ALMOND HILL JUNIOR SCHOOL MEDIUM TERM PLAN

TOPIC TITLE/SUBJECT: Computing - Programming A Repetition on shapes **YEAR GROUP: 4** **TERM: Spring**

Vocabulary	**Skills**	**What we already know**
Commands Values Algorithm Program Code Count controlled loop Repetition Sequence Decompose Chunks of action Debug	-To identify that accuracy in programming is important (Program by typing commands/change values/create code for purpose) -To create a program in a text-based language (Use a template to plan a program/Write an algorithm/test algorithm in text-based language) -To explain what 'repeat' means (identify repetition in everyday tasks/identify patterns in a sequence/use count controlled loops to produce a given outcome) -To modify a count-controlled loop to produce a given outcome (identify effects of changing the number of times a task is repeated/predict outcomes when loops are present/choose values to change in a loop). To decompose a task into small steps (identify 'chunks' of action in real world/use procedure in a program/explain that computers use procedures repeatedly) To create a program that uses count-controlled loops to produce a given outcome (design a program with count-controlled loops/use designs to make a program/develop and debug program)	This unit progresses students' knowledge and understanding of programming. It progresses from the sequence of commands in a program to using count-controlled loops. Pupils will create algorithms and then implement those algorithms as code.

pplication/Outcomes

Programming a screen turtle: introduction to programming in Logo. Logo is a text-based programming language where pupils type commands that are then drawn on screen. Learn the Logo commands and will use their knowledge of them to read and write code.

Programming letters: create algorithms for their initials. Implement these algorithms by writing them in Logo commands to draw the letter. Debug their code by finding and fixing any errors that they spot.

*Patterns and repeats: L*ook at examples of patterns in everyday life. Recognise where numbers, shapes, and symbols are repeated, and how many times repeats occur. Create algorithms for drawing a square, using the same annotated diagram as in Lesson 2. Use this algorithm to program a square the 'long' way and recognise the repeated pattern within a square. Once they know the repeated pattern, they will use the repeat command within Logo to program squares the 'short' way.

*Using loops to create shapes: W*ork with count-controlled loops in a range of contexts. Think about a real-life example, then they will move on to using count-controlled loops in regular 2D shapes. Trace code to predict which shapes will be drawn and they will modify existing code by changing values within the code snippet.

*Breaking things down: F*ocus on decomposition. They will break down everyday tasks into smaller parts and think about how code snippets can be broken down to make them easier to plan and work with. Learn to create, name, and call procedures in Logo, which are code snippets that can be reused in their programming.

*Creating a program: A*pply the skills that they have learnt in this unit to create a program containing a count-controlled loop. Design wrapping paper using more than one shape, which they will create with a program that uses count-controlled loops. Begin by creating the algorithm, either as an annotated sketch, or as a sketch and algorithm, and then implement it as code. Debug their work throughout and evaluate their programs against the original brief.

Other/Cross Curricular Links			**Adaptation for SEND**		
Maths – Shape, polygons, angles	Art- Designing aesthetic images	DT-Plan, make, evaluate	Adapted tasks	Adapted resources	Additional support