Computing Scheme

Date Policy Adopted: May 2017

	Date of Next Review by
Headteacher/SLT	May 2018
Approval	Curriculum and Standards Committee

Working with Ladygrove Primary School, William Reynolds Primary School & Nursery and Richard Smith from AmazingICT

A range of useful links can be found here
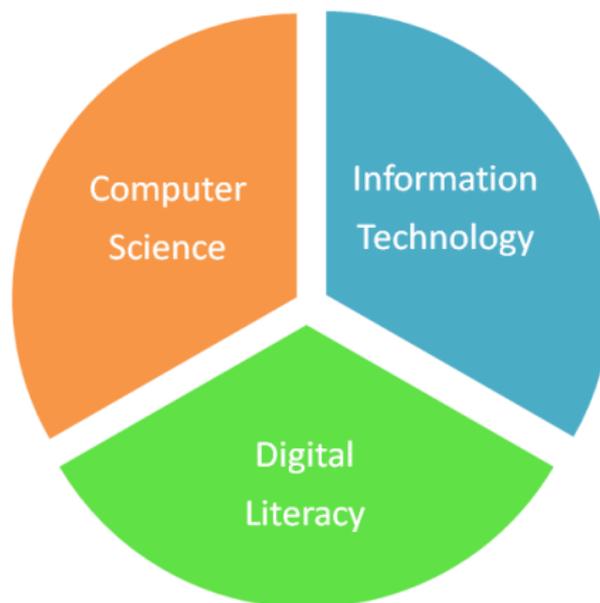
Computing Scheme of Work

What is Computing?

The subject of Computing replaces the previously named ICT. This is a move away from learning how to use technology and towards learning about **computers and computation**.
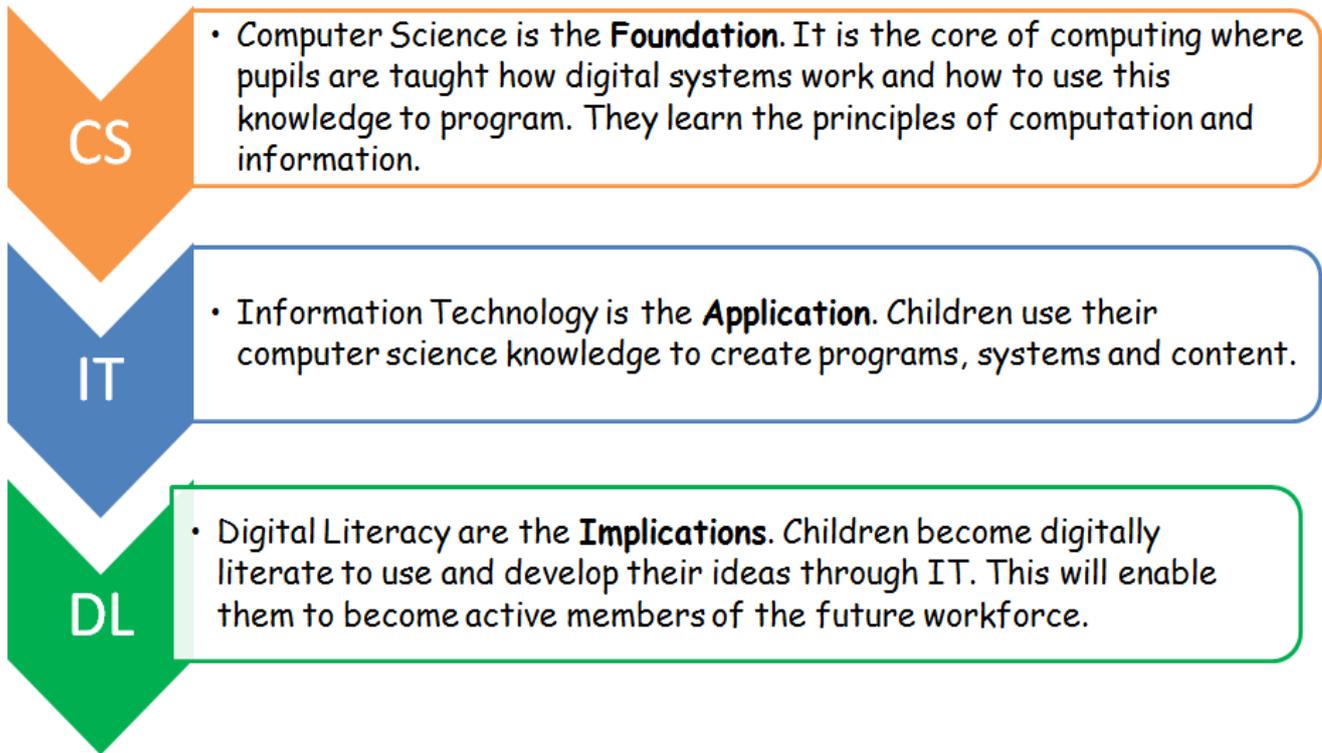
Computing is the understanding of how computer systems work, and how they are designed and programmed. This may or may not include computers as 'computational thinking' providing insights into many areas of the curriculum, and influences work at the cutting edge of a wide range of disciplines.' (Naace, Computing in the national curriculum)

