THE DIOCESE OF SHEFFIELD ACADEMIES TRUST

Treeton C of E primary school Knowledge Skills Vocabulary for Computer Science

Computer Science				
KS1 National curriculum:	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			
	Substantive knowledge	**Disciplinary knowledge (skills)**	**Vocabulary**	
Year 1	Children understand that an algorithm is a set of instructions used to solve a problem or achieve an objective. They know that an algorithm written for a computer is called a program.	• Children can work out what is wrong with a simple algorithm when the steps are out of order, e.g. The Wrong Sandwich in Purple Mash and can write their own simple algorithm, e.g. Colouring in a Bird activity. Children know that an unexpected outcome is due to the code they have created and can make logical attempts to fix the code, e.g. Bubbles activity in 2Code. • When looking at a program, children can read code one line at a time and make good attempts to envision the bigger picture of the overall effect of the program. Children can, for example, interpret where the turtle in 2Go challenges will end up at the end of the program	sort criteria instruction algorithm computer program debug/debugging direction challenge arrow undo rewind forward backwards run	scale sound background code command event execute input object properties
Year 2	• Children can explain that an algorithm is a set of instructions to complete a task. When designing simple programs, children show an awareness of the need to be precise with their algorithms so that they can be successfully converted into code.	• Children can create a simple program that achieves a specific purpose. They can also identify and correct some errors, e.g. Debug Challenges: Chimp. Children's program designs display a growing awareness of the need for logical, programmable steps. • Children can identify the parts of a program that respond to specific events and initiate specific actions. For example, they can write a cause and effect sentence of what will happen in a program.	Action Algorithm Background Button Collison detection Debug Design mode Event Nesting Object Predict scene	Scale Sound Sequence Test Text timer

National curriculum	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. 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 logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.		
	Substantive knowledge	**Disciplinary knowledge (skills)**	**Vocabulary**
Year 3	• Children can list a range of ways that the internet can be used to provide different methods of communication. • Children are beginning to understand the difference in the effect of using a timer command rather than a repeat command when creating repetition effects. Children understand how variables can be used to store information while a program is executing. • Children's designs for their programs show that they are thinking of the structure of a program in logical, achievable steps and absorbing some new knowledge of coding structures. For example, 'if' statements, repetition and variables. They make good attempts to 'step through' more complex code in order to identify errors in algorithms and can correct this. e.g. traffic light algorithm in 2Code. • In programs such as Logo, they can 'read' programs with several steps and predict the outcome accurately.	Children can turn a simple real-life situation into an algorithm for a program by deconstructing it into manageable parts. Their design shows that they are thinking of the desired task and how this translates into code. Children can identify an error within their program that prevents it following the desired algorithm and then fix it. • Children demonstrate the ability to design and code a program that follows a simple sequence. They experiment with timers to achieve repetition effects in their programs. • They can use some of these methods of communication, e.g. being able to open, respond to and attach files to emails using 2Email. They can describe appropriate email conventions when communicating in this way	Action Algorithm Flowchart Command Debug/Debugging. Repeat Timer Properties Blocks of Command Object Sequence Sound Nesting Alert Execute Button Collision Detection Detecting Develop Output Predict Procedure Scene Values
Year 4	They understand 'if statements' for selection and attempt to combine these with other coding structures including variables to achieve the effects that they design in their programs • Children's designs for their programs show that they are thinking of the structure of a program in logical, achievable steps and absorbing some new knowledge of coding structures. For example, 'if' statements, repetition and variables • In programs such as Logo, they can 'read' programs with several steps and predict the outcome accurately.	Children's use of timers to achieve repetition effects are becoming more logical and are integrated into their program designs • Children's use of timers to achieve repetition effects are becoming more logical and are integrated into their program designs • As well as understanding how variables can be used to store information while a program is executing, they are able to use and manipulate the value of variables. • Children can make use of user inputs and outputs such as 'print to screen'. e.g. 2Code. They can trace code and use step-through methods to identify errors in code and make logical attempts to correct this. e.g. traffic light algorithm in 2Code	Action Alert Background Button Code Block command Debug/debugging Execute Co ordinates Flow chart If/else Nesting Objects types Predict Prompt Prompt for input Repeat properties Repeat until Selection Timer Variable Variable value LOGO, BK,FD,RT,LT,REPEAT, SETPC, SETPS, PU, PD Motherboard CPU RAM Graphics card Network card Monitor Speakers

National curriculum	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.		
	Substantive knowledge	**Disciplinary knowledge (skills)**	Vocabulary
Year 5	When children code, they are beginning to think about their code structure in terms of the ability to debug and interpret the code later, e.g. the use of tabs to organise code and the naming of variables. • Children understand the value of computer networks but are also aware of the main dangers. They recognise what personal information is and can explain how this can be kept safe	• Children may attempt to turn more complex real-life situations into algorithms for a program by deconstructing it into manageable parts. • Children are able to test and debug their programs as they go and can use logical methods to identify the approximate cause of any bug but may need some support identifying the specific line of code. • Children can translate algorithms that include sequence, selection and repetition into code with increasing ease and their own designs show that they are thinking of how to accomplish the set task in code utilising such structures • They are combining sequence, selection and repetition with other coding structures to achieve their algorithm design. • Children can select the most appropriate form of online communications contingent on audience and digital content, e.g. 2Blog, 2Email, Display Boards	Action Abstraction Algorithm Button Called Co-ordinates Decomposition Event Function Nesting Object Repeat Physical system Properties Run Score Sequence Simulation Simplify/simplified Simulation Tab timer Variable Animation Computer game Customise Evaluation Image Instructions Interactive Screenshot Texture Perspective Playability
Year 6	• Children are able to turn a more complex programming task into an algorithm by identifying the important aspects of the task (abstraction) and then decomposing them in a logical way using their knowledge of possible coding structures and applying skills from previous programs. • Algorithms that include sequence, selection and repetition into code and their own designs show that they are thinking of how to accomplish the set task in code utilising such structures, including nesting structures within each other. • Children are able to interpret a program in parts and can make logical attempts to put the separate parts of a complex algorithm together to explain the program as a whole. • Children understand and can explain in some depth the difference between the internet and the World Wide Web. Children know what a WAN and LAN are and can describe how they access the internet in school.	Children test and debug their program as they go and use logical methods to identify the cause of bugs, demonstrating a systematic approach to try to identify a particular line of code causing a problem. Coding displays an improving understanding of variables in coding, outputs such as sound and movement, inputs from the user of the program such as button clicks and the value of functions.	Action Alert Algorithm Background Button Called Command Co-ordinates Debug/ debugging Decomposition Developer event Flow chart Function If/else Launch Command Number variable Object Predict Procedure Prompt Properties Repeat Run Scene Selection Simulation String Tab Timer User input Variable Internet World wide web Network Router Network cables Wireless Audience Blog Blog page Blog post Collaborative Icon