Children's Wellbeing – Learning and Achievement Service – Herefordshire Computing Support

The Herefordshire Primary Computing Progression 2020

Key Stage 1 (Years 1 & 2) – a revised version of the 2014 materials

This document can be found, with supporting resources, at
herefordshirecomputing.com

Learning and Achievement Service
Herefordshire Computing Support
hands on learning and teaching

Herefordshire Council

Table of Contents

Introduction

Computer Science (CS): Programming

Computer Science (CS): Computers & networks

Information Technology (IT)

Digital Literacy (DL - Online safety)

Assessment

Introduction

1 From 2014 to 2020

Six years on from the introduction of the Computing Curriculum and the original version of these materials things have moved on somewhat and so here is a revision of the Herefordshire Progression materials. So what's changed?

Computer Science

No changes here to the learning statements, except to divide them into year groups rather than pairs which is always requested by single year group schools – however please don't take that too literally. What has changed though is the bank of recommended lesson material. These are drawn entirely from three key sources: Barefoot Computing, Code Club and Code-IT (Phil Bagge). Recently it has become possible to use iPads to run Scratch albeit missing certain functionality (especially keyboard presses). The suggested activities lists for each year group indicate what will work on iPads.

The *Computers and Networks* lists of resources (KS2) have been updated.

Information Technology

The headings have been simplified (reduced) here. Learning statements have been updated and reduced, as have the cross curricular examples. The content has been altered to reflect the far greater proportion of iPads now in schools and relatively few computers.

In response to many requests over the years I've finally given way and split the learning statements in each booklet between the two year groups. The most important driver of the use of technology remains the context in which it is used, however, and this will often dictate the skills that are developed. The progression is only a guide and should be secondary to use of technology to support children's learning in this strand.

Digital Literacy (Online Safety)

This section has been entirely replaced. There remains no shortage of lesson material for online safety and many of the old favourites, which teachers know their way around remain in the table of possible of resources (with a few newer ones). Primarily though, the structure of this strand, and the suggested resources comes from the recent document Education for a Connected World and the even more recent resource bank that supports it, Project Evolve. There are eight strands at every level and often several statements within each – so quite a bit more content here than before.

It should be said that these resources reflect something of a more informed and intelligent approach to online safety, recognising that some of the messages we have given to children in the past have never really had a reasonable chance of having any impact. This should be born in mind when evaluating which older materials should be retained and which replaced.

Assessment

The more complex assessment grids have been removed and only the "simple" one remains. Some learning statements have been reorganised to create six rather than 5 "levels"

Supporting resources

These have been adapted to reflect the changes above. The software maps have been removed and the recommended iPad App list updated - this is always an evolving document anyway.

2 Organisation of the program of study

Computing breaks down into three main areas as follows (extracts from the programme of study that follow are from the *purpose of study* and *Aims* sections and are for KS1 – KS4):

Computer science (CS)

The programme of study states:

> The core of computing is computer science, in which pupils are taught the principles of information and computation, how digital systems work, and how to put this knowledge to use through programming.
>
> The national curriculum for computing aims to ensure that all pupils:
>
> - can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation
> - can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems

The main component, then of CS is programming where children engage in practical, creative experiences. There is also a requirement for children (especially at KS2) to understand physical computer systems. We've called this **Computers and Networks** in these materials

> - Pupils should be taught to understand computer networks including the internet; how they can provide multiple services, such as the world wide web; and the opportunities they offer for communication and collaboration

Information technology (IT)

> … Building on this knowledge and understanding, pupils are equipped to use information technology to create programs, systems and a range of content.
>
> The national curriculum for computing aims to ensure that all pupils:
>
> - can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems

This strand involves the productive use of readymade computer applications to create content in a variety of forms (text, image, sound, video, animation, data gathering and processing, AR, VR …) often supporting the whole curriculum. It is important that we don't lose sight of this aspect of computing.

Digital literacy (DL)

> … Computing also ensures that pupils become digitally literate – able to use, and express themselves and develop their ideas through, information and communication technology – at a level suitable for the future workplace and as active participants in a digital world.
>
> The national curriculum for computing aims to ensure that all pupils:
>
> - are responsible, competent, confident and creative users of information and communication technology.

This is online safety, "responsible" use of tech. This is far more explicit in each of the specific key stage statements of the programme of study than it is in these general ones.

3 Computational thinking

So where does computational thinking fit into all this then? It really pervades the whole subject; that said it sits most prominently in **Computer Science.**

It is a kind of literacy; a way of organising thinking that is not only relevant to computing but very much to life and learning and therefore goes well beyond the subject of computing and indeed beyond education itself – it is a skill for life. When children are "learning to code" they are also "coding to learn."

Computational thinking is a slightly problematic term; there are some very different interpretations of it. Google has a helpful page, where they identify core components of this as algorithms, decomposition, abstraction, and generalised patterns. The influence of these ideas on the computing curriculum is clear, with algorithms at KS1 and decomposing problems at KS2. Whilst these are certainly part of the computer scientist's toolkit, there's perhaps more going on than when they come to tackle new problems or design systems.

Resnick and Brennan, of Scratch fame, have an interesting paper in which they revisit break this down into concepts, practices and perspectives – it is worth reading. Whilst these concepts, practices and perspectives can (and should) all be learnt through practical experience of programming, teaching needs to go beyond the 'this is how you use Scratch' or 'this is how you use Kodu' if we are to do justice to the ambitions expressed here.

It's important that we all have an understanding of the principles and concepts behind computational thinking and keep them in mind when designing and delivering the curriculum.

4 How to use these materials

The progression is organised into three pairs of year groups:

- KS1 (Years 1 and 2)
- Lower KS2 (Years 3 and 4)
- Upper KS2 (Years 5 and 6)

Broad brush strokes are important and use of computing, especially in the IT strand should always be dictated by its relevance to learning generally. I've resisted calls to break down statements to individual year groups but Single year group classes often prefer to have "ownership" of learning statements within each class so "2" after a statement indicates that could be Y2 goal.

The Progression is based around the three headings from the programmes of Study (explained above) and each of these is broken down further but it different ways and with different resources depending on the strand.
This Computing Progression (especially the information technology [IT] strand) should be integrated into planning across all subjects. A **long term planning grid** is available as a simple mapping tool to help start this process and to ensure that the full range of computing entitlement is covered.

