

		<b>Key Stage 1</b>	
<b>National Curriculum</b>		<ul style="list-style-type: none"> <li><b>Understand what algorithms are; how they are implemented as programs on digital devices and that programs execute by following precise and unambiguous instructions</b> <i>An algorithm is a precise list of instructions on how to perform an action. In computing terms, it is instructions for what the computer will be programmed to do. Algorithms might even be written in plain English, before translating them into code that the computer will understand. Digital devices are any types of computers that you use, including laptops, tablets and smart phones. This may also includes hardware which may connect to a computer.</i></li> <li><b>Create and debug simple programs</b> <i>Debug - Errors in programs, or anything that stops them from working properly, are known as bugs. To debug means to fix or get rid of the bugs and solve problems within a program in order to make it work how it is intended. Mistakes are a normal, common part of programming and every computer programmer should get used to the fun of debugging! A program (a piece of code) is needed to tell a system what to do.</i></li> <li><b>use technology purposefully to create, organise, store, manipulate and retrieve digital content</b> <i>Digital content means any information that is stored or presented on computers or the Internet. Everything you create on the computer becomes digital content. This includes files on your computer, network or on the World Wide Web. Children need to start taking certain factors into consideration, such as where the digital content has come from and who has made it.</i></li> <li><b>use logical reasoning to predict the behaviour of simple programs</b> <i>Logical reasoning means thinking logically or systematically to solve problems. The best way to understand what a program does or solve errors in a program is to think through sensibly what is supposed to happen.</i></li> <li><b>Recognise common uses of information technology beyond school</b></li> <li><b>use technology safely, respectfully and responsibly; 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 on-line technologies</b></li> </ul>	
		<b>Year 1</b>	<b>Year 2</b>
<b>Computer Science</b>	<b>Programming and Coding</b>	<ul style="list-style-type: none"> <li>Give simple instructions o everyday devices to make things happen</li> <li>Make choices to control simple models or simulations</li> <li>Solve a problem using ICT</li> <li>Understand what an algorithm is (a sequence of instructions or set of rules for performing a specific task) (out of the context of programming).</li> <li>Understand that algorithms need to be precise, simple, clear and limited.</li> <li>Understand that an algorithm is implemented as program on a digital device.</li> <li>Input simple instructions (into programmable device or coding program) to see what happens.</li> <li>write/input a simple a program/code (no desired outcome)</li> </ul> <p><b>Bee-Bots</b></p> <ul style="list-style-type: none"> <li>Give commands including straight forwards/backwards/turn one at a time</li> <li>Explore what happens when a sequence of instructions is given</li> <li>Give a set of simple instructions to follow a task</li> <li>Give a set of instructions to form simple geometric shapes</li> <li>Improve/change their sequence of commands</li> </ul>	<ul style="list-style-type: none"> <li>Understand what algorithms are, how they are implemented as programs on digital devices and that programs execute by following a sequence of instructions</li> <li>Use logical reasoning to predict the behaviour of simple programs</li> <li>follow and predict the outcome of an program</li> <li>write/input and test a <u>simple</u> a program/ code to achieve a desired outcome (ensuring it is precise, simple, clear and limited)</li> <li>identify a bug in my programme/code (where the algorithm has gone wrong/not achieved the desired outcome)</li> <li>debug a program (fix it by changing algorithm)</li> </ul> <p><b>Bee-Bots</b></p> <ul style="list-style-type: none"> <li>Give commands including straight forwards/backwards/turn one at a time</li> <li>Explore what happens when a sequence of instructions is given</li> <li>Give a set of simple instructions to follow a task</li> <li>Give a set of instructions to form simple geometric shapes</li> <li>Improve/change their sequence of commands</li> </ul>

