

Complete the table by placing ticks in the relevant boxes.

[6]

[2]

[3]

[4]

2.



Question	Answer	Marks																									
1(a)	<table border="1"><thead><tr><th>Item</th><th>Statement</th><th>Input</th><th>Process</th><th>Output</th></tr></thead><tbody><tr><td>1</td><td>SomeChars = "Hello World"</td><td></td><td>✓</td><td></td></tr><tr><td>2</td><td>OUTPUT RIGHT(String1,5)</td><td></td><td>✓</td><td>✓</td></tr><tr><td>3</td><td>READFILE (MyFile, String2)</td><td>✓</td><td></td><td></td></tr><tr><td>4</td><td>WRITEFILE (MyFile, "Data is " &amp; String2)</td><td></td><td>✓</td><td>✓</td></tr></tbody></table> <p>Mark as follows:</p> <p>Row 1 as shown Row 2 no marks if tick in Input column, otherwise 1 mark per tick Row 3 as shown Row 4 no marks if tick in Input column, otherwise 1 mark per tick</p>	Item	Statement	Input	Process	Output	1	SomeChars = "Hello World"		✓		2	OUTPUT RIGHT(String1,5)		✓	✓	3	READFILE (MyFile, String2)	✓			4	WRITEFILE (MyFile, "Data is " & String2)		✓	✓	6
Item	Statement	Input	Process	Output																							
1	SomeChars = "Hello World"		✓																								
2	OUTPUT RIGHT(String1,5)		✓	✓																							
3	READFILE (MyFile, String2)	✓																									
4	WRITEFILE (MyFile, "Data is " & String2)		✓	✓																							
1(b)(i)	<ul style="list-style-type: none"><li>Integer / Real / Single / Double / Floating Point / Float</li><li>Boolean</li></ul>	2																									
1(b)(ii)	<table border="1"><thead><tr><th>Expression</th><th>Evaluates to</th></tr></thead><tbody><tr><td>(FlagA AND FlagB) OR FlagC</td><td>TRUE</td></tr><tr><td>FlagA AND (FlagB OR FlagC)</td><td>TRUE</td></tr><tr><td>(NOT FlagA) OR (NOT FlagC)</td><td>FALSE</td></tr></tbody></table> <p>1 mark per answer</p>	Expression	Evaluates to	(FlagA AND FlagB) OR FlagC	TRUE	FlagA AND (FlagB OR FlagC)	TRUE	(NOT FlagA) OR (NOT FlagC)	FALSE	3																	
Expression	Evaluates to																										
(FlagA AND FlagB) OR FlagC	TRUE																										
FlagA AND (FlagB OR FlagC)	TRUE																										
(NOT FlagA) OR (NOT FlagC)	FALSE																										
1(c)	<p>MyCount ← 101</p> <p>REPEAT</p> <p>    OUTPUT MyCount</p> <p>    MyCount ← MyCount + 2</p> <p>UNTIL MyCount &gt; 199</p> <p>1 mark for each of the following:</p> <ul style="list-style-type: none"><li>Counter initialisation</li><li>Repeat ... Until loop</li><li>Method for choosing (correct range of) odd numbers</li><li>Output all odd numbers in the range</li></ul> <p>Note: Counter variable name must be consistent</p>	4																									

Question	Answer	Marks
5(a)	<pre> PROCEDURE MakeNewfile   DECLARE OldFileLine : STRING   DECLARE NewFileLine : STRING    OPENFILE "EmailDetails" FOR READ   OPENFILE "NewEmailDetails" FOR WRITE    WHILE NOT EOF("EmailDetails")     READFILE "EmailDetails", OldFileLine     NewFileLine ← "00" &amp; OldFileLine     WRITEFILE "NewEmailDetails", NewFileLine   ENDWHILE    CLOSEFILE "EmailDetails"   CLOSEFILE "NewEmailDetails"  ENDPROCEDURE </pre> <p>Mark as follows:</p> <ol style="list-style-type: none"> <li>1. Variable declaration of <code>STRING</code> for <code>OldFileLine</code> (or equivalent)</li> <li>2. Open <code>EmailDetails</code> for <code>READ</code></li> <li>3. Open <code>NewEmailDetails</code> for <code>WRITE</code></li> <li>4. Correct loop checking for <code>EOF</code> (<code>EmailDetails</code>)</li> <li>5. Reading a line from <code>EmailDetails</code> in a loop</li> <li>6. Correct concatenation in a loop</li> <li>7. Writing a line to <code>NewEmailDetails</code> in a loop</li> </ol> <p>Closing both files</p>	8
5(b)	<p><b>Invalid string examples:</b></p> <p>A string with nothing <b>before</b> '@'</p> <p>A string with nothing <b>after</b> '@'</p> <p>A string with 1 or 2 characters after '@'</p> <p>A string with no '@' symbol</p> <p>A string with more than one '@' symbol</p> <p><b>Explanation</b></p> <p>Sensible explanation mapping each given string to an individual rule</p> <p>One mark for string</p> <p>One mark for explanation</p> <p>Each rule should be tested once only</p>	6