Each school should also develop its own resources map. A sample, containing recommendations for Herefordshire primary schools is available to use as a template and can be downloaded (along with all the supporting resources for this Progression) at www.herefordshirecomputing.com

5 Help and support in your classroom

Support in Herefordshire comes in the form of *Herefordshire Computing Support,* a part of Herefordshire Council's *Learning and Achievement Service.* Many primary schools in Herefordshire subscribe to our service level agreement which means we're at your disposal to come and help you with planning and with hands on support in your classroom. If your school doesn't buy into our service level agreement, or if you're in a neighbouring local authority we can still help but will need to charge on a "pay-as-you-go" basis for the support we provide. Please contact msanderson@herefordshire.gov.uk or talk to your school's Computing Coordinator to sort out what's possible.

Additionally a pool of resources is available for loan to schools to support areas of computing that are expensive to support. Courses are provided and the Digital Leaders Network is a popular initiative in schools.

Computer Science (CS): Programming

KS1 Programme of study extract

Pupils should be taught to:

- understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions
- create and debug simple programs
- use logical reasoning to predict the behaviour of simple programs

What does this mean?

Programming comes with a language of its own and getting one's head around the terminology, then using it correctly with children is half the battle and it is important that we understand the terminology and discipline of computational thinking and computer science and that we use it with children. That said, all of the above happens as a natural part of the process of programming and that's how we should (and do in these materials) approach it. So here's a blow-by-blow explanation of some of the more troublesome terms used in the program of study:

- **Algorithm** – a set of steps that accomplish a task. So a program is an algorithm, but so is the story board the children created before they started programming. See Barefoot: Teach yourself

- **Programs** – software, containing a detailed sequence of instructions, which runs on a computer or other digital device. Children create a simple program when they give instructions to a Bee-Bot or similar device.

- **Digital devices** – any electronic device (in this case that is capable of being programme).

- **Debug** – correcting errors in a program. If the bee-bot doesn't quite arrive at the desired location then the bug in the instructions and fix it. Debugging tends to happen naturally once children start programming but needs to be discussed and children need to develop strategies.

How shall we teach it?

This is likely to be the most challenging aspect of the computing curriculum for you to deliver (not necessarily so for the children, which somewhat adds to the problem!)

Ideally, programming should be embedded into the curriculum and once we all understand what's really involved this will be possible. In Early Years and in early KS1 this is perhaps already easy to achieve, as learning is child centred and child led, but becomes more difficult as children progress through KS1. At that stage, for the moment it's going to be a lot easier to teach it as a discrete element on a timetabled basis, perhaps as one or two half term units per year.

At the very beginning of KS1 programming will probably be based around very physical devices, indeed it's good to start with the children themselves (can a child "program" another child to walk from A to B) this kind of thing is now often referred to as an "unplugged" activity – i.e. away from the computer.

Devices such as Bee-bots, Roamers, Pro-Bot etc are common in Early Years and lower KS1. As children move on through Year 1 and certainly into Year 2 the programming they have done on the floor will move up onto devices, perhaps using apps such as the Bee-bot, Bee-bot Pyramid or Daisy the Dinosaur, or the Bee-Bot PC software. Children should also make a start on more advanced programming and an excellent tool for this is *Scratch Junior*.

These materials attempt to give an indication of how progression in programming might be achieved, in a variety of ways, in the grids that follow. It is, however, very difficult (and often unhelpful) to define a particular order in which skills should be introduced to children. The open nature of programming environments such as Scratch and Scratch Junior, the skills that are required for any particular project and the spirit of creativity that runs through the programme of study means that you will almost certainly jump around in any skills list. You may very well plan that you are only going to cover some very basic skills in a particular session but some children will quickly want to build on them and make much more advanced things happen. They may want to take things in a different direction and at some point you should give them this freedom. The point at which you do this is up to your professional judgement and is part of your learning curve as a teacher.

Scratch Junior (and Scratch) is of course not the only programming environment, there are many others including quite a few which require an annual subscription. If your school is using one of those then you will of course wnat to follow the scheme of work that comes with your solution for the time being. That will give you a great opportunity to build up your skills for a year or so.

Scratch Junior is great fun and it will be worth looking at later on even if you don't immediately. Giving your children access to an open environment of this nature is really much more within the spirit of the computing curriculum than following a tightly defined sequence of lessons that will, to a degree, stifle creativity.

Whatever you choose to use, you will need to think carefully about the skills your children have. Be prepared to go back and program floor robots, or even to have children programming other children acting as robots if you need to before attempting some of the more advanced activities. Although it is desirable that children are using more advanced on screen programming tools by the end of KS1 there is actually nothing in the KS1 programme of study that can't be delivered with floor robots.

How does it link to other aspects of computing?

- The programme of study extracts above are those primarily concerned with programming. But the very opening reference to programs running on digital devises illustrates the link to any use of technology and particularly a link to the **information technology** strand.
- Programming obviously links to the other aspect of **computer science** concerning **computers and networks** as programs are fundamental to physical computer science. Specifically at KS1 it is appropriate to link the programs the children design and create to the way in which they are influenced by and relate to "common uses of technology beyond school"
- If children share their projects, and look at other people's there are links to **digital literacy (**online safety**)**

- It is worth noting that there are also strong links to other subjects, especially English and mathematics and these will become apparent as you travel further down the programming road.

What kit, software and resources are there to help me (organised into a progression, simple to more sophisticated)

We've arranged the first part of this list into a natural progression of kit beginning with the simplest floor robots and moving onto on-screen programming resources. Please remember that work on algorithms also involves non-technical kit, and especially children giving instructions to each other as one of the first stages in computer science. We've highlighted our recommendations.