At Broseley we believe that computational thinking develops important key skills including problem solving, logic, designing, collaboration, creativity and resourcefulness.

The Computing curriculum is made up of three main strands.

These strands are linked together.

CS • Computer Science is the **Foundation**. It is the core of computing where pupils are taught how digital systems work and how to use this knowledge to program. They learn the principles of computation and information.

IT • Information Technology is the **Application**. Children use their computer science knowledge to create programs, systems and content.

DL • Digital Literacy are the **Implications**. Children become digitally literate to use and develop their ideas through IT. This will enable them to become active members of the future workforce.

All of these three strands are of equal importance.

Computer Science

Pupils are taught the principles of

- how computer systems work **(input, process, output)**
- finding and fixing mistakes in a computer program (**Debugging**)
- using logical thinking to solve problems
- using step by step instructions to make something happen effectively (**Algorithm**) (could be away from the computer)
- a list of instructions that tells a computer exactly what to do **(Program)**

Information Technology

Pupils then have the knowledge to use IT to

- create presentations, documents and use data to convey meaning
- be creative to convey a message effectively
- store and manipulate content
- retrieve digital content

Digital Literacy

Pupils then become digitally literate so that they are

- prepared for the future phases of education and the workplace
- responsible and safe users of technology at school and at home
- can collaborate effectively

Subject Knowledge for the Key Stage 1
Curriculum

(Taken from Computing in the national curriculum. A guide for Primary Teachers 2013)

Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions

An **algorithm** is a precisely defined procedure –a sequence of instructions, or a set of rules, for performing a specific task (e.g. instructions for changing a wheel or making a sandwich). While all correct algorithms should produce the right answer, some algorithms are more efficient than others. Computer scientists are interested in finding better algorithms, partly out of intellectual curiosity, and partly because improvements in algorithms can result in massive savings in terms of both cost and time.

Computer programs, like algorithms, are comprised of sets of rules or instructions, but they differ in that they need to be written in a precise language a computer can 'understand'. A computer's central processor understands a very limited set of simple instructions written in machine code. Very few programmers work at this level, so computer scientists have developed programming languages, which sit somewhere between the ideas in the algorithm and the computer's machine code.

A programmer can turn an algorithm into code using a programming language that has enough in common with the English language to make it easy to read, remember and write. The programming language takes care of the minute details, like how to do multiplication or where **data** should be stored in the computer's memory, which means the programmer can focus on the big picture.

There are many different programming languages. They each have their own vocabulary, grammar and features that make them appropriate for particular tasks. The current favourites in primary schools are Scratch, Logo and Kodu.

Programs are made up of statements in a limited, but precisely understood, vocabulary. Each statement in the program has one particular meaning. The computer follows the instructions given: nothing more and, almost always, nothing less.

A 'computer' is not just a traditional desktop or laptop PC; it is any device that accepts **input**, processes it according to a stored program, and produces an **output**. The input, stored program and output are all encoded as numbers, making these devices 'digital'. Digital devices include the controller in your car or microwave oven, your mobile phone, tablet, laptop and desktop, as well as high-end supercomputers and 'virtual' servers in the 'cloud'.

Create and debug simple programs

The best way for pupils to learn what an algorithm is, and how it can be implemented as a program, is to write some programs themselves. Programming involves taking an idea for doing something and turning it into instructions the computer can understand. In the infant classroom this could be writing a set of commands for a Bee-Bot or ordering commands on daisy the dinosaur.

When you write a program you need to have a clear idea of what it will do and how it should do it. This is where algorithms come in, and thinking algorithmically is an integral part of the craft of programming.

Most programs don't work as they should first time round; professional programmers have this experience all the time! One of the most rewarding aspects of programming is finding and fixing these mistakes. Mistakes in programs are called 'bugs', and finding and fixing them is '**debugging**'.