<p style="writing-mode: vertical-rl; transform: rotate(180deg);"><b>Ideas/links to activities</b></p>	<p><a href="http://www.bbc.co.uk/schoolscomputing">www.bbc.co.uk/schoolscomputing</a>                  Could begin computing sessions with a video clip from the following link then implement relevant activities:  <a href="http://www.bbc.co.uk/education/topics/z3tbwmn">http://www.bbc.co.uk/education/topics/z3tbwmn</a></p> <ul style="list-style-type: none"> <li>-What is an algorithm?</li> <li>-What is a code?</li> <li>-What are computer bugs?</li> <li>-How do you program a robot?</li> <li>-How do computer games work?</li> </ul> <p><b>Algorithms:</b></p> <ul style="list-style-type: none"> <li>➤ Teacher-bot <b>Ap.1</b></li> <li>➤ 'Nightmare' <b>Ap.2</b></li> <li>➤ Play/experiment with various robots and coding programs</li> </ul> <p><b>Programming:</b></p> <ul style="list-style-type: none"> <li>➤ Bee-bot App <b>Yr1/2</b></li> <li>➤ Bee-bot <b>Yr1</b> /Pro-bot <b>Yr2 Ap.3</b></li> </ul> <p><b>Coding:</b></p> <ul style="list-style-type: none"> <li>➤ Espresso coding <b>Yr1/2</b></li> <li>➤ Tynker App iPad <b>Yr1/2</b></li> <li>➤ Cargo-bot App iPad <b>Yr 2 HAPs</b></li> <li>➤ Hopscotch App iPad <b>Yr1/2</b></li> <li>➤ Scratch Jr. <b>Yr1/2</b></li> </ul>	
	<p style="writing-mode: vertical-rl; transform: rotate(180deg);"><b>Networks</b></p>	<ul style="list-style-type: none"> <li>• Discuss and share how and when they use ICT in everyday life</li> <li>• Complete simple tasks on a computer by following instructions</li> </ul> <p><b>Internet Research:</b></p> <ul style="list-style-type: none"> <li>• Talk about websites they have been on</li> <li>• Explore a website by clicking on the arrows, menus and hyperlinks</li> </ul> <p><b>Emails:</b></p> <ul style="list-style-type: none"> <li>• Recognise an email address</li> <li>• Find the @ key on the keyboard</li> <li>• Contribute to a class email</li> <li>• Open and select reply to an email as a class</li> </ul>
<p style="writing-mode: vertical-rl; transform: rotate(180deg);"><b>Ideas/ resources</b></p>		<p><i>Twinkl have lesson plans and presentations which are useful</i>  <i>BBC Bitesize</i>  <i>Code-it.co.uk [Networks]</i>  <i>Bearfoot Computing</i>  <i>Tocomail.com (email for kids)</i></p>

<b>Information Technology</b>	<b>Digital Literacy</b>	<ul style="list-style-type: none"> <li>• Operate a range of hardware including: cameras, video cameras, sound recording devices, tablets, keyboard, mouse etc.</li> <li>• Perform basic operations on a digital device (this includes PC's, tablets, cameras, robots etc.) e.g. switch on/off, log on/off, open/close programmes and apps, enter text etc.</li> <li>• Perform basic operations in a range of programmes/apps.</li> <li>• Create digital content, using a range of devices, including: word-processing, presentation software, paint packages, digital images and videos, computer programmes, online content (blog posts, social network updates, forum contributions and wiki entries) etc.</li> <li>• Edit content in a range of programmes/apps e.g. format text, insert images, add transitions in presentation software, edit photographs, use different tools in paint packages, send and open emails etc.</li> <li>• Combine digital content from multiple sources.</li> </ul> <p><b>Graphics:</b></p> <ul style="list-style-type: none"> <li>• Use ICT to generate ideas for their work</li> <li>• Use various tools such as brushes, pens, rubber, stamps, shapes.</li> <li>• Save and print work</li> </ul> <p><b>Text/ Word Processing:</b></p> <ul style="list-style-type: none"> <li>• Use spacebar, back space, delete, and return.</li> <li>• Start to use two hands when typing.</li> <li>• Word process a sentence/caption to present.</li> </ul> <p><b>Sound Recording:</b></p> <ul style="list-style-type: none"> <li>• Record sound at and away from the computer.