Bee-Bot: one of the simplest programmable robots. Programming is via the simple (up, down left, right – go) keypad and results are instant. It comes with an amazing array of supporting resources that are well worth looking at as they will give you plenty of ideas for activities to extend the possibilities to areas you would never have considered. The best place to look is on the TTS site

Constructa-Bot: The same keypad as Bee-bot but this time in the form of a digger which can carry other items. It has the facility for children to build onto it using LEGO, Kid K'Nex and other construction materials and it will also hold a pen to leave a trail. As with Bee-bot there is a large range of resources you can buy to go with it. The best place to look is the TTS site

Pro-Bot – The next step up in the "-bot" range! Especially useful towards the top of KS1 and into KS2 where Bee-Bot will be too basic. Much more advanced programming is possible (the limit really, while crawling on the floor!) with repeat commands possible and the robot itself has inbuilt light sensors to control working headlamps. It can be voice activated and has sensors in the bumper. Also has a screen and can be programmed at the computer using optional software. See TTS

Roamer – now called roamer classic, it's been around for years. It offers a through solution from EYFS to lower KS2. Valliant (the manufacturers) have an excellent search tool on their site linking to a range of lesson activities (free and paid for) from many sources. Roamer World software allows you to program at the computer and links floor activities to on-screen programming.

Roamer-Too – A very substantial new tool (pre-school to A-level). It achieves this by coming with different interchangeable keypads for different age groups.

Pip and Pixie – from Swallow systems are in a lot of Herefordshire schools and have been for years because they last for years. It has to be said that they are not the most imaginative looking devices though these days and are expensive.

Tes-iboard is a great place to look for some simple activities for programming and pre-programming. Many are free and there are many more if you subscribe.

Bee-Bot (iPad app - free) & Bee-Bot Pyramid (£0.69) ☑ EYFS ☑ KS1 ☒ KS2

A free app from TTS. There are 12 mazes of increasing complexity around which children write programs to program the bee-bot. Points are given as they do so. A must for all EYF and KS1 children. Ideally children will have had experience of programming a bee-bot in real life before or alongside this. Bee-Bot Pyramid takes this on further into an Ancient Egyptian setting again with 12 mazes.

Daisy Dinosaur (iPad app - free) ☑ EYFS ☑ KS1 ☒ KS2

A great little app for EYFS and KS1. Best to use it in *challenge mode* first where five simple challenges teach children how to use it. After that, *free-play mode* allows them to set their own challenges. Based on simple Logo programming and great for EYFS / KS1

A.L.E.X. (iPad app - free) ☒ EYFS ☑ KS1 ☑ KS2

A fun puzzle game and a great way to train your brain. A.L.E.X. helps you think and plan logically as you program your robot, A.L.E.X., with a sequence of commands to get through a series of levelled challenges. The lower levels of the games are suitable for children as young as six and the game is enjoyable for adults too! FREE VERSION - Includes 25 levels and the ability to create your own puzzle.

Kodable (iPad app - free) ☒EYFS ☑ KS1 ☑ KS2

Kodable is a free educational iPad game offering a child-friendly introduction to programming concepts and problem solving. For ages 5 and up, and tools for grownups too! In the latest version teachers can create account for children

Scratch Junior: www.scratchjr.org An iPad app with an Android one and a browser based version also available. It's really aimed at KS1. It's a lovely programming environment which includes a subset of commands from Scratch (its older and bigger brother) that are just right for KS1 and EY. The web link given here is to a website for education professionals where some learning and teaching resources are already appearing. It is, and will remain, free.

Scratch: www.scratch.mit.edu Widely accepted as the mainstream coding environment, certainly for KS2. Scratch set the benchmark for all similar environments. This is the big and much more powerful brother of Scratch Junior and may be useful at the top of KS1 for children who eventually exhaust the possibilities of Scratch Junior. Scratch is totally free.

Complete schemes of work:

Espresso Coding: www.espressocoding.co.uk Espresso have created their own coding environment and supported it with many readymade, self-contained lessons. This is a complete scheme of work (Y1-6) and a great way of building a teacher's skills (especially in KS1) but the approach isn't really within the creative spirit of the computing programme of study. It certainly is not open ended enough for upper KS2. Annual subscription is necessary.

2Code (Purple Mash): www.purplemash.com A rather similar approach to Espresso Coding and the programming environment has a similar feel. It is much easier to access the full coding environment than in Espresso Coding and differentiation is built into each of the activities. A subscription to Purple Mash (an excellent and popular resource), of which 2Code forms a part, is required.

J2Code (J2E): www.j2e.com/j2code The coding environment is very similar to Scratch. There are also some quite open ended activities with supporting documentation and videos built into the product aimed at KS1, Lower KS2 and Upper KS2. KS1 actually uses JIT's Logo program rather than J2code. Limited access is free (but you can't save anything). It's possible to buy it on its own but the only sensible way to do it is to buy the whole of J2E as a package (a very useful resource).

Tynker – www.tynker.com/school a US online programming site with an ITunes and Android app too. It contains a great resource that some schools used recently as part of the hour of code. It uses the same style of blocks as Scratch. It contains a complete scheme of work but you have to pay for this ($399 per year per classroom). There is a lot you can do for free though and skills are easily transferable to Scratch.

Useful planning and background sites:

Scratch Ed site - scratched.gse.harvard.edu/ - This site is dedicated to educational use of Scratch and is aimed at educators. You'll find many activities and resources here that you can use in the classroom as well as tutorials and educational discussion from the Scratch Ed community. The Scratch Cards are a particularly useful resource – worth printing out and getting to know them.

Junior Computer Science – Phil Bagge – www.code-it.co.uk - This is an excellent site packed with comprehensive resources all ready for you to use in the classroom. Many of the lessons that follow are from this site. Phil Bagge is a Computing Adviser with Hampshire LA and until recently has been teaching has been teaching across five primary schools. He is a regional coordinator and master teacher for CAS (see below)

Computing at School (CAS) – www.computingatschool.org.uk - CAS is a community of teachers, academics, industry professionals, school governors, parents etc. with a mutual interest in computing. The CAS community has very much been instrumental in the development of the new curriculum. There are many resources on their site, though not always easy to find them. We've showcased many in this progression.

Barefoot Computing – www.barefootcas.org.uk – A very new organisation helping primary school teachers get ready for computer science. They are developing free high-quality, practical resources and computer science workshops to support primary school teachers in England. Funded by the DfE. You'll find "teach yourself" resources as well as lesson material.

Computational Fairy Tales – a wonderful collection of over 70 stories, in fairy tale style, that attempt to explain many aspects of computer science. Many will be found on the website and you can download them. A great way of reinforcing learning in a way that will appeal to children. You can search by topic or level. Topics covered well exceed the expectations of the KS1 and 2 programme of study.

