



Progression Map: Computing

The Intent for our Curriculum is to create responsible, creative Digital Citizens. We strive to do this through three strands supported by a variety of resources outlined below.

	Digital Literacy (Strand 1)	Information Technology (Strand 2)	Computer Science (Strand 3)
What?	Express themselves and develop their ideas through information and communication technology, and learning how to be a safe and responsible digital citizen through learning about staying safe online	Understand the purpose of technology and how we can communicate effectively through networks, systems and searches. Being exposed to different devices and systems.	The principles of computational thinking Design and build programs
How?	Jigsaw PSHE DigiDuck (EYFS&KS1) Hector's World (KS1) Google Be Internet Legends (Y2-Y4) YouTube Be Internet Citizens (Y5&6) Examples of Digital Outcomes: A Science investigation explained and demonstrated through videos using Clips A Geography unit summarised by creating an ebook on Pages A historical documentary produced on iMovie Statistics in Maths through Numbers	Teach Computing Scheme (Y1-6) Access to iPads and Laptops across school	Coding curriculum: Rodocodo (Y1-6) <i>Also Supported by</i> CodeSpark Academy (Y1&2) Tynker (Y3&4) Scratch Junior (EYFS&KS1) Scratch (KS2) Sphero (KS2)

Strand 1a: Digital Literacy (Online Safety)

	EYFS	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Online Safety	Can understand who their trusted adults are. Can understand that it is ok to say no. Understands what to do when they feel worried online Understands that it is important to be kind online and offline Understands healthy online habits Understands how to take care when using devices.	Can understand what private information is. Can understand who can help them with technology	Can understand that technology can help them and needs to be looked after respectfully. Can understand different ways of keeping information private. Can understand where to go for help with technology using contacts (people they know) and technology.	Can understand their responsibility to be safe online. Can understand respectful, acceptable/unacceptable behaviour with technology.	Can understand where to report concerns with technology (content & contact) within apps and programmes they are using. Can understand respectful, acceptable/unacceptable behaviour with technology	Can understand various reporting sites/ helplines. Can access different ways of receiving help (linked to responsibility) Can understand that not everyone uses technology safely. Can understand the importance and risks associated with acceptable /unacceptable behaviour	Can understand various reporting sites/ helplines. Can access different ways of receiving help (linked to responsibility) Can understand the importance and risks associated with acceptable /unacceptable behaviour

Strand 1b: Digital Literacy (Creative Media)

Year	Technology to Persuade	Technology to Inform	Technology to Discuss	Technology to Entertain
EYFS	Skills developed in order to facilitate upcoming year groups: Can make marks using a tablet / Can make patterns / Can order numbers using a device / Can take photos and annotate it / Can save and share a file / Can make a digital artefact / Can understand how to use a keyboard / Can sign in or log in to a device / Can start to use a mouse or trackpad / Can develop some mouse and trackpad skills / Can access a variety of apps to create digital content			
1	An infographic presenting the evidence for dinosaurs (PicCollage)	A presentation of key features of Sneinton as a local area (Keynote)	A podcast discussing the key question of whether animals are as important as humans (VoiceMemos)	An animated story of Robin Hood retold and produced as an eBook (BookCreator)
2	A persuasive advert for 'Votes for Women' (Pages)	A video of themselves speaking French (Clips) A presentation explaining the chronology of the Great Fire of London (Keynote)	An infographic showing how the oceans have been damaged and how to protect them (should we?) (Pages)	
3	Save the rainforest! A film or advert encouraging others to save the rainforest (Clips)	A presentation about the layers of the rainforest (Keynote)	A podcast discussing the cultural impact of the Romans (GarageBand)	
4	An eBook about rivers, sources, routes linking to Vikings (BookCreator)	An animation showing how the digestive system works (Keynote)	A blog discussing the effects of flooding and how different countries are differently impacted (Pages)	An audiobook including sound effects linked to part of a narrative (GarageBand)

5	A website synthesising the topic including pages about areas of learning including Lord Carrington and to compare Nottingham and the Caribbean (BookCreator with hyperlinks)	An animation showing key battle strategies of the Battle of Thermopylae in Ancient Greece (FlipAClip/Keynote)	A OneShow style report on whether the Victorian era was a period of progression (iMovie)	
6	An advert persuading the public about global climate change, the impact on rainforests and suggestions for how to mitigate the impact (iMovie)	A news report on the impact of the Blitz on families in Nottingham (iMovie)	A whole-curriculum podcast synthesis discussing whether the Mayans were the most advanced society to no longer survive, incorporating knowledge from all other civilization studies (GarageBand)	

Strand 2: Information Technology

	EYFS	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Digital Literacy: Purpose of Technology	Can understand that technology is used at home and school for different reasons.	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand how other people in our school & family use technology	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand how other people use technology in their community.	Can understand that technology is for creating, storing, manipulating and retrieving content for specific purpose. Can recognise that they can share information internally and use variety of software.	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand they can share information internally and use variety of software.	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand that content can be shared externally, combining variety of software and range of devices.	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand that content can be shared externally, combining variety of software and range of devices.
Digital Literacy: Networks				Can understand class connected via network.	Can recognise school network and possibility of communication and collaboration.	Can recognise the World wide web and understand how provide multiple services	Can recognise different types of network.