</li> <li>• Use software to record sounds</li> </ul> <p><b>Video:</b></p> <ul style="list-style-type: none"> <li>• Capture video</li> <li>• Discuss which videos to keep and which to delete</li> </ul>	<ul style="list-style-type: none"> <li>• Edit content in a range of programmes/apps e.g. format text, insert images, add transitions in presentation software, edit photographs, use different tools in paint packages, send and open emails etc.</li> <li>• Combine digital content from multiple sources.</li> <li>• Make changes to digital content for an audience and purpose taking into account principles of good design</li> </ul> <p><b>Graphics:</b></p> <ul style="list-style-type: none"> <li>• Use ICT to generate ideas for their work</li> <li>• Use various tools such as brushes, pens, rubber, stamps, shapes.</li> <li>• Save, retrieve and print work</li> </ul> <p><b>Text/ Word Processing:</b></p> <ul style="list-style-type: none"> <li>• Use spacebar, back space, delete, arrow keys, return, shift.</li> <li>• Start to use two hands when typing.</li> <li>• Word process short texts to present.</li> <li>• Change text size and font.</li> <li>• Use word art for effect</li> </ul> <p><b>Sound Recording:</b></p> <ul style="list-style-type: none"> <li>• Record sound at and away from the computer.</li> <li>• Use software to record sounds</li> <li>• Change sounds recorded</li> <li>• Save, retrieve and edit sounds</li> </ul> <p><b>Video:</b></p> <ul style="list-style-type: none"> <li>• Capture video</li> <li>• Discuss which videos to keep and which to delete</li> <li>• Arrange clips to create a short film</li> <li>• Add a title and credits</li> </ul> <p><b>Presentation (Powerpoint)</b></p> <ul style="list-style-type: none"> <li>• Create a title slide and choose a style</li> <li>• Insert a picture/text/graph from the internet or personal files</li> <li>• Add text</li> <li>• Decide upon and use effective transitions</li> <li>• Present to the class</li> </ul>
-------------------------------	-------------------------	---	---

	<b>Ideas/ Resources</b>	<p><b>NOTE:</b>  <b>Information Technology should be incorporated into other subjects using a variety of hardware and software where appropriate to support learning.</b>  <b>Specific skills can be taught during ‘Computing’ sessions e.g. how to log on/off, how to open and save files, how to add a ‘transition’ to movie maker etc.</b>  <b>Digital content e.g. images and text can be combined in various programmes/apps, being obtained from numerous sources e.g. using a search engine or camera for images and Microsoft Word or Notes etc. for text.</b></p> <p><b>Resources:</b></p> <table border="1" style="width: 100%;"> <tr> <td style="width: 50%; vertical-align: top;"> <p><b>Hardware:</b>  <i>Cameras</i>  <i>video cameras</i>  <i>sound recording devices e.g. talking tins</i>  <i>tablets</i>  <i>keyboard</i></p> </td> <td style="width: 50%; vertical-align: top;"> <p><b>Software:</b>  <i>Microsoft office: word, power point, movie maker etc.</i>  <i>Paint packages</i>  <i>Search engines: Google/Bing etc. plus images</i>  <i>Blogging</i>  <i>Social networks: twitter, Facebook</i>  <i>Emailing</i>  <i>ShowMe (iPad)</i>  <i>PicCollage (iPad)</i>  <i>Purplemash</i>  <i>Digital-literacy.org.uk [this has digital literacy resources by year group]</i>  <i>i-movie</i>  <i>photostory</i>  <i>Paint</i>  <i>Photoshop</i></p> </td> </tr> </table>	<p><b>Hardware:</b>  <i>Cameras</i>  <i>video cameras</i>  <i>sound recording devices e.g. talking tins</i>  <i>tablets</i>  <i>keyboard</i></p>	<p><b>Software:</b>  <i>Microsoft office: word, power point, movie maker etc.</i>  <i>Paint packages</i>  <i>Search engines: Google/Bing etc. plus images</i>  <i>Blogging</i>  <i>Social networks: twitter, Facebook</i>  <i>Emailing</i>  <i>ShowMe (iPad)</i>  <i>PicCollage (iPad)</i>  <i>Purplemash</i>  <i>Digital-literacy.org.uk [this has digital literacy resources by year group]</i>  <i>i-movie</i>  <i>photostory</i>  <i>Paint</i>  <i>Photoshop</i></p>	