Programming: progression by objectives

Understanding algorithms

- Give precise instructions to, and respond to instructions from, other children involving movement around the room.
- Describe what actions are needed for a particular task (not necessarily an IT one) and begin to use the word algorithm.
- Begin to understand that sequence (order) is important when devising algorithms and programming devices
- Understand that a number of different algorithms will often all solve the same problem. 2
- Be able to predict what will happen in an algorithm or program which they may not have written themselves. 2
- Understand why algorithms are useful for solving a wide range of problems and that we use algorithms every day 2

Programmable robots (Bee-bot, Roamer etc)

- Describe clearly what they expect to happen while programming a robot.
- Begin to understand that sequence (order) is important when devising algorithms and programming devices
- Understand that programs respond to inputs to carry out actions.
- Be able to predict what will happen in an algorithm or program which they may not have written themselves. 2
- Be able to execute a program, observe the results carefully spot errors and be able to debug them. 2

Programming: Progression by activity / lessons

- Human Crane – Algorithm design - lesson plan & Resources – Phil Bagge
- Bee-Bot / Blue-bot scheme of work – many useful lessons here from Phil Bagge.
- Bee-bot app Algorithm design – A differentiated activity focussing on designing algorithms for programming the Bee-bot app. Pupil prompt sheet here – Simon Haughton
- Algorithm design lessons and activities – many great ideas for exploring what algorithms are from Simon Haughton. Supported by resources
- Read the story of The ant and the grasshopper: A fable of algorithms and talk about it with the children (there are many more like it on this site)

- Bee-Bot / Blue-bot scheme of work – many useful lessons here from Phil Bagge.
- You'll find a vast range of lessons, activities and resources (often involving a purchase) on the TTS site
- Valliant, the manufacturer of Roamer, have an excellent search tool linking to a huge range of lesson activities (free and paid for) from many sources. Many activities apply equally well to other programmable robots.
- Use a more advanced robot (such as Pro-Bot) to write more efficient programs using repeat commands (e.g. draw a square). Encourage to work together, discuss possible algorithms, design, test refine and debug.
- 5 lessons on Pro-Bot from Simon Haughton, leading to using repeat commands. Detailed planning and pupil's resources available free of charge.

Programming: progression by objectives

On-screen programming – short activities

- Describe clearly what they expect to happen while programming an object on screen.
- Understand that sequence (order) is important when devising algorithms and programming devices
- Write programs successfully to create movement on-screen.
- Be able to execute a program, observe the results carefully spot errors and be able to debug them. 2
- Understand that programs on screen respond to inputs to carry out actions. 2
- Use different kinds of inputs in programming (key press, mouse click tap on a sprite, automated start condition …) 2

Programming: Progression by activity / lessons

- Bee-bot app Algorithm design – A differentiated activity focussing on designing algorithms for programming the Bee-bot app. Pupil prompt sheet here – Simon Haughton
- TES connect is a good repository of lesson plans for Bee-bot activities (you'll need to create a free account)
- Computer software based on programmable robots such as Bee-Bot Lesson Activities and Roamer World are a great way of takings children's learning with physical devises up onto the screen.
- Get children to explore the SWGfL on TES *Big Day Out* simulation, especially activities in the West Midlands region.
- Poisson Rouge (for a very small subscription) is a great source of short focussed activities for EYFS and KS1
- iPad apps (many of them free) for programming are appearing thick and fast now. Good ones for KS1 often include a structured series of activities to get children into the app followed by free exploration. Try Daisy the Dinosaur, A.L.E.X, Bee-Bot and Bee-Bot Pyramid.

Programming: progression by objectives

Open ended programming (Scratch Junior / Scratch)

- Understand that a number of different algorithms will often all solve the same problem.
- Describe clearly what they expect to happen while programming a robot.
- Begin to understand that sequence (order) is important when devising algorithms and programming devices
- Be able to predict what will happen in an algorithm or program which they may not have written themselves. 2
- Write programs successfully to create movement on-screen.
- Be able to execute a program, observe the results carefully spot errors and be able to debug them. 2
- Understand that programs on screen respond to inputs to carry out actions. 2
- Use different kinds of inputs in programming (key press, mouse click tap on a sprite, automated start condition …) 2

Programming: Progression by activity / lessons

- Scratch Junior programming activity – Drive across the city – Sprite selection and programming simple movement
- Scratch Junior programming activity – Dance Party – Background and Sprite selection, programming movement and sound, programming events on sprite collision.
- Scratch Junior programming activity – Dribble a basketball – Background and Sprite selection, programming movement, beginning to use repeat commands.
- Scratch Junior programming activity – Spooky Forest – Background and Sprite selection, programming movement and sound, using repeat commands, using click / tap sprite as an input.
- Scratch Junior lesson sequence (9 lessons) – A well designed and detailed series of lesson plans introducing Scratch Junior and extending its use well beyond the objectives in the KS1 programme of study.
- Scratch Junior lesson sequence (several lessons - English) – A well designed and detailed series of lesson plans which cover more advanced use of Scratch Junior in the context of letter names and sounds, handwriting and upper and lower case letters.
- Phil Bagge has many brilliantly planned lessons for Scratch Junior
- Barefoot computing is a beautifully constructed site with a wealth of material for KS1 accessed via an efficient search system. You will need to create a free account.

Computer Science (CS): Computers & networks

KS1 Programme of study extract

Pupils should be taught to:

- recognise common uses of information technology beyond school.

What does this mean?

This statement is fairly self-explanatory. We have added in some progression statements that cover good practice in basic file management (opening and saving work) so that children begin to lay good foundations for KS2 and be well placed to go on and develop a fuller understanding of how networks and computers work, what services they typically offer and how computers process information.

How shall we teach it?

The programme of study statement will already have been touched upon in Early Years, not least through role play involving digital devices (working or not) from the home. This should be continued into KS1. Aspects relating to file management and how computers work will really best taught in context, often through regular reminders to save work with a conversation about where to and why.

How does it link to other aspects of computing?

- There are obvious links to **information technology**, when saving and sharing work locally and online.
- Links to **digital literacy (**online safety**)** when using or creating internet resources or communicating online.
- Clearly there are links to the **programming** strand of **computer science** through the obvious links between understanding programming and the systems on which programs run.

What kit, software and resources are there to help?

