# Lesson 3 – Algorithms

|  |  |
| --- | --- |
| The Big Picture – Why Is This Relevant? | Learning Objectives |
| * Prior to creating computer programs, the structure of the code should be planned using a flow chart. The flow chart will represent the core algorithm that needs to be followed to solve a problem * Learners will have followed algorithms throughout their lives without realising it * Throughout the course so far Learners have looked at input, processes and outputs. This lesson will now enable them to plan their programs using flow chart diagrams | * Understand what an algorithm is * Understand that when computer programs produce unexpected outputs they could be due to bugs in programming code * Create a simple flow chart using input, output, process and decisions |
| Engagement – How Can I Engage Learners? | Assessment for Learning |
| * Learners will be able to review flow charts and trace through them to identify outputs * Learners will be able to review everyday tasks and identify how they have been following algorithms throughout their lives * Learners will have the opportunity to create a flow chart to demonstrate a task that they perform every day. They will enjoy being able to use the correct flow chart symbols to diagrammatically represent this task | **Expected Progress:**   * All Learners will understand that an algorithm is a set of instructions which allow a problem to be solved * Learners will understand the meaning of different shapes of flow chart symbols   **Good Progress:**   * Learners will be able to identify the correct flow chart symbols for inputs, processes, output and decisions * Learners will be able to identify bugs in a program   **Exceptional Progress:**   * Learners will be able to draw an accurate flow chart to represent a daily task which they perform of their own choice * Learners will be able to identify and correct bugs in a program |
| Links to Program of Study | |
| * Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems | |
| Key Concepts | Key Words |
| * Algorithms provide a set of instructions which, when followed, solve a problem. They should begin with a ‘Start’ and terminate with a ‘Stop’. They should always produce the correct output for any valid input * If an unexpected output is produced this normally highlights a bug in a program. This may not necessarily cause the program to crash, but it will cause it to behave unexpectedly * Different symbols in flow charts represent different types of task. Learners will be introduced to the shapes for Start/Stop, Input/Output, Process and Decision | * Algorithm * Input * Process * Output * Decision * Bug |
| Differentiation | Resources |
| Once learners have produced a flow chart for a task that they perform on a daily basis they should think about how the overall task can be further broken down. This should then allow them to add additional processes/decisions into their flow charts. They could expand their work by identifying any variables that would be needed. | * Lesson 3 ppt * Lesson 3 Computer Bugs worksheet * Lesson 3 Flow Charts worksheet |
| Lesson Flow | |
| * Introduce Learners to the term ‘algorithm’. Explain how they will have been following algorithms throughout their lives without even knowing * Explain that a successful algorithm should always produce a correct output for any valid input * If an algorithm is followed and produces an unexpected output this is due to a ‘bug’ in the programming code * Learners should work through the ‘bugs’ worksheet and try to identify where the errors in the programs exist * When initially planning an algorithm, the initial step is to show the different elements on a diagram called a flow chart. Explain the purpose of the symbols for input/output, start/stop, process and decision. Stress that if a question has to be answered to determine what to do next then this highlights the need for a decision * Learners should then trace through the flow charts on the worksheet to determine what the outputs will be * Learners should then choose a task that they perform on a regular basis and draw a flow chart for it. For example, eating breakfast | |
| Making | |
| There are no making activities in this lesson. | |