<table>
<tr><th>Question</th><th>Part</th><th>Marking guidance</th><th>Total marks</th></tr>
<tr><td>01</td><td>1</td><td>2 marks for AO1 (recall)

A sequence/number/set of steps/instructions;
that can be followed to complete a task/to solve a problem;

A. Different wording with similar meaning</td><td>2</td></tr>
<tr><td>01</td><td>2</td><td>3 marks for AO1 (recall)

One mark for each correct distinct label.

If the answers given were, for example, C, C, B then award only 1 mark for the B as the C is duplicated. Likewise if C, C, C was the answer then no marks would be given. The correct table is:

<table><tr><th></th><th>Label</th></tr><tr><td>Breaking a problem down into a number of sub-problems.</td><td>C</td></tr><tr><td>The process of removing unnecessary detail from a problem.</td><td>A</td></tr><tr><td>Defines the sort of values a variable may take.</td><td>B</td></tr></table>
A. If actual terms are written out instead of labels
R. All instances of duplicate labels</td><td>3</td></tr>
</table>

Question	Part	Marking guidance	Total marks
02	1	**Mark is for AO2 (apply)** Boolean//bool; **I.** Minor spelling mistakes	1

<table>
<tr><th>Question</th><th>Part</th><th>Marking guidance</th><th>Total marks</th></tr>
<tr><td>03</td><td>1</td><td>2 marks for AO1 (understanding)

Correct table is:

<table>
<tr><th>Values</th><th>Data type</th></tr>
<tr><td><code>true</code>, <code>false</code></td><td>Boolean;</td></tr>
<tr><td><code>0</code>, <code>1</code>, <code>2</code></td><td>Integer;</td></tr>
</table>

A. Bool/bool/boolean instead of Boolean
A. Int/int instead of integer</td><td>2</td></tr>
</table>

<table>
<tr><th>Qu</th><th>Part</th><th>Marking guidance</th><th>Total marks</th></tr>
<tr><td>04</td><td>1</td><td>Mark is for AO2 (apply)

B: Integer;

R. If more than one lozenge shaded.</td><td>1</td></tr>
<tr><td>04</td><td>2</td><td>1 mark for AO2 (apply)

Boolean/bool;</td><td>1</td></tr>
<tr><td>04</td><td>3</td><td>3 marks for AO2 (apply)

1 mark for each correct value of <code>valid</code>;;;

<table>
<tr><th>Value of <code>instr</code></th><th>Final value of <code>valid</code></th></tr>
<tr><td><code>ADD R0, R1</code></td><td><code>False</code></td></tr>
<tr><td><code>ADD: R0, R1</code></td><td><code>True</code></td></tr>
<tr><td><code>HALT</code></td><td><code>True</code></td></tr>
</table></td><td>3</td></tr>
<tr><td>04</td><td>4</td><td>Mark is for AO1 (understanding)

Machine code;

A. binary;
A. object code;</td><td>1</td></tr>
<tr><td>04</td><td>5</td><td>2 marks for AO1 (understanding)

Max 2 marks from:

(High-level languages) are better supported;
(High-level languages) provide built-in subroutines;
(High-level languages) provide programming structures such as iteration and selection;
(Code written in high-level languages) is normally shorter;
(High-level languages) allow creation of subroutines;
(High-level languages) provide data structures;
(High-level languages) are easier to understand/read;
(High-level languages) are easier to debug;

