KS3 - Woodpeckers - Computing - Programming

All medium term plans represent an aspirational curriculum. Teachers and support staff may need to adapt the learning according to the class needs. This may include filling gaps in basic learning, preparation for the main curriculum and/or to build relationships between the staff and students.

Key Stage/Year	KS3 - Woodpeckers
Approximate Number of Lessons and Term	Autumn Half Term 2 - 8 lessons
Qualification/Exam (if applicable)	N/A

Consideration of prior learning	Teacher taught these students previously and is aware of their capabilities. Achieved objectives listed in computing tracker. Some students have previously completed programming activities in the last academic year (creating mazes on scratch)
How will learners' knowledge, skills and understanding be checked at the start of the unit?	Lessons allow for flexible complexity of tasks and completion of work to varying levels. Students are assessed continually while working and challenge is adapted dynamically.

How will learners' knowledge, skills and understanding be checked at the end of the unit?	Students are assessed continually while working and work towards creating an end piece that will be assessed against a set of criteria.

Learning Outcome	Approx. No. of Lessons	Potential Activities	Behaviour/Safety/Personal Development/SMSC Opportunities
• Compare how humans and computers understand instructions (understand and carry out) • Recognise that computers follow the control flow of input/process/output • Define a sequence as instructions performed in order, with each executed in turn • Predict the outcome of a simple sequence • Modify a sequence	1	Create a program that produces a repeating musical score	Using equipment safely. Culture - music from other countries, creating and playing music, singing. Using creativity and problem solving.
• Define a variable as a name that refers to data being stored by the computer • Recognise that computers follow the control flow of input/process/output • Predict the outcome of a simple sequence that includes variables • Trace the values of variables within a sequence • Make a sequence that includes a variable	1	Pupils work on tasks in Scratch that require them to use and change variables to debug programs.	Using equipment safely. Using creativity and problem solving.

• Define a condition as an expression that will be evaluated as either true or false • Identify that selection uses conditions to control the flow of a sequence • Identify where selection statements can be used in a program • Modify a program to include selection	1	Pupils work on tasks in Scratch that require them to use and change selection statements to debug programs.	Using equipment safely. Using creativity and problem solving.
• Create conditions that use comparison operators (>,<,=) • Create conditions that use logic operators (and/or/not) • Identify where selection statements can be used in a program that include comparison and logical operators	1	Pupils work on tasks in Scratch that require them to use and change logical and comparison operators to debug programs.	Using equipment safely. Using creativity and problem solving.
• Define iteration as a group of instructions that are repeatedly executed • Describe the need for iteration • Identify where count-controlled iteration can be used in a program • Implement count-controlled iteration in a program • Detect and correct errors in a program (debugging)	1	Pupils work on tasks in Scratch that require them to use and change count controlled iterations to solve and debug programs.	Using equipment safely. Using creativity and problem solving.
• Independently design and apply	1-2	Pupils use all of the skills covered in	Using equipment safely.

programming constructs to solve a problem (subroutine, selection, count-controlled iteration, operators, and variables)		the unit to design and create a Dance Moves program.	Using creativity and problem solving.
Complete end of unit assessment	1	Kahoot Quiz and end of unit assessment	Using equipment safely.

Possible Adaptations for Higher and Lower Achievers	Higher - create own program from scratch, encourage use of more advanced features and a more professional finish. Encourage peer teaching. More complex challenges can be given. Lower - use of frameworks and partially/fully constructed algorithms. Support in lessons to learn new features. Peer support.

How will the knowledge, skills and understanding imparted in this topic support them with future learning/qualifications or development?	This unit covers KS3 programming concepts linked to the NC, it will prepare learners for future programming units in the sequenced curriculum.