The process of debugging often involves identifying that there is a fault, working out which bit of the program (or underlying algorithm) has caused the problem, and then thinking logically about how to fix it. In the classroom, this can provide a great opportunity for collaborative work.

As a teacher, you should identify clear steps that pupils can follow so that they can fix their code. These might involve identifying what the fault is, finding out which part of the code is creating the problem, and then working towards a fix.

Pupils should be encouraged to work together to identify bugs, as programmers are often blind to their own mistakes. Although it might be appropriate to help pupils compare code or identify which section to look at, it is rarely helpful for you to fix a bug for pupils until they have worked through the stages of debugging themselves.

Debugging code develops valuable learning skills that are transferable right across the curriculum, such as independence, resilience and persistence (linked to Guy Claxton's 'building learning power')

Use logical reasoning to predict the behaviour of simple programs

Computers are deterministic machines. We can **predict** exactly how they'll behave through repeated experience or by developing an internal model of how a piece of software works. Stepping through the program can give a clear sense of what it does, and how it does it, giving a feel for the algorithm that's been implemented.

In the classroom, getting one pupil to role-play a floor turtle or screen sprite while another steps through the program can give a far more immediate sense of what's going on. When working with a computer, encourage pupils to
make a **prediction** about what the program will do before they press return or click the button, and to explain their prediction logically; this is part of
computer science.

Logical reasoning also implies that pupils are following a set of rules when making predictions. Pupils who step outside the boundaries of these
rules are not using logical reasoning. A pupil who expects a bee-bot to jump doesn't understand the constraints of its programming language or
hardware.

	Programme of Study	Learning objectives	Key skills	Cross Curricular Ideas
Year 1 Computer Science	Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions	• To understand that devices respond to commands • To begin to understand how a computer processes instructions and commands (computational thinking) • To understand that they can programme a simple sequence of commands into a programmable robot or toy to send it on a route	❖ Explore a range of controllable toys and devices ❖ Begin to develop computational thinking by following instructions to move around a course and creating a series of instructions to move their peers around a course ❖ Explore outcomes when individual buttons are pressed on robots, such as floor turtles and combine these together to draw simple shapes or follow a route.	❖ Writing instructions (making a cake etc) ❖ Position and direction in Maths ❖ Exploring technological devices ❖ Creative writing linked to the objects being moved
	Create and debug simple programs	• To plan and write a simple program • To recognise that programs don't always work first time • To recognise a mistake and begin to fix it	❖ Turn an idea into a set of instructions for a device to follow ❖ Collaboratively identify an error and find what part of the code is creating the problem. ❖ Begin to think logically about how to fix (Debug) a program	❖ Problem solving (linked to debugging)
	Use logical reasoning to predict the behaviour of simple programs	• To use role play to predict how a program will work • To start to visualise if a program will work.	❖ Demonstrate logical reasoning ❖ Use directional and positional language to describe an object.	❖ Predicting skills in science
	Resources	Beebots, Beebot mats, beebot apps, A.L.E.X Cheese Sniffer http://www.iboard.co.uk/iwb/Cheese-Sniffer-657 **See 'Rising Stars', Switched On, Year 1 – 'We are Treasure Hunters' and 'We are Collectors' and 'We are TV chefs'.**		

	Programme of Study	Learning objectives	Resources

Year 1
Information Technology
	Use technology purposefully to **create**, **organise**, **store**, **manipulate** and **retrieve** digital content
	
	Teachers to link the program of study for computing with other curriculum areas and the half termly theme.
		• To retrieve files from a location on a network.
		• To store files in a location on a network.
		• To organise their own work in folders on a network.
		• To name stored work appropriately so that it can be retrieved and edited.
		
		We are Storytellers
		   • Use sound recording equipment to record sounds.
		   • Develop skills in saving and storing sounds on the computer.
		   • Develop collaboration skills as they work together in a group.
		   • Understand how a talking books differs from a paper- based book.
		   • Talk about and reflect on their use of ICT.
		   • Share recordings with an audience.
		We are celebrating
		   • Develop basic keyboard skills through typing and formatting text.
		   • Develop basic mouse skills.
		   • Use the web to find and select images.
		   • Develop skills in storing and retrieving files.
		   • Develop skills in combining text and images.
		   • Discuss their work and think about whether it can be improved.
		We are painters
		   • Use the web safely to find ideas for an illustration.
		   • Select and use appropriate painting tools to create and change images on the computer.
		   • Understand how this use of ICT differs from the use of paint and paper.
		   • Create an illustrations for a given purpose.
		   • Know how to save, retrieve and change their work.
		   • Reflect on work and act on feedback.
			All of the following suggestions can be accessed through 2simple infant video kit.
			