<p><b>Hardware:</b>  <i>Cameras</i>  <i>video cameras</i>  <i>sound recording devices e.g. talking tins</i>  <i>tablets</i>  <i>keyboard</i></p>	<p><b>Software:</b>  <i>Microsoft office: word, power point, movie maker etc.</i>  <i>Paint packages</i>  <i>Search engines: Google/Bing etc. plus images</i>  <i>Blogging</i>  <i>Social networks: twitter, Facebook</i>  <i>Emailing</i>  <i>ShowMe (iPad)</i>  <i>PicCollage (iPad)</i>  <i>Purplemash</i>  <i>Digital-literacy.org.uk [this has digital literacy resources by year group]</i>  <i>i-movie</i>  <i>photostory</i>  <i>Paint</i>  <i>Photoshop</i></p>				
	<b>Data</b>	<ul style="list-style-type: none"> <li>• Know that images give information</li> <li>• Say what a pictogram is showing them</li> <li>• Put data into a program</li> <li>• Sort objects and pictures into simple lists or simple tables</li> </ul>	<ul style="list-style-type: none"> <li>• Know that images give information</li> <li>• Say what a pictogram is showing them</li> <li>• Put data into a program</li> <li>• Sort objects and pictures into simple lists or simple tables</li> <li>• Make a simple Y/N tree diagram to sort information</li> <li>• Create and search a branching database</li> </ul>		
	<b>Ideas/ Resources</b>	<p><i>Purplemash?</i>  <i>Microsoft Office</i>  <i>Survey Monkey</i>  <i>Branching Data</i></p>	<p><i>Purplemash?</i>  <i>Microsoft Office</i>  <i>Survey Monkey</i>  <i>Branching Data</i></p>		
<b>E Safety</b>		<p><b>NOTE:</b>  <b>Understanding technology should be incorporated into other subject areas: e.g. science, electricity, robots and machines etc.</b>  <b>E-safety should be incorporated in <i>all</i> sessions/subjects when using information technology.</b></p> <ul style="list-style-type: none"> <li>• Be aware of the main risk associated with the internet.</li> <li>• Recognise that they should not share certain types of personal information online.</li> <li>• Have a clear understanding of what to do if they have concerns about inappropriate behaviour online.</li> <li>• Identify devices that can be used to search the internet</li> <li>• Understand rules around e-safety</li> <li>• Make decisions about whether or not statements found on the internet are true or not</li> <li>• Identify what things count as personal information</li> <li>• Identify when inappropriate content is accessed and know how to act appropriately</li> <li>• Consider other people’s feelings on the internet</li> </ul>	<p><b>NOTE:</b>  <b>Understanding technology should be incorporated into other subject areas: e.g. science, electricity, robots and machines etc.</b>  <b>E-safety should be incorporated in <i>all</i> sessions/subjects when using information technology.</b></p> <ul style="list-style-type: none"> <li>• Develop sensitivity to others online, treating them with respect and showing respect for their privacy.</li> <li>• Know how to report a worry and talk to teachers or parents about any concerns they have.</li> <li>• Identify devices that can be used to search the internet</li> <li>• Identify obviously fake information in a variety of contexts.</li> <li>• Identify personal information that should be kept private</li> <li>• Understand rules around e-safety</li> <li>• Identify when inappropriate content is accessed</li> <li>• Recognise that a variety of devices can be used to connect to a number of people</li> <li>• Consider other people’s feelings on the internet</li> </ul>		

<b>Ideas/ Resources</b>	<p><b>Resources:</b>                  CEOP: <a href="http://www.ceop.police.uk">www.ceop.police.uk</a>                  Think you Know: <a href="http://www.thinkuknow.co.uk">www.thinkuknow.co.uk</a>                  See: E-safety folder in G drive.<i>Twinkl</i>  <i>Hectors World</i>  <i>kidsmart.org</i>  <i>Kim and Lee</i>  <i>Thinkyouknow.co.uk (CEOP)</i>  <i>saferinternet.org</i></p>	<p><b>Resources:</b>                  CEOP: <a href="http://www.ceop.police.uk">www.ceop.police.uk</a>                  Think you Know: <a href="http://www.thinkuknow.co.uk">www.thinkuknow.co.uk</a>                  See: E-safety folder in G drive.<i>Twinkl</i>  <i>Hectors World</i>  <i>kidsmart.org</i>  <i>Kim and Lee</i>  <i>Thinkyouknow.co.uk (CEOP)</i>  <i>saferinternet.org</i></p>
-----------------------------	--	--

**Appendices:**

Activity		Resources
