DEBUGGING ALGORITHMS – OVERVIEW

Overview

This unit builds on students' understanding of algorithms developed in Year 1 and introduces them to the concept of debugging as a purposeful process. Students move beyond simply following and creating instructions to identifying, explaining, and correcting errors in algorithms. They learn that when an algorithm does not work as intended, it is because of a mistake that can be found and fixed through careful thinking.

Through practical activities, students explore different types of bugs, including missing steps, incorrect order, and unclear instructions. They develop an understanding that algorithms can fail in different ways and that debugging involves logical reasoning to identify the cause of the problem. Students begin to predict what an algorithm will do before testing it and use the results of testing to help them improve it.

The unit places an emphasis on reasoning and explanation. Students are encouraged to explain why an algorithm does not work and how it can be improved, using simple computing vocabulary. They compare different versions of algorithms and consider which is most effective in achieving a goal.

Learning remains rooted in familiar, real-life contexts and unplugged activities, ensuring that students focus on the underlying concepts rather than specific software. This prepares students for applying these skills in block-based programming environments in Year 3.

Knowledge and understanding	Computing concepts
To identify different types of bugs in an algorithm.	Debugging
To predict what an algorithm will do before it is tested.	Logical reasoning
To test algorithms and use the outcome to identify and fix errors.	Testing and improving
To explain why an algorithm does not work and how it can be improved.	Evaluation
To understand that debugging is a process used to improve outcomes.	Problem solving.

The Computing Curriculum

You can see where the knowledge and understanding developed in this unit fits into the computing curriculum in the table below:

Prior Learning	Future Learning
EYFS Unit: Introduction to programmable toys	Year 2 Unit: Programming with Scratch Junior
Year 1 Unit: Introduction to algorithms	Year 3 Unit: Introduction to Scratch
Year 1 Unit: Patterns, predictions, and fixing	Year 3 Unit: Programming a quiz Scratch
	Year 4 Unit: Introduction to micro:bits
	Year 4 Unit: Introduction to HTML
	Year 5 Unit: Advanced micro:bits
	Year 5 Unit: Programming simulations

Prior Learning	Future Learning
	Year 6 Unit: Scratch – Programming a computer game
	Year 6 Unit: Introduction to Python

Cross-curricular links and extension activities

This unit provides links with mathematics, particularly through logical reasoning, sequencing, and identifying patterns. Students use prediction and reasoning skills that support problem-solving across the curriculum.

Links to science are developed through testing and observing outcomes. Students learn to make predictions, carry out simple tests, and use results to explain what has happened, mirroring scientific enquiry.

PSHE links are supported through collaborative problem-solving activities, where students work together to identify and fix errors. They develop resilience and understand that mistakes are a normal and valuable part of learning.

English links are developed through speaking and listening, as studentss explain their thinking, describe errors, and justify improvements. Structured discussion and simple written explanations support reasoning and communication skills.