			• 2paint (picture creating)
			
			• 2publish and 2publish plus (word processing and publishing)
			
			• 2count (data presentation)
			
			• Revelation Natural Art
			
			• 2animate(simple mode)
			
			• 2create (simple presentations)

	Programme of Study	Learning objectives	Key skills	Resources
Year 1 **Digital Literacy**	Recognise common uses of information technology beyond school	• To understand that information comes from different sources e.g. books, web sites, TV etc • To understand that ICT can give access quickly to a wide variety of resources • To understand how digital technology is used at home and in work. • To be able to explore a variety of electronic information as part of a given topic	❖ Select appropriate buttons to navigate web sites or stored information ❖ Begin to understand that computers use icons, menus, hyperlinks to provide information and instructions ❖ Access different types of information from different sources ❖ Recognise digital technology used in everyday life	Think you know! Hectors World KidRex See medium term plans.
	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	• To understand that anyone can access the internet. • To understand that personal information should not be shared online • To understand what to do if you are worried	See medium term plans for series of 6 lessons.	

	Programme of Study	Learning objectives	Key skills	Cross curricular ideas
Year 2 Computer Science	Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions	• To understand that an algorithm is a sequence of instructions or step-by-step guide. • To understand that devices or on screen characters are controlled by algorithms, and that these can be inputted using icons and by text. • To create, edit and refine sequences of instructions for a variety of programmable devices.	❖ Further develop their understanding of computational thinking. ❖ Continue to explore floor turtles, combining sequences of instructions to follow a pattern or create a shape. ❖ Explore an on screen turtle navigate it around a course or grid and/or draw shapes by inputting a sequence of instructions. ❖ Begin to understand that the on screen turtle can be directed through the use of text.	❖ Moving around a grid ❖ Positional language in maths ❖
	Create and debug simple programs	• To plan and write a simple program • To recognise that programs don't always work first time • To identify a program's bugs and find a logical way of debugging.	❖ Turn an idea into a set of instructions for a device to follow ❖ Collaboratively identify an error and find what part of the code is creating the problem. ❖ To think logically about how to fix (Debug) a program	❖ Story creation for Scratch junior
	Use logical reasoning to predict the behaviour of simple programs	• To logically predict how a program will work	❖ Demonstrate logical reasoning	❖ Using the correct terminology such as predict and test
	RESOURCES	Beebot ipad apps Daisy the Dinosaur Scratch junior App **See 'Rising Stars', Switched On, Year 2 – 'We are Astronauts' and 'We are games testers'**		

	Programme of Study	Learning objectives	Resources
Year 2 **Information Technology**	Use technology purposefully to **create**, **organise**, **store**, **manipulate** and **retrieve** digital content Teachers to link the program of study for computing with other curriculum areas and the half termly theme.	• To retrieve files from a location on a network. • To store files in a location on a network. • To organise their own work in folders on a network. • To name stored work appropriately so that it can be retrieved and edited.	• 2paint, 2draw (picture creating), Revelation Natural Art.
		We are photographers. • To use a digital camera or camera app. • Review, edit and enhance photographs. • Create a portfolio of images.	• 2simple photo (simple mode) • 2animate (advanced mode)
		We are researchers. • To refine research skills. • To use mind mapping as a way of note taking. • To create a short multimedia presentation.	• 2connect (mind mapping) • 2create OR PowerPoint
		We are detectives. • To understand that email can be used to communicate. • To compose, open and send emails. • To open and listen to audio files. • To select and use appropriate language. • To edit text in emails.	• 2publish or 2publishplus (word processing and publishing), 2simple create a story
		We are zoologists. • To sort and classify groups of items. • Collect data with tick charts and tallies. • Create pictograms and basic charts. • Record data on a digital map.	• 2question (branching database) • 2graph (more detailed data presentation) • My World 3

	Programme of Study	Learning objectives	Key skills	Resources
Year 2 Digital Literacy	Recognise common uses of information technology beyond school	• To talk about the different forms of information (text, images, sound, multimodal) and understand some are more useful than others • To understand and talk about how the information can be used to answer specific questions • To understand how digital technology supports our lives at home. • To understand how digital technology supports professionals in work.	❖ Select appropriate buttons to navigate web sites or stored information ❖ Begin to understand that computers use icons, menus, hyperlinks to provide information and instructions ❖ Recognise digital technologies around the home and the part they play in our everyday lives. ❖ Recognise how digital technologies are part of the workplace and how they benefit businesses.	See medium term plans
	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 technology	• To understand that anyone can access the internet. • To understand that personal information should not be shared online • To be aware of age-appropriateness • To understand what to do if you are worried	See medium term plans for series of 6 lessons.	See medium term plans.

