Lancasterian Primary School

A safe and welcoming learning community where:

- we all aim high;
- everyone is included;
- creativity is valued.

KS1/2 Curriculum Map

COMPUTER SCIENCE

	EYFS	**Y1**	**Y2**	**Y3**	**Y4**	**Y5**	**Y6**
Autumn 1	I recognise that a range of technology is used in places such as homes and schools. I can select and use technology for particular purposes.	I understand what algorithms are. I can create and debug simple programs. I can use logical reasoning to predict the behaviour of simple programs. I can record the route of a Bee Bot.	I understand that an algorithm is a sequence of instructions. I understand that computers need instructions (algorithms) to know what to do. I can create simple algorithms. I can debug simple programs. I can create programs using a variety of objects, actions, events and outputs successfully.	I can design a map for Blue Bot to travel. I can plan more complex sequences of instructions. I can debug my program. I can use repeats and turns of 45/90 degree angle turns. I can solve open ended problems with a floor robot.	I can create algorithms using sequence and loops. I understand that programmers refine algorithms to improve accuracy and efficiency. I can debug by identifying errors in the programming language.	I can write a simple algorithm to help write a program in Scratch. I can use sequencing [dealing with one thing at a time in a particular order] to achieve a specific goal. I can use a range of commands in Scratch to achieve a specific goal, including movement, animation and sound. I can use selection in programming by using the if, then and else commands in Scratch I can identify problems or bugs in a program and correct them.	I can control an on screen mimic or simulation with inputs and outputs. I can sequence instructions to control a number of output devices such as alarms, lights etc. I understand algorithms can be represented symbolically [flowcharts] or using instructions in a clearly defined language [turtle graphics]

	Software /Activities	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**
	Busythings LGFL Interacting with a touch screen	Control a Bee Bot	2Code Chimp Barefoot Resources - Unplugged	Control a floor robot (Blue Bot)	Busycode Purplemash -2Code	Scratch- Barefoot	Fairground simulation
Autumn 2	I can name the main parts of a computer. I can switch on and log into a computer. I can use a mouse to click and drag.	I can combine start up and input events to create more advanced apps and programs I can control an object on the screen using a range of instructions.	I understand that algorithms are step by step instructions and robots do exactly what they have been instructed to do. I can use logical reasoning to predict the behaviour of simple programs. I can create a simple algorithm to carry out specific actions. I can program a Blue Bot to perform a sequence of moves. I can debug a simple program . I can record the route of a BlueBot.	I can use a selection of tools to create images. I can use the transparent onion layer to edit images. I can make images appear to move when I run the program. I can add text and sound to my animation.	I can design and build a model with a motor and a sensor. I can program a lego model to move using keyboard keys or a sensor to control it. I can add a variable to makes changes. I can identify bugs and fix them.	I understand that floor robots can be controlled using Bluetooth or via a USB connection. I can design write and debug programs that accomplish specific goals. I can use sequence, selection and repetition in programs. I can use logical reasoning to explain how some simple algorithms work	I can create a scene for an animation. I can create a plan thinking about abstraction, what is important/ Unimportant and should/doesn't need to be included. I can use animation software to take still images. I can edit images and put a series of photos together to create a movie. I can add sound and text.
	Software /Activities	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**
	Busythijngs LGFL Mouse Control	2Go	Control a BlueBot	Animation linked to topic	Lego WeDo Code IT	In Obot	Stop frame Animation

Spring 1	I can use a mouse to create a picture. I can use a range of tools in the toolbar to create images. I can drag an image and make changes.	I can use a range of tools to draw images. I can use the mouse to drag an image to the next frame. I can use the onion layer to create a transparent image and make changes.	I can talk about how everyday devices can be controlled. I can control a turtle on the screen using appropriate buttons/blocks; make predictions and estimate distances and turns. I can create a sequence of instructions to control a programmable robot to carry out a pre-determined route to include direction, distance and turn. Input/re-arrange code into function boxes for the program to run. I can use trial and error when programming and use this to refine and improve my program.	I can use sequence in programs. I can use simple repetition (loops) and selection (if) in programs. I can debug programs for simple errors and make refinements.	I can choose appropriate code blocks to make things happen on the screen. I can create a set of instructions (Sequence, Code) which includes loops (Repetition). I can program (Control) an onscreen object (Sprite). I can refine and edit the process (De-bug)	I am able to name and talk about different types of inputs and outputs (Events) and use them within simple games game, simulations or models. I can create a set of instructions (Sequence, Code) which includes loops (Repetition) and or Selection to program (Control) a on screen object (Sprite) or floor robot and to be able to refine and edit the process (De-bug) to create a game, simulation or model.	I can write an algorithm to help design a multi-level game. I can use decomposition to break down the game into smaller parts. I can use a range of game features in Scratch including movement, scoring, rewards, Obstacles, Timer (extension), End of game (extension) I can use selection in programming by using the if, then and else commands in Scratch . I can use repetition and a variable in game design I can create a control system using input and output. I can identify problems or bugs in a program and fix them
	Software /Activities	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**
	Animation. - Purplemash 2Animate	Animation	Busycode	Scratch- Barefoot	Logo J2e	Scratch Barefoot 2Code	Scratch/Bsrefoot Code IT
Spring 2	I can use the arrow keys on the keyboard to control	I can explore an on-screen turtle and	I understand that an algorithm is a	To be able to give multiple sprites a series of instructions.	I can explain what selection is.	I can create a set of instructions	I can name and talk about different types of inputs and outputs.

