Computer Science: Programming			
Unit	Declarative Knowledge	Procedural Knowledge	Vocabulary
Yr 1 Programming A – Moving A Robot	I understand that a program is a set of commands that a computer can run. I know that a command produces an outcome. I understand that a series of instructions can be issued before they can be enacted.	I can predict and then run my command on a floor robot. I can choose a series of commands that can be run as a program. I can build a sequence of commands that can be run as a program on a device	Bee-Bot, forwards, backwards, turn, clear, go, commands, instructions, directions, left, right, route, plan, algorithm, program.
Yr 1 Programming B – Programming Animations	I understand that a program is a set of commands that a computer can run. I know that a command produces an outcome. I understand that a series of instructions can be issued before they can be enacted. I know a list of step-by-step commands is a sequence.	I can choose a series of commands that can be run as a program. I can build a sequence of commands that can be run as a program on a device	ScratchJr, command, sprite, compare, programming, area, block, joining run, program, background, delete, reset, algorithm, predict, value, instructions
Yr 2 Programming A – Robot Algorithms	I know that a sequence is a series of instructions (commands). I know that a program is a sequence that can be run (executed). I understand that there is a different outcome when the order of the instructions is changed. I know that outcomes can be predicted before running (executing) a program.	I can predict and run a command on a given device. I can list commands, giving commands for a given purpose. I can choose a series of commands that can be run as a program, building a sequence of commands in steps. I can run a program on a device.	sequence, command, program, run, start, outcome, predict, blocks, design, actions, sprite, project, modify, change, algorithm, build, match, compare, debug.
Yr 2 Programming B –	I know that a sequence is a series of instructions (commands). I know that a program is a sequence that can be run (executed).	I can predict and run a command on a given device. I can list commands, giving commands for a given purpose.	sequence, command, program, run, predict, blocks, design, sprite, project, modify, algorithm, build, debug,

Programming Animations	I understand that there is a different outcome when the order of the instructions is changed. I know that outcomes can be predicted before running (executing) a program.	I can choose a series of commands that can be run as a program, building a sequence of commands in steps. I can run and debug a program on a device	evaluate, decomposition, code.
Yr 3 Programming A	I know that a program includes sequences of commands. I understand that the order of commands can affect a program's output. I know that different sequences can achieve different outputs, or the same output.	I can build a sequence of commands. I can combine and order commands in a program .I can create a sequence of commands to produce a given outcome.	Debug, sequence, decompose, selection, repetition, variables, input, output, algorithms, programs, code, block-based coding, Scratch, sprite, staging area, code block, run, event block, control blocks
Yr 3 Programming B	I know that different sequences can achieve different outputs, or the same output. I understand that a program starts because of an input.	I know that a program includes sequences of commands. I understand that the order of commands can affect a program's output. I know that different sequences can achieve different outputs, or the same output.	Sequence, sprite, event, action, program, pen, stage, algorithms, selection, repetition, code, debug, output
Yr 4 Programming A Repetition in Shapes	I can understand and identify 'loops' of repeated code within programs as repeating sets of instructions. I know that looped code within programs can run for an indefinite amount of time, *or* for a specified number of times. I know and understand when to use a looped instruction in my program and justify its use.	I can use a count-controlled loop or an infinite (forever) loop to produce a specified output. I can plan a program that includes loops to produce a given outcome. I can create two or more sequences that can run at the same time.	Loops, count-controlled loops, infinite loops, repetition, algorithms, logo, input, output.
Yr 4 Programming B – Repetition in Games	I understand that a loop command can be used to repeat instructions in a program. I understand that you can program a loop to stop after a specific number of times. I understand the importance of instructional order in a loop, and understand when to use	I can use an infinite (Forever) loop in a program, to produce a given outcome. I can use a count-controlled loop (e.g. repeat x10) in a program, to produce a given outcome. I can create two or more sequences of code in a program, that run at the same time.	Repetition, loop, infinite, count-controlled, sequence, algorithm, sprite, debug.

	a loop and when not to. I know the difference between a count-controlled and Infinite (forever) loop.		
Year 5 Programming A – Selection in Physical Computing	I understand that a conditional statement can either be true or false. I know the difference between a count-controlled loop and a condition-controlled loop I understand that selection can be used to branch the flow of a program I understand that a forever (infinite) loop can be used to repeatedly check whether a condition has been met.	I can create a condition-controlled loop. I can use a condition in an 'if…then…(else)' statement to start an action (produce an outcome)	Crumble controller, algorithm, sequence, selection (if…then…statements), repetition, loop, count-controlled or infinite loop, conditional statement, LED, sparkle, debug.
Yr 5 Programming B – Selection in Quizzes	I understand that a conditional statement can either be true or false. I know the difference between a count-controlled loop and a condition-controlled loop I understand that selection can be used to branch the flow of a program I understand that a forever (infinite) loop can be used to repeatedly check whether a condition has been met.	I can choose and condition and create a condition-controlled loop. I can use a condition in an 'if…then…(else)' statement to start an action (produce an outcome)	Algorithm, sequence, repetition, selection, loop, condition, count- controlled loop, condition-controlled loop.
Yr 6 Programming A – Variables in Games	I know that a variable is something that can be used and changed in a program. I can identify examples of variables and recognise that variable scan be numbers or letters. I know that variables have specific names and its value can be used by programs to change outcomes. To recognise a variable can hold only one	I can identify variables in existing programs and experiment with changing them. I can decide where in a program to set a variable and use an event or input to update it. I can use a variable in a conditional statement to control the flow of a program. I can use the same variable in more than one location in. program.	variable, change, name, value, set, event, declare

	value at any one time and you cannot access the previous value. I recognise a variable can be set as constant (have a fixed value) To explain that the name of a variable needs to be unique and is meaningless to a computer		
Year 6 Programming B – Sensing Movement	I know that a variable is something that can be used and changed in a program. I can identify examples of variables and recognise that variable scan be numbers or letters. I know that variables have specific names and its value can be used by programs to change outcomes. To recognise a variable can hold only one value at any one time and you cannot access the previous value. I recognise a variable can be set as constant (have a fixed value) To explain that the name of a variable needs to be unique and is meaningless to a computer	I can identify variables in existing programs and experiment with changing them. I can decide where in a program to set a variable and use an event or input to update it. I can use a variable in a conditional statement to control the flow of a program. I can use the same variable in more than one location in. program.	micro:bit, MakeCode, input, process, output, flashing, USB, trace, selection, condition, if then else, variable, random, sensing, accelerometer, value, compass, direction, navigation, algorithm, step counter, test, debug.