12

Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts

The focus on algorithms at key stage 1 leads pupils into the design stage of programming at key stage 2.

Algorithms are the necessary start of the process of creating working **code**, (blocks of code in scratch or kodu or text based code as used in secondary) and identifying the steps needed to solve any problem is essential.

Splitting problems into smaller parts is part of computational thinking. For example, designing a game in Scratch will involve thinking about

- algorithms,
- programming,
- drawing sprites and backgrounds,
- making animations,
- and even composing music or recording sound effects.

We think of computers as boxes with keyboards, mice and displays, but built-in computers (or 'embedded control systems') are an increasingly significant application of information technology. Pupils can gain valuable insights into how
computers are used to monitor and control real-world systems by using sensors, switches, motors and lights.

Computers also make it possible to explore real-world situations that would be too difficult, too expensive or too dangerous to create in real life.

Use sequence, selection, and repetition in programs; work with variables and various forms of input and output
Sequence in this context is the step-by-step nature of computer programs, mirroring the sequence of steps the algorithm would list.

Selection refers to instructions such as if ... then ...otherwise decisions in which the operation (what the program does) depends on whether or not
certain conditions are met. For example, a quiz provides different feedback if the player answers the question correctly or incorrectly. It is helpful
to refer pupils to selections (choices) they make in everyday life; for example, if it rains in the morning, then I will wear my anorak to school, otherwise I won't.
Repetition is a programming structure such as a repeat ... until loop in which the computer runs part of the program a certain number of times or until a particular condition is met.
In the case of the quiz, we might want to ask ten questions, or keep going until the player has scored five correct answers. Again, it is useful to refer pupils to loops or repetition in daily routines. For example, the traffic lights on a pelican crossing will stay green until someone presses the button to cross the road; an oven heats up until it reaches the right temperature. There are many loops in the wider world, such as the days of the week or the moon travelling around the Earth.

Variables are used to keep track of the things that can change while a program is running. They are a bit like x or y in algebra, in that the values may not initially be known. Variables are not just used for numbers. They can also hold text, including whole sentences ('strings'), or the logical values 'true' or 'false'. For our quiz we would use variables to keep track of the player's score and the number of questions they attempt. Variables are like boxes, in that the computer can use them to store information that can be changed by the user, the program or by another variable.
We may think of input as keyboard and mouse (or touch screen), and output as the computer display, but pupils' experiences should be widened beyond this. Working with sound is straightforward, as

laptops have built-in microphones and speakers. The latest version of Scratch provides support for using webcams.
Digital cameras allow interesting work using image files.
The reference in the programme of study to 'controlling physical systems' implies the use of sensors, motors and perhaps robotics.

Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs

Key stage 2 pupils should be able to explain the thinking behind their algorithms, talking through the steps and explaining <u>why</u> they've solved a
problem the way they have.

They also need to be able to look at a simple programming project and explain what's going on. This is made easier with languages like Scratch and Kodu which feature an on-screen sprite or turtle.

The immediate feedback helps pupils to understand and debug their programs. Pupils might also be expected to look at someone else's algorithm and explain how it does what it does.

Thinking through programs and algorithms helps develop pupils' abilities to think logically and algorithmically, which leads to planned debugging of code rather than just a trial-and-error approach.

	KS2 **Programme of Study**	**Learning objectives**	**Key skills**
Year 3 Computer Science	Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts	• To understand that an open ended problem can be broken up into smaller parts. • To describe and create an algorithm to achieve a specific outcome. • To recognise when to test a program and recognise when to debug it.	❖ To be able to break down an open ended problem into smaller (achievable) parts. ❖ Plan and enter a sequence of instructions specifying distance and turn. This could be done using a robot or programming software ❖ Test and improve (debug) programmed sequences ❖ Begin to type commands to achieve outcomes ❖ Explore outcomes when giving sequences of instructions.
	Use sequence, selection, and repetition in programs	• To understand that repeating instructions can save time and make a program simpler.	❖ Create a simple program using repeated instructions.
	Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs	• To explain how a program will work • To predict possible outcomes of a planned algorithm.	❖ Use logical reasoning to explain to others how a program will work. ❖ Talk about algorithms planned by others and identify any problems and the expected outcome
	RESOURCES	Scratch Junior APP **See 'Rising Stars', Switched On, Year 3 – 'We are Programmers' and 'We are bug fixers'**	