Digital Literacy: Data	Can sort data Can play yes or no games Can begin to be introduced to branching databases Can understand and create a pictogram			Can understand how to collect data.	Can understand how to collect and analyse data.	Can understand how to collect, analyse and evaluate data.	Can understand how to collect, analyse, evaluate and present data.
Digital Literacy: Searching online				Can understand how to search basic information online.	Can understand how to search files and information online.	Can understand how to search more detailed information online and recognise that not all results are useful.	Can understand how search results are organised, selected and used appropriately. Can evaluate content.
Supporting Resources		Technology around us	IT around us	Connecting Computers	The Internet	Sharing Information	Communication

Strand 3: Computer Science

Strands for progression in Computer Science lessons:

Sequencing	Debugging	Creating loops	Creating Functions	Using Selection	Using Variables
A sequence is the order in which things happen, like patterns and events.	Debugging is the process of correcting errors or 'bugs' in code	A loop is an instruction to repeat a set of commands for a specific number of times.	Creating functions is writing a section of a program that performs a specific task that can be used multiple times	Selection is how a computer makes decisions, and those decisions are based on conditions	Variables help computers remember values that can change

All good Computer Science lessons involve computational thinking skills.

These are the definitions we use for each skill:

Logical reasoning Logical reasoning helps us explain why something happens. We can use logical reasoning to work out exactly what a program or computer system will do.	Pattern spotting By identifying patterns, we can create rules and solve more-general problems. In computing, the method for looking for a general approach to a class of problems is called generalisation.	Decomposition Decomposition is the process of breaking down a task into smaller more-manageable parts. It helps us manage large project.
Debugging Bugs are errors in algorithms and code. Debugging is the process of finding and fixing these, and it can often take much longer than writing code in the first place.	Evaluating Evaluating is about making judgements, where possible in an objective and systematic way. Evaluating is about judging the quality, effectiveness and efficiency of products, solutions, processes and systems.	Tinkering Tinkering is where we try out something new to discover what it is and how it works. Tinkering is closely associated with logical reasoning.

Strand	EYFS	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Sequencing	To understand that a sequence is a set of actions performed in the correct order to achieve something To understand and explain what the commands do To understand that there can be more than 1 solution to a problem	To understand and explain what each command will do To create simple programs using a variety of commands	To understand and explain what each command will do To create simple programs using a variety of commands	To understand that decomposition is the process of breaking down larger problems into smaller, more manageable parts To use decomposition to break down larger problems into multiple parts	To understand that decomposition is the process of breaking down larger problems into smaller, more manageable parts	To write complex programs using Conditional Loops, Functions and Selection. To write complex programs that include Variables, Loops and Functions	To write complex programs that include Variables and Loops (basic and nested) To write complex programs that include Variables, Loops and Functions
Debugging	To understand that bugs are errors in code To understand that debugging is the process of correcting bugs	To debug simple programs with 1 or 2 bugs	To explain the run-step-fix method of debugging To debug programs which include Loops and Functions	To explain the run-step-fix method of debugging To debug programs which include Loops and Functions	To debug programs which include Loops and Functions	To debug programs which include Functions and Loops	To debug programs which include Functions, Loops and Selection

Creating Loops		To understand that Loops are used when you want to repeat actions To identify that using Loops creates more optimal solutions To use Loops with a program to control character's movement	To identify that using Loops creates more optimal solutions To use pattern recognition to identify Loops To create simple programs using Loops To use Loops to control the character's movement and rotation	To identify that Loops create more optimal solutions To use pattern recognition to identify Loops To use Loops to control the character's movement and rotation	To understand that Loops can be placed within other Loops, and that these are called Nested Loops To use Nested Loops within simple programs To write complex programs using selection and loops (basic and nested) To write complex programs using Selection, Loops and Functions.	To understand the difference between counting loops and conditional loops To understand that Conditional Loops repeat until a certain condition has been reached To create programs using Functions and Loops To write complex programs using Selection, Loops and Functions To write complex programs using Conditional Loops, Functions and Selection To write complex programs that include Variables, Loops and Functions	To understand the difference between Counting Loops and Conditional Loops To understand that Conditional Loops repeat until a certain condition has been reached To write complex programs that include Variables and Loops (basic and nested) To write complex programs that include Variables, Loops and Functions

Functions		To create programs using ready-made Functions	To understand that a function is a section of a program that performs a specific task and can be used multiple times To create programs that use ready-made Functions To debug programs which include Functions	To identify that Functions create more optimal solutions To complete ready-made Functions that are not fully complete, and use them in sequences To debug programs which include Functions	To create a Function and use it in a program To write complex programs using Selection, Loops and Functions.	To write complex programs using Selection, Loops and Functions To write complex programs using Conditional Loops, Functions and Selection To write complex programs that include Variables, Loops and Functions	To write complex programs that include Variables and Loops (basic and nested) To write complex programs that include Variables, Loops and Functions
Selection					To understand that Selection is how a computer program makes decisions, and that those decisions are based on conditions To use selection within simple programs To write complex programs using selection and loops (basic and nested)	To understand that Selection is how a computer program makes decisions, and that those decisions are based on conditions To write complex programs using Selection, Loops and Functions To write complex programs using Conditional Loops,	To understand that Selection is how a computer program makes decisions, and that those decisions are based on conditions To use the If Else command to write more complex Selection chains that instruct the character to perform different actions

					To write complex programs using Selection, Loops and Functions.	Functions and Selection To write complex programs that include Variables, Loops and Functions	To write complex programs using Conditional Loops, Functions and Selection
Variables						To understand that variables help computers remember values that can change To write complex programs that include Variables, Loops and Functions To use variables within simple programs	To understand that variables help computers remember values that can change To use variables within a simple program To write complex programs that include Variables and Loops (basic and nested) To write complex programs that include Variables, Loops and Functions