

Kit's Clever Coding Planner

Learning objective: To design a logical algorithm and debug potential errors in a computing project.

Welcome, digital architect! Use this planner to design your own sequence of instructions (an algorithm) for a computer program. Think about the steps required to achieve your goal, anticipate potential 'bugs' or mistakes, and explain how you would solve them. Remember to use precise, logical language.

Word bank: algorithm · sequence · variable · debug · input · output · iteration · condition

1. Project Concept: Describe the main goal of your program and who the intended user is. Justify why this program would be useful. (3 marks)

2. The Algorithm: List the sequence of steps your program must follow. Ensure these are in the correct order for the computer to process. (4 marks)

3. Identifying Bugs: What might happen if the user inputs incorrect information? Explain how your code could handle this error. (3 marks)

4. Iteration: Identify one part of your code that needs to repeat (a loop). Explain why iteration is more efficient than writing the same instruction multiple times. (3 marks)

5. Variables: Define at least one variable for your program (e.g., a score or a timer). How will this value change during the program? (2 marks)

6. Evaluation: Compare your final algorithm to a simpler version. Why is your structured design better at solving the problem effectively? (3 marks)

Extension challenge: Creative Challenge: If you were to add a 'condition' (an 'if-then' statement) to your program to make it interactive, what would it be? Write the logic statement and explain the outcome if the condition is met.