"[Unplugged activities] can make it much easier to explore the concepts involved and makes it easier to ask questions about things that aren't understood. […] By providing a physical representation, the learner can point to and ask the question at the level of the analogy rather than having to fully verbalize it at the technical level."



Curzon et al. in Computer Science Education, ed. Sentence et al, 2018

Unplugged Activities

Unplugged CT activities:

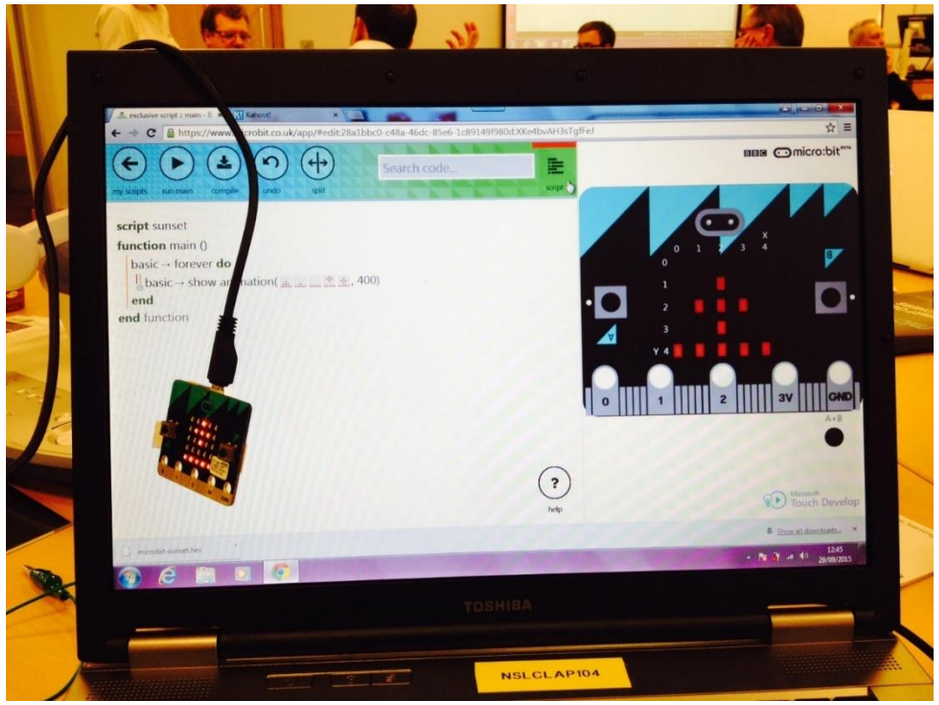
- Provide **multimodal** ways of accessing learning
- Can be **personalised** and relate to pre-existing knowledge to reduce cognitive load
- Can provide a **concrete representation** of an abstract concept
- Can provide students with the **language** to talk about their learning and an approach to **solving problems** in other curriculum areas
- Can be made **relevant** to a range of learners who may or may not be working with computational models

Programming & CT

- Which strategies have been effective in teaching programming and CT to your students?

- Personalised, relevant context (9)
- Unplugged activities (19)
- Physical computing (24)

Physical Computing



"When I first started teaching here I tried to do more complex stuff like Scratch, HTML, MS Logo and different things like that, and the pupils just weren't interested whereas things where they can actually build it, or program it with an instant effect seems to keep their interest and keep them motivated."

Ryan Hayes, ICT Lead Teacher

Resources (CS)

- What resources have you found effective in teaching programming and computational thinking to your students?

Physical computing:

- **Floor robots**, e.g. Bee-Bot (16)

Apps:

- **Scratch Jnr** (5)

Websites/software:

- **Scratch** (10)

Teaching resources:

- **Barefoot** (8)

Conclusions

- Some great examples of effective and creative practice
- Patchy coverage across the country
- Need for more resources and advice
- Common approaches to CT and programming
- Computational Thinking as one possible problem-solving approach to benefit SEND learners

Next Steps

- Creation of a bank of resources for SEND Computing (http://www.sendcomputing.info/)

- Recommend further research into effective programming approaches and computational thinking benefits for SEND learners

Any questions?

celliott@sheffieldclc.net

@catherinelliott