Junior Computer Science – Phil Bagge – www.code-it.co.uk - This is an excellent site packed with comprehensive resources all ready for you to use in the classroom. Many of the lessons that follow are from this site.

Computational Fairy Tales – a wonderful collection of over 70 stories, in fairy tale style, that attempt to explain many aspects of computer science. A great way of reinforcing learning in a way that will appeal to children. You can search by topic or level.

Teach Computing is an excellent site with resources for the whole of the Computing curriculum, It's particularly rich on this strand with many really well constructed lessons by Andy Bush. You will need to register for a free account on the site before you can access the materials https://teachcomputing.org

Progression Statements
CS: Computers & networks

Lesson ideas and resources

Information technology beyond school

• Be aware of obvious uses of technology in and beyond school (i.e. things that clearly look like computer devices)	• Walk around the school and photograph all the uses of technology that the children can find and use the photographs for a display. As a possible extension, photographs could be joined in a web to show how they connect to each other.
• Understand some of the things that people do with computers at work and at home.	• Design a poster showing uses of technology in and beyond school. Talk to parents about how they use technology at work.
• Have a growing awareness of things in and beyond the home that have some kind of computer in them (microwave, washing machine…) 2	• On a trip to a local supermarket spend some time looking at how technology is used obviously and behind the scenes in the store. How a supermarket works is a good activity from Phil Bagge
• Understand that most computers, tablets and phones are connected to the internet. 2	• Children should experience repeated use of internet services in the course of their computing work. Take time out to discuss them and what other devices might usefully access them, talk about the advantages and disadvantages of each.
• Recognises that any one of a range of digital devices can be considered a computer. 2	• Ask children to bring in example of different types of digital technology (the real thing, or photographs) • Keep a diary of the number of digital devices they or their parents use in a day
File management	
• Be able to save (and successfully retrieve) their own work on a tablet.	• Save / open files in the course of learning. Discuss different ways of saving for different devices / purposes.
• Understand that sometimes data is stored in "the cloud" to make it accessible on other devices and by other people. 2	• Use Seesaw to build their own portfolio of work, to share it with others and to comment on others' work. Show the children wireless access points in the classroom and cables connecting them and wired computers. Trace their path back to the server…
• See related statements and lesson ideas in digital literacy around the importance of passwords, sharing files on the internet etc.	

Information Technology: Multimedia (including online tools)

Text | Image, film & sound | Internet | Data sorting

KS1 Programme of study extract

Pupils should be taught to:

- **use technology purposefully to create**, **organise,** store, manipulate and retrieve **digital content**

What does this mean?

This aspect is all about using readymade applications to communicate; to create, gather and share information. It's the sort of thing that teachers have been doing, and an important part of Computing and the whole curriculum.

At the early stages children will be focussing on skill development and awareness of technologies but by the end of key stage two the focus should be shifting to real consideration of audience and purpose and to making sure that what they create is truly fit for purpose and to growing independence in selecting the right tool(s) for the job.

The programme of study does not mention specific technologies or media but the kind of things we should be doing here (in various combinations) are:

- **Text processing and design** – Yes, word processing, desk top publishing (typing skills even, at a basic level). Online this means blogging, posting, commenting, wikis and similar social media tools – Seesaw!
- **Digital image, film and sound** – including animation.
- **Using the Internet –** the focus in this strand on search skills etc.
- **Working with simple data –** sorting, classifying, simple graphing.

How shall we teach it?

The better question would be "Why?" The answer should really be that it enhances learning in the curriculum as a whole, and there are links here across every subject. This then leads to the "how" question and the answer is within other subject areas. Having said that, there may well be times when computing concepts need to be developed in isolation first and this should be left to individual teacher's discretion. For this reason schools may decide not to timetable this aspect of computing, unless access to kit dictates that this is necessary.

It is most important that a long term planning grid is completed at the start of the year to ensure that there is coverage, perhaps across the two year groups, of a range of computing tools. A possible format for that will be found at herefordshirecomputing.com.

How does it link to other aspects of computing?

- There are strong and important links to **digital literacy (online safety)** when using or creating internet resources or communicating online
- Any work online, searching for resources links strongly to **computer science** where there is a requirement at KS2 to understand the internet, World Wide Web and how search engines rank their results.

What kit, software and resources are there to help?

Please see the recommended iPad apps list on the website.

For precise lesson ideas Mr P ICT (Lee Parkinson) is also a truly excellent resource but you have to subscribe. He really is worth his money though!

Progression statements

Text & design

- Select or create appropriate images / sound to add to work
- Write and send a short comment perhaps in Seesaw
- Understand the different ways that messages can be sent, email, text letter, phone … and begin to consider advantages of each 2
- Edit work in the light of their own discussions and observations. 2
- Develop familiarity and correct use of a keyboard (onscreen or real) – spacebar (single press not "finger space"), backspace, shift / caps lock), return etc. 2

Cross curricular application of skills

- Use their own work to create and publish their own digital picture book.
- Write a caption for a non-fiction book or on photos of themselves
- Children's poems are published, by word processing or presentation apps, to combine words and images.
- Children sequence a process using Shadow Puppets for example making a model or making biscuits
- Children photograph riding a bicycle and add a caption and/or narration
- Make labels or captions to match objects on display in a class toy museum
- Use images and text to tell the life story of an historical character
- Children add sound effects to a poem to enhance performance
- As a class, email authors / poets
- Use different forms of communicating in role play – tablet, mobile etc
- Use email for Barnaby Bear to send emails home
- Compose a class blog on a diary of a tadpole
- Contact children in other class / school to request information
- Share similar work electronically with children in another class / school
- Use Skype to ask a visitor questions in preparation for the visit

