Progression of Skills in Computing

Our Aims:

At Barrington, it is our aim to ensure the teaching of computing knowledge, concepts and skills are part of a broad and balanced curriculum, based on the skills of:

- E-Safety
- Digital literacy
- Information Technology
- Computer Science

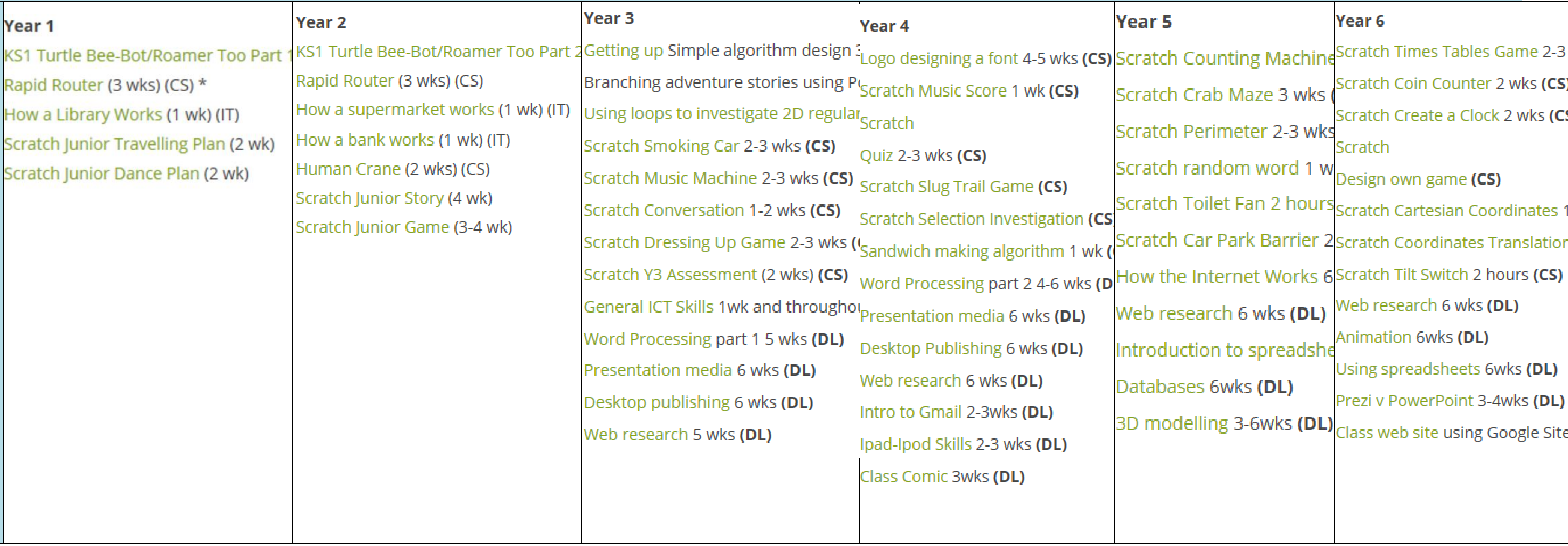
In order to deepen their skills as computational thinkers, pupils will be taught the knowledge, skills and understanding of computing. Throughout their primary school education, they will be encouraged to develop computational thinking skills and design and create their own digital content in order to become safe, independent learners.

The vocabulary children use to discuss computing will be built on each year so that by the time they leave Barrington Primary School they will have a secure and extensive knowledge to allow them to communicate effectively as logical computational thinkers.

INTENT					
	EYFS **Year 1/2**	**Year 3/4**		**Year 5/6**	
Knowledge	Pupils should be taught to: • Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions • create and debug simple programs • use logical reasoning to predict the behaviour of simple programs • use technology purposefully to create, organise, store, manipulate and retrieve digital content • recognise common uses of information technology beyond school • 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.	Pupils should be taught to: • design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts • use sequence, selection, and repetition in programs; work with variables and various forms of input and output • use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs • understand computer networks including the internet; how they can provide multiple services, such as the world wide web; and the opportunities they offer for communication and collaboration • use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content • 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 • use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.			
Skills **We are working through these using new teacher planning. Will re-assess after 1 year.**	**EYFS** **Year 1** **Year 2**	**Year 3**	**Year 4**	**Year 5**	**Year 6**
Digital Literacy	• Recognise common uses of information technology beyond school • 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	• Understand the opportunities [networks] offer for communication and collaboration • Be discerning in evaluating digital content • Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact			