<p><b>Ap. 1:</b> Teacher-bot</p>	<p><b>Related Learning Objectives:</b></p> <ul style="list-style-type: none"> <li>- Understand what algorithms are</li> <li>- Understand that an algorithm is implemented as program on a digital device.</li> </ul>	<p><a href="https://www.youtube.com/watch?v=leBEFaVHlIE">https://www.youtube.com/watch?v=leBEFaVHlIE</a></p>
<p>See: <a href="https://www.youtube.com/watch?v=leBEFaVHlIE">https://www.youtube.com/watch?v=leBEFaVHlIE</a>                  Teacher to pretend to be a robot to make sandwich (or anything which requires specific instructions).                  Ask children to write or discuss with friend how they will tell teacher-bot how to make the sandwich. Children to write or verbally give instructions to teacher bot. Teacher-bot to following exact commands- this should go wrong!</p> <p>Discuss what happened: <i>What did/didn't teacher bot understand? What went wrong/right? Why did this happen? Did teacher-bot understand your language? What do we need to do to ensure teacher bot doesn't make such a mess next time?!</i>                  Explain that teacher-bot needs precise, clear, step-by-step instructions that he/she can understand and we call these 'algorithms'.</p> <p><b>Next step-</b> Transfer knowledge to computer programming:                  Explain that computers (these being anything from the computer in a washing machine to a laptop*) need very precise algorithms to <b>operate</b>/produce an <b>output</b> (do what we want it to!).                  In a computer <b>algorithms</b> are <b>encoded</b> in a language that the computer understands- which we call a <b>programme!</b>                  - Also see Teacher-Bot 2/Friend-Bot activity.</p> <p><b>*Note:</b> you may need to cover <b>LO: 'Recognise and discuss common uses of information technology beyond school'</b> (Digital Literacy- Understanding Technology) beforehand, i.e. understanding what a computer is- 'A device that accepts <b>input</b>, processes it according to a stored program, and produces an <b>output</b>' (a machine that follows precise instructions to do something we want it to! These can also be called <b>digital devices!</b></p>		
<p><b>Ap.2:</b> 'Nightmare'</p>	<p><b>Related Learning Objective:</b></p> <ul style="list-style-type: none"> <li>- Understand that algorithms need to be precise, simple, clear and limited.</li> </ul>	
<p>See: <a href="http://www.youtube.com/watch?v=jls0kCeadQs">http://www.youtube.com/watch?v=jls0kCeadQs</a>  <b>Note:</b> You may wish to change the name of this activity due to sensitivity of children or to fit in with current topic e.g. 'Buried Treasure' for pirates</p> <p>Initially chose one child to wear the helmet or blind fold. Set up some sort of obstacles for child to work move around. Ask children to working pairs/group to discuss or write down instructions in steps in order to direct the child to a certain point, using only the following: forwards, backwards, turn left/right and number of steps (as the child will only understand this language!!) Explain that we can call these instructions an algorithm. Could have cards for children to order instead of writing. Work as a class: choose one group to read out algorithm as blindfolded child follows (one step at time).</p> <p>Discuss outcome (key questions):  <i>Did this algorithm work? What went well/wrong? Why did this happen? Was the algorithm precise/simple/clear enough? Did we use any language the child didn't understand? What do we need to change to make our algorithm work better/achieve the correct outcome?</i> Point out that they need to ensure they make their algorithm very simple, clear and precise.                  Children could then work in groups to write an algorithm for their own obstacles.</p> <p><b>Differentiation:</b> LAPs- work in pairs blind folded, one child to give algorithm verbally, one step at a time to get to a certain location (no recording needed). Discuss: <i>Did anything go wrong? Why? What do you need to do next time?</i> Point out that they need to ensure they make their algorithm very simple and clear.</p> <p><b>Next Steps:</b>  <b>LO: Identify a bug in my programme/code</b>  <b>Debug a program</b>                  Children to identify errors in their algorithm (detect a bug) and correct (debug) it.</p>		