Image, film & sound

- Use a painting app to create a picture
- Take photographs
- record an audio track
- Work with a simple animation app such as Puppet Pals or Shadow Puppets to tell a story
- Listen to pre-recorded sound
- Record and playback sounds (eg voices, instruments, sounds around them …)
- Begin to discuss the quality of their image and make decisions (e.g delete a blurred image) 2
- Explore a range of electronic music and sound devices including keyboards, tablets perhaps in Garage Band 2
- Begin to understand that music and sound can affect mood and atmosphere 2
- Use a time lapse animation app to record the growth of a seed
- Children use an iPad to film the acting out of story boards they have created
- Children photograph shapes in their local environment and use them in their work on shape
- Children sequence a timeline of images perhaps in Shadow Puppets
- Children photograph safety signs around the school put them in Book Creator and record warnings to match the pictures.
- Use an art app to explore techniques (e.g. patterning, tiling, stamping)
- Children use a painting app to create a design for Joseph's coat
- Children create T-shirt designs using a graphics program / app
- Children research "designs in nature" capturing images on an iPad. They could use the KaleidaCam app to create patterns of their own.
- Create the setting from a familiar traditional tale and provide costumes and props to encourage children to take on particular roles. Children record audio responses / conversations while in role.
- Children add sound effects to a poem to enhance performance
- Children record stories for others to listen to, perhaps in Seesaw, share with the class and comment
- Record sounds around the school and identify them
- Children photograph riding a bicycle and add a caption and/or voice recording in Book Creator
- Children use Garage Band to create sounds and simple musical phrases using the live loops grid
- Children add sound effect to a poem to enhance performance

Progression statements

Internet

Children's use of the internet should still be guided by adults. It is not advisable to allow children unsupervised access to search engines.

- Use appropriate buttons, menus and hyperlinks to navigate web sites for stored information.
- Enter text into a search engine to find specific given web sites
- Understand that different forms of information (text, images, video) exist and that some are more useful than others for specific purposes.
- Locate specific sites by typing a website address (URL) into the address bar in a web browser (Safari). 2
- Begin to develop key questions to help find information 2
- Be aware of responsible internet use and the school's acceptable use policy (see digital literacy strand) 2

At this stage children's use of the internet needs to be carefully guided and always supervised by adults.

Cross curricular application of skills

- Children talk about their use of a talking book (fiction or non-fiction)
- Compare, contrast and discuss a range of fantastic settings from a variety of paper and IT sources (films, paintings, picture books, photographs). Backgrounds in Puppet Pals perhaps.
- Children explore a given webpage to find out information about a topic.
- Children use the Barnaby Bear website to find out about his visits and how he travels

Data sorting

- Develop simple classification skills by carrying out simple sorting activities (probably away from the computer)
- Sort and classify a group of items by asking simple yes / no questions
- Talk about the different ways technology can be used to collect information, (e.g. camera, microphone, accelerometer) 2
- Understand that technology can be used to sort items and information 2
- Record how a minibeast looks, how they move, etc. Use this information to help in writing poems about minibeasts
- Answer a question by collecting and recording data in lists and tables; represent the data as block graphs or pictograms to show results; use technology to organise and present data
- Children create graphs linked to health and growth
- Children interpret a pictogram showing the types of houses people live in
- Children undertake a traffic survey and interpret a pictogram
- Children talk about images of toys now and then

Digital Literacy (Online Safety)

Self image | Relationships | Reputation | Bullying | Information | Wellbeing | Privacy | Ownership

KS1 Programme of study extract

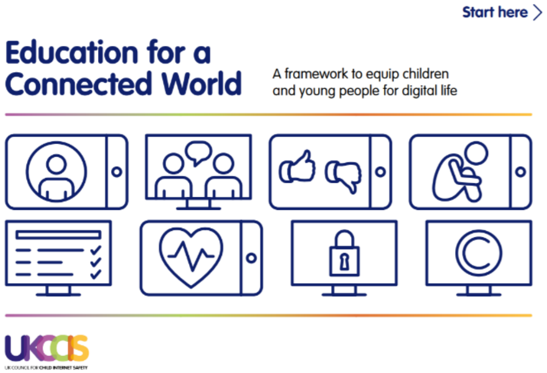
Pupils should be taught to:

- use technology safely and respectfully, keeping personal information private;
- identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies.

What does this mean?

Indeed! These are very short and simple statements but behind them there is a great deal to cover. Many ways have been suggested of organising learning (including one in earlier versions of this Progression), but one that is emerging, head and shoulders above others in recent months is Project Evolve.

Project Evolve is slowly providing a wealth of resources to support the statements produced in 2018 in Education for a Connected World. There are still gaps that are slowly being filled but there is a good structure and a splendid set of search tools to help you get to where you need to be.



Start here >

Education for a Connected World

A framework to equip children and young people for digital life

UKCCIS

How shall we teach it?

Ideally there should be two approaches to the delivery of this strand of Computing:

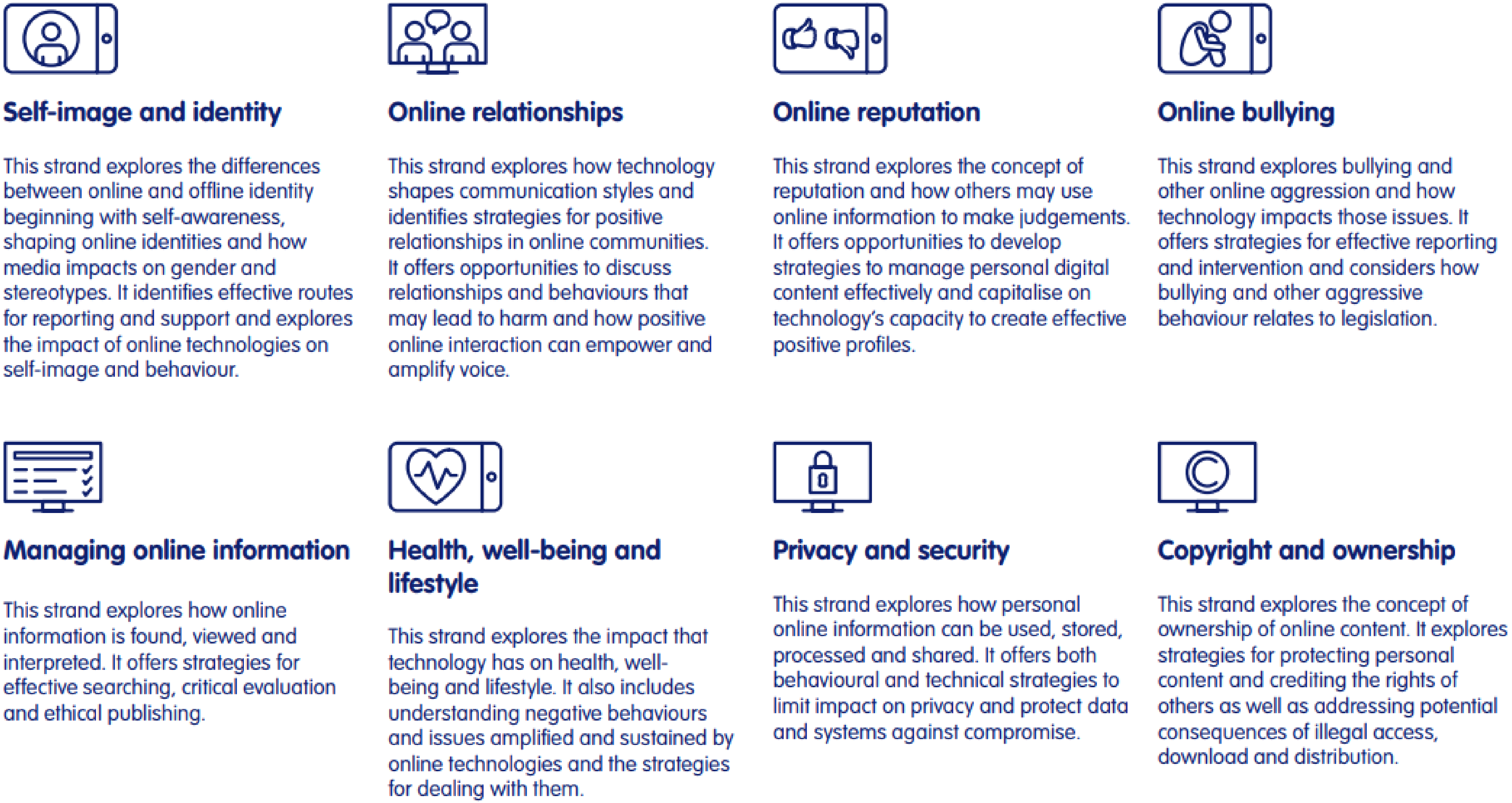
- Primarily, online safety needs to be planned and taught systematically, with appropriate links to PSHE and behaviour. The easy way to deal with this would be to teach a unit of online safety each year, possibly half a term. You could use any of the lessons from Project Evolve, additionally the table that follows includes more traditionally used resources that are tried and tested.
- Secondly, you will want to have conversations with your children and if appropriate even ad-hoc lessons, on particular aspects of online safety as they arise. This may be from incidents that have affected the children (at home or in school) or perhaps because there is a strong link to a particular aspect in other teaching. Again, you will find the resources in Project Evolve very useful.

ProjectEVOLVE

Shaping a better online life for all

What kit, software and resources are there to help me?

Project Evolve and Education for a Connected World organises learning around 8 strands with learning statements for each year group, starting in Early Years.



Self-image and identity

This strand explores the differences between online and offline identity beginning with self-awareness, shaping online identities and how media impacts on gender and stereotypes. It identifies effective routes for reporting and support and explores the impact of online technologies on self-image and behaviour.

Online relationships

This strand explores how technology shapes communication styles and identifies strategies for positive relationships in online communities. It offers opportunities to discuss relationships and behaviours that may lead to harm and how positive online interaction can empower and amplify voice.

Online reputation

This strand explores the concept of reputation and how others may use online information to make judgements. It offers opportunities to develop strategies to manage personal digital content effectively and capitalise on technology's capacity to create effective positive profiles.

Online bullying

This strand explores bullying and other online aggression and how technology impacts those issues. It offers strategies for effective reporting and intervention and considers how bullying and other aggressive behaviour relates to legislation.

Managing online information

This strand explores how online information is found, viewed and interpreted. It offers strategies for effective searching, critical evaluation and ethical publishing.

Health, well-being and lifestyle

This strand explores the impact that technology has on health, well-being and lifestyle. It also includes understanding negative behaviours and issues amplified and sustained by online technologies and the strategies for dealing with them.

Privacy and security

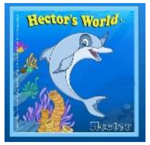
This strand explores how personal online information can be used, stored, processed and shared. It offers both behavioural and technical strategies to limit impact on privacy and protect data and systems against compromise.

Copyright and ownership

This strand explores the concept of ownership of online content. It explores strategies for protecting personal content and crediting the rights of others as well as addressing potential consequences of illegal access, download and distribution.

Older, trusted, resources for online safety

Lee and Kim's Adventure - 8 minute animation from CEOP introduces Lee & Kim, a brother and sister navigating the online world with the help of their trusted superhero friend SID! Teachers materials provide lesson plans



Hector's World - A tried and trusted resource from CEOP's Think U Know materials. An underground adventure for KS1. Hector and his friends are guided through potential traps in their under-water online world. 6 cartoons with complete lesson plans.

Cyber Café – CEOP's KS2 resources are based around their interactive Cyber Café. This is beginning to look a little "long in the tooth" but still very relevant. Teachers' resources centre around nine readymade lessons. You will need to create an account for your school or yourself (free) to these resources

Keeping Myself E-Safe – some resources from Learning Curve Education. A series of excellent animated real life stories packed with ideas. Supported by complete lesson plans and further background videos for teachers. We've paid for you to access these through WMnet but the best way to get them will be to download them from the Herefordshire My Learning site where you will find them in the *Herefordshire ICT Resources* course. You will need your school's login for this, let us know if you need a reminder.

Common Sense Media – Some very comprehensive materials from the USA. Complete lesson plans and full resources are provided. The South West Grid for Learning have produced their own *Digital Literacy and Citizenship* Scheme based on these resources and this is available free of charge.

Cyber Smart – Another complete scheme, this one from Australia. Some really useful material.

Some of the **ChildNet** and **Kid Smart** resources also contain whole lesson resources.

Education for a connected world Y1

1 Self-image and identity

- I can recognise that there may be people online who could make me feel sad, embarrassed or upset.
- If something happens that makes me feel sad, worried, uncomfortable or frightened I can give examples of when and how to speak to an adult I can trust.

2 Online relationships

- I can use the internet with adult support to communicate with people I know.
- I can explain why it is important to be considerate and kind to people online.

3 Online reputation

- I can recognise that information can stay online and could be copied.
- I can describe what information I should not put online without asking a trusted adult first

4 Online bullying

- I can describe how to behave online in ways that do not upset others and can give examples.