	Programme of Study	Learning objectives	Resources
Year 3 **Information Technology**	Use search technologies effectively	Refining search techniques • Not always the top item the most useful • Adverts can come to the top (sponsored) • Search could include 'ks2' to reduce number of items • Could use 'news' dropdown to get latest (eg volcanoes eruption)	• Different web browsers – Internet Explorer, Google Chrome OR Safari.
	Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information **Teachers to link the program of study for computing with other curriculum areas and the half termly theme.**	**We are presenters** • To frame shots and shoot live video. • To edit video by editing clips and adding narration. • To understand the qualities of effective video.	• Windows Movie Maker or similar (iMovie). • Different web browsers – Internet Explorer, Google Chrome OR Safari. • 2simple photo (advanced mode).
		We are network engineers • To understand the physical hardware connections that allow networks to work. • To understand the features of internet protocols. • To understand the diagnostic tools for investigating connections. • To have a basic understanding of how domain names are converted to IP addresses.	• Mind Manager Smart
		We are communicators • Develop a basic understanding of how email works. • To be aware of 'netiquette' and e-safety. • To work collaboratively with a remote partner. • To experience video conferencing.	• Skype/FaceTime
		We are opinion pollsters • To understand some elements of survey design. • To understand some ethical and legal aspects of online data collection. • Use the web to facilitate data collection. • Use charts to analyse and interpret data.	• Decisions3 (Black Cat) – Branching databases. • Survey Monkey or similar • Microsoft Excel or similar.

	Programme of Study	Learning objectives	Key skills
Year 3 Digital Literacy	Understand the opportunities networks offer for communication and collaboration	• To understand how to save and retrieve work on the Internet, the school network and a personal device. • To understand how to communicate with others online. • To understand that the World Wide Web is the part of the Internet that contains websites.	❖ Save work on the school network, on the Internet and on individual devices. ❖ Talk about the parts of a computer. ❖ Use appropriate tools to collaborate on-line.
	Be discerning in evaluating digital content	• To recognise search tools to find and use appropriate website • To understand that not all online images should be used	❖ Use search tools to find and use appropriate websites
	Use technology safely, respectfully and responsibly; recognise acceptable and unacceptable behaviour; identify a range of ways to report concerns about content and contact. online technologies	• To understand that anyone can access the internet. • To understand that personal information should not be shared online • To be aware of age-appropriateness • To understand what to do if you are worried	❖ Use simple search tools and find appropriate websites. ❖ Recognise that a password can be a safe way to protect your information ❖ Recognise appropriate age restriction symbols ❖ Understand and follow 'think then click' agreement
	RESOURCES	See medium term plans.	

17

	KS2 Programme of Study	Learning objectives	Key skills
Year 4 Computer Science	Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts	• To recognise an error in a program and debug it so that it works. • Understand that algorithms will help to solve and sequence more complex programs. • Understand that using algorithms will also help solve problems in other subjects • To recognise the need to test and retest whilst a program is being developed.	❖ Create an algorithm that contains step by step instructions to solve a problem. ❖ Use of algorithms to solve problems to work in Maths, Science and Design and Technology ❖ Testing a program throughout development ❖ Use logical thinking to solve an open ended problem by breaking it up into smaller parts
	Use sequence, selection, and repetition in programs.	• To understand that input is data that goes into a computer system. • To understand that output is information that comes out of a computer system • To understand that repeating instructions can save time and make a program simpler.	❖ List five types of input device e.g microphone ❖ List five types of output device e.g speaker ❖ Create a program using repeated instructions
	Use logical reasoning to explain how some simple algorithms work.	• To predict and explain how a program will work. • To simplify a program if necessary.	❖ To evaluate an existing program and explain to somebody else how it works. ❖ Talk about algorithms planned by others and identify any problems and the expected outcome
	RESOURCES	Scratch online Hopscotch **See 'Rising Stars', Switched On, Year 4 – 'We are software developers', 'We are HTML editors' and 'We are toy designers'**	

