Component 1: Algorithms and Constructs

Dry run, amend and write algorithms

Term	Definition
Algorithm	In programming, an algorithm is a set of instructions that can be used to solve a given problem.
Dry run	A dry run is the process of a programmer manually working through their code to trace the value of variables.

A dry run involves a programmer working through the code of a program to trace the value of variables to see that it is used and updated as expected.

A programmer will usually use a trace table to record the value of the variables as they work through the program. The table will have a column for each variable and each row will represent a line of the programming code.

```
1    outValue is integer
2    set outValue = 0
3
4    Declare Subroutine Multi
5
6     for i = 1 to 3
7      for j = 1 to 3
8       outValue = i * j
9       output outValue
10     next j
11    next i
12
13   End Subroutine
```

Example question 1

Complete the table to show all the outputs of this algorithm.

Trace table for solution

i	j	Output value
1	1	1
1	2	2
1	3	3
2	1	2
2	2	4
2	3	6
3	1	3
3	2	6
3	3	9

Amending algorithms

In some exam questions candidates will be asked to amend an algorithm that contains an error or to insert missing lines.

Example question 2

An algorithm is intended to calculate the area of a circle.
The algorithm accepts a single input; the radius, and outputs the area.
The area is calculated by multiplying Pi by the radius squared.

```
1    set Pi = 3.142
2    radius is real
3    . . .
4
5    Declare FindArea
6      {procedure to calculate the area of a circle}
7
8    . . .
9
10   End Subroutine
11
12   startMainProg
13    output "Type in the radius"
14    . . .
15
16    call FindArea
17
18    output "The area is:"
19    . . .
20    endMainProg
```

Lines 3, 8, 14 and 19 are missing.

Using four of the lines of code below complete the algorithm.

- `else is TRUE`
- `output area`
- `input radius`
- `input flag`
- `area = Pi * radius * radius`
- `End Subroutine`
- `area is real`

Writing algorithms

Sometimes a question will ask for an algorithm to be written for a given situation.

Variables

A variable is a holder for an item of data (that can be changed) of a specific type and length. Data types include character, integer, real and Boolean.

Variables must be defined in a program by specifying type and length and assigning a self-identifying name, indicative of the data stored.

Variables are assigned a memory location when the program is loaded into memory. The data in a variable may be created, edited and deleted whilst the program is running.

Local and global variables

A variable declared in a sub-procedure has **'local scope'** because it can only be accessed from within that sub-procedure.

A variable declared in the main program has **'global scope'** because it can be accessed from all parts of the program.