Education for a connected world Y2

- I can explain how other people's identity online can be different to their identity in real life.
- I can describe ways in which people might make themselves look different online.
- I can give examples of issues online that might make me feel sad, worried, uncomfortable or frightened; I can give examples of how I might get help.

- I can use the internet to communicate with people I don't know well (e.g. email a pen pal in another school / country).
- I can give examples of how I might use technology to communicate with others I don't know well.

- I can explain how information put online about me can last for a long time.
- I know who to talk to if I think someone has made a mistake about putting something online.

- I can give examples of bullying behaviour and how it could look online.
- I understand how bullying can make someone feel.
- I can talk about how someone can/would get help about being bullied online or offline.

Education for a connected world Y1	Education for a connected world Y2
5 Managing online information • I can use the internet to find things out. • I can use simple keywords in search engines • I can describe and demonstrate how to get help from a trusted adult or helpline if I find content that makes me feel sad, uncomfortable worried or frightened.	• I can use keywords in search engines. • I can demonstrate how to navigate a simple webpage to get to information I need (e.g. home, forward, back buttons; links, tabs and sections). • I can explain what voice activated searching is and how it might be used (e.g. Alexa, Google Now, Siri). • I can explain the difference between things that are imaginary, ‘made up’ or ‘make believe’ and things that are ‘true’ or ‘real’. • I can explain why some information I find online may not be true.
6 Health, wellbeing and lifestyle • I can explain rules to keep us safe when we are using technology both in and beyond the home. • I can give examples of some of these rules.	• I can explain simple guidance for using technology in different environments and settings. • I can say how those rules/guides can help me
7 Privacy and security • I can recognise more detailed examples of information that is personal to me (e.g. where I live, my family’s names). • I can explain why I should always ask a trusted adult before I share any information about myself online. • I can explain how passwords can be used to protect information and devices.	• I can describe how online information about me could be seen by others. • I can describe and explain some rules for keeping my information private. • I can explain what passwords are and can use passwords for my accounts and devices. • I can explain how many devices in my home could be connected to the internet and can list some of those devices.
8 Copyright and ownership • I can explain why work I create using technology belongs to me. • I can say why it belongs to me (eg ‘it is my idea’ or ‘I designed it’). • I can save my work so that others know it belongs to me (e.g. filename, name on content).	• I can describe why other people’s work belongs to them. • I can recognise that content on the internet may belong to other people.

Assessment

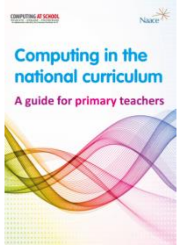
In every national curriculum programme of study the same statement on assessment is to be found:

> 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.

Further DfE guidance has made schools' responsibilities with regard to assessment clear (very much in line with previously):

> Schools will be able to introduce their own approaches to formative assessment, to support pupil attainment and progression. The assessment framework should be built into the school curriculum, so that schools can **check what pupils have learned and whether they are on track to meet expectations at the end of the key stage, and so that they can report regularly to parents.** Full DfE article here

Miles Berry, in his excellent publication *Computing in the national curriculum: A guide for primary teachers* (pp 22-25) makes some very helpful suggestions as to how we might approach formative and summative assessment. Briefly these are:



Self-assessment – where each child maintain a blog of their work; attaching examples and reflecting on learning
Peer-assessment –perhaps through shared blog entries and comments on peers' blogs
Open questioning – by teachers
Target setting - using KWL lists (what pupils already know, want to learn, what they have learned)

Were teacher assessment takes place, teachers will use their professional judgement to determine the most effective method of gathering evidence of progress but in computing it will certainly require knowledge of the context in which work was completed rather than simple scrutiny of a finished outcome.

One good approach is to consider, perhaps on an annual basis, what a child has accomplished for each of the strands (CS, IT, DL). Then take into account attainment across all aspects and adopt a "best fit" approach when arriving at an overall judgement.

Miles Berry offers a breakdown of the programme of study statements to create a hierarchy (or progression) of learning. An adapted version of his approach is below **in the simple assessment grid**. The 2014 version of these materials contained three more complex assessment grids which were not widely used.

SIMPLE ASSESSMENT GRID			
	COMPUTER SCIENCE	**INFORMATION TECHNOLOGY**	**DIGITAL LITEACY**
1	• Understand what algorithms are • Create simple programs • Understand that algorithms are implemented as programs on digital devices • Recognise common uses of information technology beyond school	• Use technology purposefully to create digital content • Use technology purposefully to store digital content • Use technology purposefully to retrieve digital content	• Use technology safely • Keep personal information private
2	• Understand that programs execute by following precise and unambiguous instructions • Debug simple programs • Use logical reasoning to predict the behaviour of simple programs	• Use technology purposefully to organise digital content • Use technology purposefully to manipulate digital content	• Use technology respectfully • Identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies
3	• Write programs that accomplish specific goals • Use sequence in programs • Work with various forms of input • Work with various forms of output	• Use search technologies effectively • Use a variety of software to accomplish given goals • Collect information Collect data • Design and create content • Present information	• Use technology responsibly • Identify a range of ways to report concerns about contact
4	• Design programs that accomplish specific goals • Design and create programs • Debug programs that accomplish specific goals • Use repetition in programs • Control or simulate physical systems • Use logical reasoning to detect and correct errors in programs • Appreciate how search results are selected	• Select a variety of software to accomplish given goals • Select, use and combine internet services • Analyse information • Evaluate information • Present data • Understand the opportunities computer networks including the internet offer for communication	• Identify a range of ways to report concerns about content • Recognise acceptable/unacceptable behaviour
5	• Solve problems by decomposing them into smaller parts • Use selection in programs • Work with variables • Use logical reasoning to explain how simple algorithms work • Use logical reasoning to detect and correct errors in algorithms • Understand how computer networks can provide multiple services, such as the World Wide Web	• Combine a variety of software to accomplish given goals • Select, use and combine software on a range of digital devices • Design and create systems • Analyse data	• Be discerning in evaluating digital content
6	• Understand computer networks, including the internet • Appreciate how search results are ranked	• Understand the opportunities computer networks including the internet offer for collaboration • Evaluate data	• Be discerning in evaluating digital content