	Programme of Study	Learning objectives	RESOURCES
Year 4 Information Technology	Use search technologies effectively	Refining search techniques • Not always the top item the most useful • Adverts can come to the top (sponsored) • Search could include 'ks2' to reduce number of items • Could use 'news' dropdown to get latest (eg volcanoes eruption)	Thinking with pictures. Decisions3 (Black Cat) Photo story 3 2simple photos (advanced mode)
	Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information **Teachers to link the program of study for computing with other curriculum areas and the half termly theme.**	**We are musicians** • Use one or more programs to edit music • Create and develop a musical composition. • Use a composition to enhance work in other media.	Audacity
		We are co-authors • Understand the conventions for collaborative work (wikis) • Be aware of responsibilities when editing somebody else's work. • To know the problems with Wikipedia. • To refine and practise research skills. • To write for a target audience using a wiki tool.	Various web browsers Wikipedia app Wiki tools etc
		We are meteorologists • To understand different measurement techniques. • To use computer based data-loggers. • Use and create spreadsheets. • Analyse and explore data. • Present findings.	Dataharvesters Weather station (netatmo) app Excel OR similar (numbers app) PowerPoint OR similar (IWB Notebook)

19

	Programme of Study	Learning objectives	Key skills
Year 4 Digital Literacy	Understand the opportunities networks offer for communication and collaboration	• To recognise resources from the Internet, the school network or a personal device. • To understand that a hyperlink can take you directly to the world wide web	❖ Discuss and save work on the school network, on the Internet and on individual devices. ❖ Talk about the parts of a computer. ❖ Use appropriate tools to collaborate on-line.
	Be discerning in evaluating digital content	• To understand that information is not always reliable • To understand that key words can help to search safely on the internet • To understand that not all online images should be used • To understand when to use photographs, text and clipart	❖ Use various sources to find information and consider the reliability ❖ Recognise more reliable sources e.g bbc news ❖ Learn keywords to enable safe search e.g KS2 ❖ Identifying the owner before copying photos, clipart or text.
	Use technology safely, respectfully and responsibly; recognise acceptable and unacceptable behaviour; identify a range of ways to report concerns about content and contact. online technologies	• To understand that anyone can access the internet. • To understand that personal information should not be shared online • To be aware of age-appropriateness • To understand what to do if you are worried	❖ Use simple search tools and find appropriate websites. ❖ Recognise the risks of sharing information online ❖ Recognise appropriate age restriction symbols ❖ Understand and follow 'think then click' agreement
	RESOURCES	See medium term plans.	

	KS2 **Programme of Study**	**Learning objectives**	**Key skills**
Year 5/6 Computer Science	Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts.	• To recognise an error in a program and debug it so that it works. • Understand that algorithms will help to solve and sequence more complex programs. • Understand that using algorithms will also help solve problems in other subjects • To recognise the need to test and retest whilst a program is being developed.	❖ Create an algorithm that contains step by step instructions to solve a problem. ❖ Use of algorithms to solve problems to work in Maths, Science and Design and Technology ❖ Testing a program throughout development
	Use sequence, selection, and repetition in programs; work with variables and various forms of input and output	• To understand that input is data that goes into a computer system. • To understand that output is information that comes out of a computer system. • To understand that repeating instructions can save time and make a program simpler. • To understand that data can change in value (variable)	❖ Describe three types of input e.g audio, movement, touch (pressure), light, heat ❖ Describe three types of output e.g audio, text/images, printing including 3d printing, CAD cam ❖ Create a program using repeated instructions ❖ Create a program that uses a variable
	Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs	• To predict and explain how a program will work • To simplify a program if necessary • To correct a program if it doesn't work (debug)	❖ Use prior knowledge to make reasonable predictions about what will happen when a program is run. ❖ Evaluate algorithms planned by others and identify any problems and the expected outcome
	RESOURCES	Scratch (online) Hopscotch Kodu **See 'Rising Stars', Switched On, Year 5 – 'We are Cryptographers', 'We are Games Developers' and 'We are Web developers'**	

	Programme of Study	Learning objectives	Resources
Year 5/6 **Information Technology**	Use search technologies effectively	Students should be aware that everything on the internet is not necessarily true. Teachers to link the program of study for computing with other curriculum areas and the half termly theme.	Microsoft Thinking with pictures Movie Maker (iMovie) Photoshop Elements
	Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information **Teachers to link the program of study for computing with other curriculum areas and the half termly theme.**	**We are architects** • Understand the work of architects, designers and engineers working in 3D. • Develop familiarity with a simple CAD tool. • Develop spatial awareness by exploring and experimenting with a 3D virtual environment. • Develop greater aesthetic awareness. **We are artists** • Develop an understanding of the links between geometry and art. • Become familiar with the tools and techniques of a vector graphics package. • Develop an understanding of turtle graphics. • Experiment with tools available, refining and developing their work as they apply their own criteria to evaluate it and receive feedback from their peers. • Develop some awareness of computer-generated art, in particular fractal-based landscapes. **We are bloggers** • Become familiar with blogs as a medium and genre for writing. • Create a sequence of blog posts on a theme. • Incorporate additional media. • Comment on the posts of others. • Develop a critical, reflective view of a range of media, including text.	

