Important notes

- Much of the work delivered in the 'old' ICT scheme is still valuable and should be continued to be taught
- The ideas discussed in the middle two pages are not exhaustive but give an understanding of what needs to be covered
- Each of the statements do not need the same teaching time. Some statements will only take a lesson or two others will need to be delivered as long term projects
- Computing needs to enable pupils to progress to further study and areas such as programming and how computers work are key to this
- All teachers of 'ICT' are able to teach computing—they just need training!

Assessment

- Levels no longer exist
- Each school will have to consider how they are going to assess across all subjects—the ICT/Computing department will have to follow this lead
- Many schools will continue to use 'national curriculum' level type structure—school will have to construct their own level criteria (or adopt someone else's) for the new topics in the PoS
- All assessment should emphasise progress—it should be formative not summative

Where to find resources

- Somerset eLIM Website http://bit.ly/elimks34
- Computing at Schools http://www.computingatschool.org.uk/
- TeachICT http://www.teach-ict.com/
- TES http://www.tes.co.uk/
- CSUnplugged http://csunplugged.org/resources-teachers
- CS4fn http://www.cs4fn.org/teachers/

Contact eLIM

The eLearning and information team at Somerset offer support for schools in their curriculum work.

their website can be found at	**www.somerstelim.org**
they can be contacted by email at	**elimadimin@somerset.gov.uk**
or by phoning	**01823 356832**
Ian Gover is the Education Technology Adviser for secondary schools and he can contacted at	**igover@somerset.gov.uk**
or by phoning	**07557 481372**

The Computing PoS at KS3 and what it means

In this booklet

Why Computing?

With the review of the national curriculum, the government decided that a move needed to be made towards having more programming and technology related learning in the curriculum.

They decided to change the subject name from ICT to Computing to emphasise this change.

Computing is a statutory subject at all key stages including for at ks4

Purpose of Study

The purpose of study states that computing equips pupils to 'to use computational thinking and creativity to understand and change the world.' providing insights 'into both natural and artificial systems'.

This is the core of the subject—it should get pupils to think about the world and the systems that exist.

It states that they should be taught some computing theory and how digital systems work.

They should be 'equipped to use information technology to create programs, systems and a range of content'.

Computing also ensures that pupils become digitally literate at a level suitable for the future workplace and as active participants in a digital world.

There are also elements of esafety.

Aims

There are four main aims related to the areas of:

- understanding and applying the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation. **Computer Science**
- analysing problems in computational terms, and repeat practical experiences of writing computer programs in order to solve such problems. **Computer Science**
- evaluating and applying information technology, including new or unfamiliar technologies, analytically to solve problems. **Information Technology**
- being responsible, competent, confident and creative users of information and communication technology. **Digital Learning**

It is important to note that there are no suggested learning hours for each aim.

an explanation of the words used in the programme of study

‘pupils should be taught to:’

design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems

Abstraction is one of the most terrifying words in the PoS. In fact it is quite easy to understand. An abstraction is just a representation of something. A circle is an abstraction of a wheel, a spreadsheet could be an abstraction of a financial budget, and a simulation could be an abstraction representing population growth.

The process of abstraction is simplifying and making an analysis to identify the essential features.

In teaching terms pupils should always be relating their work to real life. In making a spreadsheet they should model a household budget, in using a database they could copy what is happening for a firm supplying goods on a web site.

Pupils should also look at real world control systems, such as those used to control hydro electric power or in robotics.

understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem

An algorithm is a step by step guide to a process: A knitting pattern or a flowchart are examples.

Pupils should learn how to write simple algorithms or sequence of instructions. It does not have to be in a precise form and can just be a list of steps.

If the algorithm is developed into program like sentences it is then said to be pseudo-code.

There are different algorithms for sorting a set of numbers. Insertion, Selection, Quick and Bubble are types of sorts with plenty of teaching resources available. Look for the Hungarian folk dancers on YouTube!

When searching a large database, there are algorithms for speeding up the process. For example, rather than just starting with the first item and then looking at the second, a binary search compares the search term with the middle item of a sorted list. It then repeats the split until the term is found.

use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions ,

Pupils should have experience of two different languages. In most cases this will be Scratch and a text based language like Python, Small Basic or Java. The text based language cannot be HTML—that is coding.

Within these languages, pupils should meet problems that involve the use of a set of data. An example of this could be a program to select at random the name of a pupil from a class list.

They should also use the idea of a program being created in sections which include the idea of procedures or functions.

Pupils can be taught programming by looking at existing programs —they do not necessarily have to create them from new. Many Java programs are constructed in sections taken from an open source library.

Avoid using difficult maths as this is a barrier to learning.

understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal]

Part of the theory that pupils should meet revolves around how computers use electricity to work.

This centres around how they use binary – 0’s and 1’s, to represent numbers. Pupils should be shown how to convert from binary to ordinary numbers and vice versa.

They then should also be shown how binary numbers can be added together.

Electronic circuits in the computer combine binary digits using what are called logic gates. The gates are named after George Boole a famous mathematician. Three types of gate, AND, OR and NOT, process the digits in different ways. More advanced pupils could be shown how these gates combine to form a ‘half adder’.

understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems

Hardware
As well as being able to describe the major parts of a computer, pupils could experience a practical session, putting a system together either using an ‘old’ pc or raspberry pi.

Software
Pupils should learn the difference between the different types of software. They should also be taught how to keep software up to date and about virus protection.

It might be a good idea to explore the idea of open source software, comparing them with proprietary software.

Communication
Pupils should be taught how networks are linked together including LANs , WANs and the Internet. This work should include the need for protocols to allow all devices to talk to each other.

Pupils should also be given an understanding of wireless networks and how to link to these networks in a secure way.

understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits

This section deals with the theory around the storage of data and programs inside the computer.

With computers having to use binary they have to have program translated into very simple instructions. These simple instructions translate something such as INPUT into a code which the computer has been instructed to treat in a specific way. Creating a human computer, with individual pupils only being able to do one task, is a good way of teaching this topic.

All data inside the computer also has to be coded into binary digits. With characters this is completed using ASCII codes.

Pictures are split into dots, with each dot having numbers that represent attributes such as colour. This is a fun area to teach!

undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users

This is the statement that deals with using applications to create work.

Pupils should meet a range of applications including word processing, spreadsheet, database, graphic, sound and video editing. They can of course meet other programs with many schools, for example, using web2.0 apps to create mobile phone apps.

In each case pupils should design solutions for a purpose - meeting the needs of an end user.

create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability

An artefact in computing terms is something that someone has produced. This statement refers to creating or re-purposing the ‘something’ for a given audience.

Examples in this area could include producing a company logo that works across many forms of communication: PowerPoints, letters, or business cards or changing a Prezi presentation into one suitable for children.

understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.

This statement deals with esafety and must be part of a progressive curriculum for all pupils. Areas to be covered should include: security, safe behaviours, obsessive use technology, bullying, digital footprint, the reliability, validity of bias of websites and copyright.

The laws relating to the use of computers should also be introduced.