Assessment

Assessing and tracking progress

How can we collect evidence of learning?

When Ofsted reported on ICT in schools, assessment came in for particular criticism. Assessing computing can provide some particular challenges.

- It's too easy to focus on the outcomes of a task at the expense of assessing the learning that takes place in the process.
- It's too easy to focus on assessing pupils' skills in using particular software instead of assessing their knowledge and understanding.
- If pupils have worked with a partner or in a group to complete a project, how can you assess each individual's learning?

You can do much to meet these challenges and develop robust approaches to assessment, so that you can form a judgement about what individual pupils can do, know and understand, as well as helping pupils themselves reflect on how they've applied computational thinking.

Blogs for showcasing, reflection and feedback

Probably, the most effective thing you could do is to start a class blog. Ask pupils to use this to upload the outcomes of their work and document the computational thinking processes they worked through, focusing on any challenges they overcame.

Blogs provide a way for pupils to get feedback through the comments section. Invite pupils to respond to any questions raised. You can create a tagging system so you and your pupils can use their blog to track progress. A blog begun in Year 1 and continued up to Year 6 would provide rich evidence of both progress and attainment.

Other approaches

Naace suggest using an interview at the end of a project. A pupil might explain the computational thinking they used in solving a problem, but could also reflect on what and how they have learnt.

There's a place for formal testing in computing. In programming work, code tracing and debugging challenges are useful ways of assessing both specific knowledge of a programming language as well as logical reasoning and other problem-solving skills.

Evidence for pupils' computational thinking will be found in how they approach projects, but well-designed questions might provide one way of assessing this more directly. The Beaver Challenge (see Further resources) is a series of online questions assessing computational thinking.

How can we track progress?

The 'Progression Pathways Assessment Framework' is used by many for both planning and monitoring progression in computer science. It was derived from CAS's original computer science curriculum and provides detailed treatment of progression across six strands:

- algorithms
- programming and development
- data and data representation
- hardware and processing
- communication and networks
- information technology.

The authors, Mark Dorling and Matthew Walker, explain:

> *The progression through each strand of computing is broken down into rows. The rows are colour coded (like karate belts) to help the teacher to assess whether students are showing competence at different levels and recognise achievement or attainment.*

Schools can choose to assign arbitrary values (levels) to the coloured rows if they would like to use them with existing reporting systems.

The focus of this assessment framework is progression through and across the strands of computing. If you plan to use this assessment framework with your existing assessment/reporting system then you can agree the benchmark 'level' for the pupils entering a particular key stage and assign the arbitrary benchmark value (level) to the appropriate progression statements for each strand.[5]

They recommend that primary teachers focus on the badge statements from the Pink to Purple rows.

How can we assess attainment?

The old national curriculum levels have been removed and not replaced. The statutory attainment target is clear:

By the end of each key stage, pupils are expected to know, apply and understand the matters, skills and processes specified in the relevant programme of study.[6]

Computing in the national curriculum: A guide for primary teachers (see Further resources) outlines an approach to assessment based on tracking achievement of the individual statements from the programmes of study. The advantage of a granular method like this, similar to EYFS assessment, is that it shows pupils, parents and teachers exactly what has been achieved and what aspects of the curriculum remain targets for subsequent work.

Evidence of attainment

Even just one Scratch script, such as this for a duck shoot game: http://scratch.mit.edu/projects/15907506/#editor, provides evidence of attainment for the key stage 2 programme of study.

From the Scratch scripts themselves, we have evidence of:

- write programs that accomplish specific goals
- use sequence in programs
- work with various forms of input (keyboard and mouse in this case)
- design programs that accomplish specific goals
- design and create programs
- use repetition in programs (forever loop, two different repeat until loops)
- simulate physical systems
- use selection in programs (if ... then ... else)
- work with variables (score).

If pupils had also explained how they'd solved the problems, then you might also have evidence of a number of the 'logical reasoning' statements.

SEN

Pupils with special educational needs working below the level of the programme of study for their key stage should be assessed using the P-scale statements, as in the past. There's much in these statements which reflect emergent computational thinking.

Digital badges

Digital badges can provide a great way to record and reward pupils' attainment in computing. One approach (used by Rising Stars with Makewav.es) would be to have a badge for each of the bullet points in the programme of study, with clear criteria for each particular bullet point. If you're using the 'CAS Progression Pathways Assessment Framework', there are Makewav.es badges for these too.

Further resources

- Bebras International Contest on Informatics and Computer Fluency and Computational Thinking Challenge, available at: www.bebras.org/ and www.beaver-comp.org.uk/.
- Berry, M., *Computing in the national curriculum: A guide for primary teachers* (Computing At School, 2013), available at: www.computingatschool.org.uk/data/uploads/CASPrimaryComputing.pdf.
- Brennan, K. and Resnick, M., 'New frameworks for studying and assessing the development of computational thinking' (2012), available at: http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf.
- DfE (2014) P-scales: attainment targets for pupils with SEN, available at: www.gov.uk/government/publications/p-scales-attainment-targets-for-pupils-with-sen.
- Dorling, M. and Walker, M., Progression Pathways Assessment Framework (2014), available at: http://community.computingatschool.org.uk/resources/2324.
- Makewav.es badges for Progression Pathways, available at: https://www.makewav.es/cas and for the Attainment Targets, available at: https://www.makewav.es/badges/18419/.
- Switched on Computing Scratch debugging challenges, available at: http://scratch.mit.edu/studios/306100/.

[5] Progression Pathways Assessment Framework (Computing At School).
[6] National Curriculum in England, *Computing Programmes of Study* (Department for Education, 2013).