	Programme of Study	Learning objectives	Key skills
Year 5/6 Digital Literacy	Understand the opportunities networks offer for communication and collaboration	• To recognise and name different parts of the Internet. • To understand the different parts of a webpage. • To understand that the internet can be used to communicate	❖ Identify different parts of the Internet. ❖ Choose appropriate tools for communication and collaboration and use them responsibly. ❖ Use effective strategies to search with appropriate search engines. ❖ Name different elements on web pages.
	Be discerning in evaluating digital content	• To understand and evaluate different types of information found on the World Wide Web. • To understand that not all online images should be used • To understand that the information on a webpage has been created by a person.	❖ Evaluate information on the world wide web. ❖ Use search tools to find appropriate websites ❖ Recognise more reliable sources e.g bbc news ❖ Identifying the owner of the information on a webpage.
	Use technology safely, respectfully and responsibly; recognise acceptable and unacceptable behaviour; identify a range of ways to report concerns about content and contact. online technologies	• To understand that personal information should not be posted or shared online • To understand the social etiquette required when communicating online. • To be aware of age-appropriateness • To understand that computer virus' can harm your computer. • To understand what to do if you see inappropriate content.	❖ Recognise the risks of sharing information, photographs online ❖ Know appropriate language when communicating online ❖ Recognise appropriate age restriction symbols ❖ Understand and follow 'think then click' agreement ❖ Understand reporting procedures
	RESOURCES	See medium term plans	

Computer Science- This focuses on how computer systems work and how they are programmed, this will usually present itself as specifically taught lesson.

Information Technology- This focuses on how technology can support learning across the curriculum and will be evident in all or most lessons.

Digital Literacy- This focuses on the positive and negative implications of using technology, E-awareness.

Glossary

algorithm – an unambiguous procedure or precise step-by-step guide to solve a problem or achieve a particular objective.

computer networks – the computers and the connecting hardware (wifi access points, cables, fibres, switches and routers) that make it possible to transfer data using an agreed method ('protocol').

control – using computers to move or otherwise change 'physical' systems. The computer can be hidden inside the system or connected to it.

data – a structured set of numbers, representing digitised text, images, sound or video, which can be processed or transmitted by a computer.

debug – to detect and correct the errors in a computer program.

digital content – any media created, edited or viewed on a computer, such as text (including the hypertext of a web page), images, sound, video (including animation), or virtual environments, and combinations of these (i.e. multimedia).

information – the meaning or interpretation given to a set of data by its users, or which results from data being processed.

input – data that goes into a computer, such as via a keyboard, mouse, microphone, camera or physical sensors.

internet – the global collection of computer networks and their connections, all using shared protocols (TCP/IP) to communicate.

logical reasoning – a systematic approach to solving problems or deducing information using a set of universally applicable and totally reliable rules.

output – the information produced by a computer system for its user, typically on a screen, through speakers or on a printer, but possibly though the control of motors in physical systems.

program – a stored set of instructions encoded in a language understood by the computer that does some form of computation, processing input and/or stored data to generate output.

repetition – a programming construct in which one or more instructions are repeated, perhaps a certain number of times, until a condition is satisfied or until the program is stopped.

search – to identify data that satisfies one or more conditions, such as web pages containing supplied keywords, or files on a computer with certain properties.

selection – a programming construct in which the instructions that are executed are determined by whether a particular condition is met.

sequence – to place programming instructions in order, with each executed one after the other.

services – programs running on computers, typically those connected to the internet, which provide functionality in response to requests; for example, to transmit a web page, deliver an email or allow a text, voice or video conversation.

simulation – using a computer to model the state and behaviour of real-world (or imaginary) systems, including physical and social systems; an integral part of most computer games.

software – computer programs, including both application software (such as office programs, web browsers, media editors and games) and the computer operating system. The term also applies to 'apps' running on mobile devices and to web-based services.

variables – a way in which computer programs can store, retrieve or change simple data, such as a score, the time left, or the user's name.

World Wide Web – a service provided by computers connected to the internet (web servers), in which pages of hypertext (web pages)
are transmitted to users; the pages typically include links to other web pages and may be generated by programs automatically.