	on object on the screen. I can learn the letters on the computer keyboard.	navigate it around a course or grid via a sequence of instructions. I can navigate around a course on a computer predict what will happen once the next command is entered. I can debug a program.	sequence of instructions. I understand that computers need instructions (algorithms) to know what to do. I understand that programs respond to different sorts of inputs.	I can write more complex programs. I can create an if statement in my program. I can use a timer and if statement to introduce selection in their program. I can create a variable set/change it. I can debug my program.	I can write a program using selection. I understand abstraction and can identify what is important and should be included and vice versa.	(Sequence, Code) which includes loops (Repetition) and or Selection to program (Control) a on screen simulation or model. I can edit the process (De-bug) to create a game, simulation or model. I understand that algorithms can include selection (if) and repetition (loops). I know that the behaviour of a program should be planned.	I can design a model and attach various inputs to get a desired output. I can write and debug programs to accomplish a specific goal.
	Software /Activities	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**
	Keyboard skills - Busythings Arrow Keys Letters	J2e - JiT	Junior Scratch	2Code	Scratch Barefoot	Flowol – Traffic light Lego - We Do	Crumble Code IT
Summer 1	I can explore outcomes when individual buttons are pressed on a robot. I can control a Bee Bot to move forwards/ backwards/ turn to reach a target	I understand that programs execute by following clear instruction. I can build one and two step instructions. I can use a simple program to produce a	I can use a range of tools to draw images. I can use the mouse to drag an image to the next frame. I can use the onion layer to create a transparent image and make changes.	I can create a simple set of instructions (Sequence, Code), to program (Control) a on screen object (Sprite) I can refine the process (De-bug) to create a game, simulation or model.	I can plan, write, test and debug programs. I can write programs that use selection. I can write programs that use inputs and output.	I understand that computers and devices use binary code. I understand how computers use numbers to represent things on devices. I can use 0 and 1 to code a scratch device.	I can explain that search results are ordered I can explain that a search engine follows rules to rank relevant pages I can suggest some of the criteria that a search engine checks

		background, create moving characters and objects. I can program a sound to occur when an object collides. I can make predictions on what happens based on the titles of different coding.	I can add text and sound to my animation.			I can use co ordinates to make specific lights light up on a 8x8 grid. I can use the web to find hex colour codes for my program.	to decide on the order of results I can describe some of the ways that search results can be influenced I can recognise some of the limitations of search engines I can explain how search engines make money
	Software /Activities	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**
	Control a Bee Bot	Control Sprites on the Screen (Chimp)	Animation	Busycode Lego	Microbit	Scratch LED Rainbow Matrix	NCCE - Computer Systems and Networks
Summer 2	I can use a mouse to create a picture. I can use the keyboard to write my name.	I understand that programs consist of instructions that are executed in order, one by one I can make simple programs.	I understand how block code works. I can program a sprite on the screen using the move and turn blocks. I can use a wait command. I can add speech to my program.	I can add sprites and create a setting in Kodu. I can program the sprites to move using keys. I can use selection in programming to score in the game I can identify problems and debug the program.	I can create a multi level game, simulation or model. I can create characters and landscape in Minecraft. I can add instructions and use selection in programming. I can create a multi-page game I can add scoring to the game I can add additional features such as sound and instructions.	I can recognise that data is transferred using agreed methods I can explain that networked digital devices have unique addresses I can explain that data is transferred over networks in packets	I can apply knowledge of control sequences to real life situations – where is control in our lives what inputs and outputs are there? I can create a set of instructions (Sequence, Code) which includes loops (Repetition) and Selection and Variables to program (Control) floor robot and to be able to refine and edit the process (De-bug) to create a model.

					I can identify problems and fix them by debugging.		I am able to use computational thinking to edit (De-bug) deconstruct and evaluate a program with smaller parts (Sub Procedures) to solve a problem or make the program more challenging or refined.
	Software /Activities	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**	**Software /Activities**
	J2e - JiT	Busycode	J2e Visual	Kodu	Minecraft	NCCE - Computer Systems and Networks	Lego EV3