Information Technology	• Use technology purposefully to create, organise, store, manipulate and retrieve digital content	• Use search technologies effectively • 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
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 • Create and debug simple programs • Use logical reasoning to predict the behaviour of simple programs	• Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts • Use sequence, selection, and repetition in programs; work with variables and various forms of input and output • Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs • Understand computer networks including the internet; how they can provide multiple services, such as the World Wide Web • Appreciate how [search] results are selected and ranked
IMPLEMENTATION		
Vocabulary	**Search engine, programme-sequence, output device, input device, network, e-safety, debug, coding, algorithm, cyberbullying, loop/repeat, data, embed, variables, keyboard, mouse, screen, right-click, left-click, escape, arrows, enter, backspace, close, copy, cut, save, delete, enter, layout, design, data, text, size, format, shift, caps lock, logging on, logging off, icon, www, internet, software, hardware, apps, ipads, laptop, desktop.**	
IMPACT		

Assessment opportunities



Year 1	**Year 2**	**Year 3**	**Year 4**	**Year 5**	**Year 6**
KS1 Turtle Bee-Bot/Roamer Too Part 1	KS1 Turtle Bee-Bot/Roamer Too Part 2	Getting up Simple algorithm design 3	Logo designing a font 4-5 wks **(CS)**	Scratch Counting Machine	Scratch Times Tables Game 2-3
Rapid Router (3 wks) (CS) *	Rapid Router (3 wks) (CS)	Branching adventure stories using P	Scratch Music Score 1 wk **(CS)**	Scratch Crab Maze 3 wks (	Scratch Coin Counter 2 wks **(CS)**
How a Library Works (1 wk) (IT)	How a supermarket works (1 wk) (IT)	Using loops to investigate 2D regular	Scratch	Scratch Perimeter 2-3 wks	Scratch Create a Clock 2 wks **(C**
Scratch Junior Travelling Plan (2 wk)	How a bank works (1 wk) (IT)	Scratch Smoking Car 2-3 wks **(CS)**	Quiz 2-3 wks **(CS)**	Scratch random word 1 w	Scratch
Scratch Junior Dance Plan (2 wk)	Human Crane (2 wks) (CS)	Scratch Music Machine 2-3 wks **(CS)**	Scratch Slug Trail Game **(CS)**	Scratch Toilet Fan 2 hours	Design own game **(CS)**
	Scratch Junior Story (4 wk)	Scratch Conversation 1-2 wks **(CS)**	Scratch Selection Investigation **(CS**	Scratch Car Park Barrier 2	Scratch Cartesian Coordinates 1
	Scratch Junior Game (3-4 wk)	Scratch Dressing Up Game 2-3 wks **(**	Sandwich making algorithm 1 wk **(**	How the Internet Works 6	Scratch Coordinates Translation
		Scratch Y3 Assessment (2 wks) **(CS)**	Word Processing part 2 4-6 wks **(D**	Web research 6 wks **(DL)**	Scratch Tilt Switch 2 hours **(CS)**
		General ICT Skills 1wk and throughou	Presentation media 6 wks **(DL)**	Introduction to spreadshe	Web research 6 wks **(DL)**
		Word Processing part 1 5 wks **(DL)**	Desktop Publishing 6 wks **(DL)**	Databases 6wks **(DL)**	Animation 6wks **(DL)**
		Presentation media 6 wks **(DL)**	Web research 6 wks **(DL)**	3D modelling 3-6wks **(DL)**	Using spreadsheets 6wks **(DL)**
		Desktop publishing 6 wks **(DL)**	Intro to Gmail 2-3wks **(DL)**		Prezi v PowerPoint 3-4wks **(DL)**
		Web research 5 wks **(DL)**	Ipad-Ipod Skills 2-3 wks **(DL)**		Class web site using Google Site
			Class Comic 3wks **(DL)**		

A whole-school **programming** strand might look something like this.
Year 1 Solving problems with Bee-Bots
Year 2 Turtle graphics on the floor and screen

Year 3 Scripted animations
Year 4 A maths quiz
Year 5 Computer games
Year 6 Developing applications for the mobile phone

http://ccc-computing.org.uk/capabilities

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 provided to a computer system, 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.