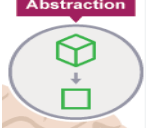
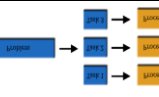
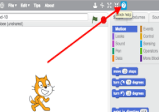
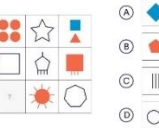
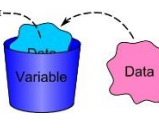


We are bug fixers

Prior Knowledge

In **Unit 3.1 We are programmers**, you developed your coding skills with Scratch.

Key Vocabulary:

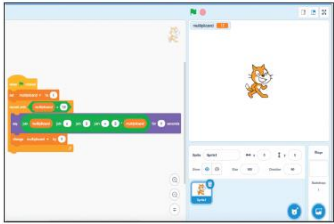
abstraction:		computational thinking approach to managing complexity by simplifying things, through identifying what is important and what detail can be hidden or ignored	output		information produced by a computer – in this case, an animation
algorithm:		a sequence of precise instructions or steps (sometimes a set of rules) to achieve an objective	parallel processing		when programs run (or appear to run) simultaneously
bug		an error or mistake in a program or algorithm, causing the computer or robot to behave in a manner that was not originally intended	program		a sequence of instructions (or sometimes a set of rules) that can be followed by a computer
code:		instructions (or sometimes rules) that can be understood by a computer	repetition		programming construct which allows a group of instructions to be repeated a number of times, or until a certain condition is met
debug:		correct mistakes in a computer program or algorithm	scratch		simple, block-based programming language in which programs for characters are built by snapping together code blocks
event		something that happens within a computer program to cause some particular code to be run, such as a sprite being tapped by the user	sequence		placing programming instructions in order, so that each happens one after the other
input		data supplied to a computer, in this case, pressing buttons on the robot	sprite		a graphical character in a program that can be given its own sequence of instructions
logical reasoning		to be able to give a reason for something which others would have to accept as correct	variable		named storage location in a computer's memory

Key Skills

How do I know that a program has a bug?

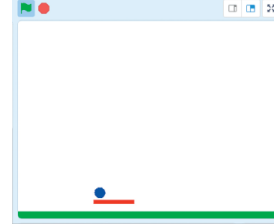
How do I correct a bug?

Key learning:



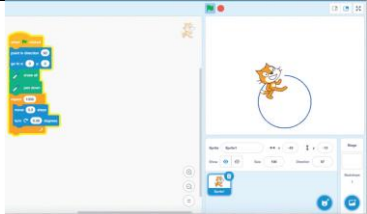
Session 1: Fixing an off-by-one bug

What is an **Off-by-one** bug?



Session 4: Fixing a conceptual bug

What is a **conceptual** bug?



Session 2: Fixing a performance bug

What is a **performance** bug?



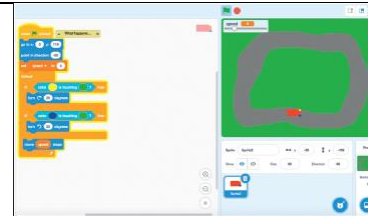
Session 5: Fixing an arithmetical bug

What is an **arithmetical** bug?



Session 3: Fixing a multithread bug

What is a **multithread** bug?



Session 6: Fixing a resource bug

What is a **resource** bug?

Key Applications (apps)



Next Steps

You will continue to develop Scratch programs and apply debugging skills to your own programs in **Unit 4.1: We are software developers** and **Unit 5.1: We are game developers**.