<p><b>Ap.3:</b> <b>Bee-bots/Pro-bots</b></p>	<p><b>Related Learning objectives:</b></p> <ul style="list-style-type: none"> <li>- Experiment with various forms of programming</li> <li>- follow and predict the outcome of an program</li> <li>- Input simple instructions to see what happens.</li> <li>- write/input a simple a program/code (no desired outcome)</li> <li>- write/input a simple a program/ code to achieve a desired outcome (ensuring it is precise, simple, clear and limited)</li> <li>- identify a bug in my programme/code (where the algorithm has gone wrong/not achieved the desired outcome)</li> <li>- debug a program (fix it by changing algorithm)</li> </ul>	<p>Bee-bots (12) Pro-bots (6) Mats. Bee-bot cards with simple visual algorithms.</p>
<p>The following computing <b>LO's</b> can be covered by the use of bee-bots/pro-bots (and other robots!):</p> <p>Children need to learn how to <b>operate</b> the robots, understanding that when they press the buttons on the robots they are <b>inputting</b> a programme and that the movement it makes is the <b>output</b>. The <b>output</b> of the programme they <b>input</b> using a pro-bot can also be the marks that it makes with the pen attachment.</p> <ul style="list-style-type: none"> <li>➤ Initially children need to play with the robots to familiarise themselves.</li> <li>➤ They could then use obstacles in the classroom or the bee-bot mats to navigate their way around.</li> <li>➤ They then need to identify where/if the program they entered went wrong- explain that this is called a <b>bug</b>.</li> <li>➤ They then need to correct the programme by <b>debugging</b> it.</li> <li>➤ Children could be given an <b>algorithm</b> to follow- verbally or using visuals.</li> <li>➤ Children could also write their own <b>program</b> for a friend to follow (visual cards could be used to order instead of writing)</li> <li>➤ Before entering an algorithm children could try to predict where the robot will end up and whether the algorithm will work/ achieve <b>desired outcome</b>.</li> </ul> <p><b>Note:</b> All of the above is not intended to be covered in one session!!</p> <p><b>Cross Curricular links:</b> The above learning intentions could be integrated into other lessons e.g. in a Maths lesson using the shape mat, in a Geography lesson looking at maps and routes etc.</p>		
<p><b>Coding Games</b></p>	<p><b>Related Learning Objectives:</b></p> <ul style="list-style-type: none"> <li>• Input simple instructions to see what happens.</li> <li>• write/input a simple a program/code (no desired outcome)</li> <li>• write/input a simple a program/ code to achieve a desired outcome (ensuring it is precise, simple, clear and limited)</li> <li>• identify a bug in my programme/code</li> <li>• debug a program</li> </ul>	<p>iPads (15) Computers/inter net</p>
<p><b>Tynker</b> (iPad) Use all free levels working to appropriate stage for ability. Extend by progressing through the LO's above.</p>		
<p><b>Espresso coding</b> - From link on espresso.com –use login from G. drive. Begin with appropriate key stage and work through each level (cover more than one level during session). Ensure children are secure with each step before proceeding to next level. Choose one/two of the above LO's to focus on. Games can be linked with current topic. Each level has a tutorial which could be watched as a class or in groups of ability to enable differentiation.</p>		
<p><b>Hopscotch</b> (iPad) Use blank level to begin inputting simple codes. Use to experiment with coding. Once secure move to other levels.</p>		
<p><b>Scratch Jr.</b> (iPad) Use blank level to begin inputting simple codes (experiment to familiarise). Once secure give children objectives e.g. can you make Scratch (the cat) walk forwards and jump, can you make Scratch grow when he meets the penguin, etc. <b>Differentiation-</b> Provide different objectives to suite ability- Could have cards for children to work through which get more complicated.</p>		
<p><b>CargoBot</b> (iPad) Logical coding game. Use first few levels- Quite difficult but will challenge yr2 HAPs/G&amp;T children.</p>		