A. any other correct justification.</td><td>2</td></tr>
</table>

Qu	Part	Marking guidance	Total marks

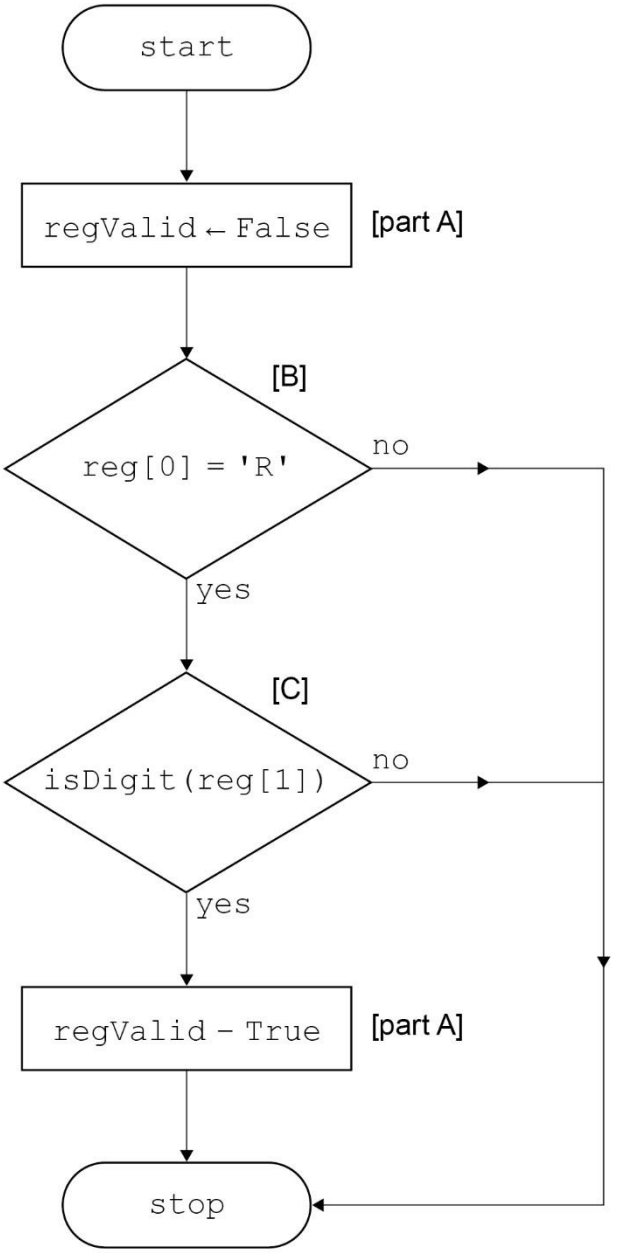
04	6	**3 marks for AO3 (program)**
		
		Mark A for setting the variable `regValid` to `True`/`False` within a selection structure;
		Mark B for using a Boolean condition that checks if the first character is an `'R'`;
		Mark C for using a Boolean condition that checks if the second character is a digit;
		
		Max 2 marks if any errors in the answer.
		
		A. minor changes to variable identifiers if the meaning is still clear.
		
		Example of fully correct answer:
		
		`regValid ← False` [part A]
		`IF reg[0] = 'R' and isDigit(reg[1]) THEN` [B,C]
		`  regValid ← True` [part A]
		`ENDIF`
		
		Example of another fully correct answer:
		
		`IF reg[0] = 'R' THEN` [B]
		`  IF isDigit(reg[1]) THEN` [C]
		`  regValid ← True` [part A]
		`  ELSE`
		`  regValid ← False` [part A]
		`  ENDIF`
		`ELSE`
		`  regValid ← False` [part A]
		`ENDIF`
		
		Example of 2 mark answer:
		
		`IF reg[0] = 'R' or isDigit(reg[1]) THEN` [B,C]
		`  regValid ← True` [part A]
		`ELSE`
		`  regValid ← True` [part A]
		`ENDIF`
		
		(only 2 marks awarded as the answer contains an error in the Boolean condition)
			3

Example of another 2 mark answer:

```
IF reg[0] = 'R' and isDigit(reg[1]) THEN [B,C]
     regValid ← True                     [part A]
ENDIF
```

(only 2 marks awarded as only part of mark A is given)

Example of a fully correct flowchart solution:



05		**4 marks for AO2 (apply)** A record could be used to store the data of one song; An array could store all of the songs/records; One mark for one of the following, two marks for all three: • The song title could be a string • The singer could be a string • The year of release could be an integer/date.	4

<table>
<tr><th>Question</th><th>Part</th><th>Marking guidance</th><th>Total marks</th></tr>
<tr><td>06</td><td>1</td><td>2 marks for AO2 (apply)

<table>
<tr><th>Input value of
orderTotal</th><th>Input value of
deliveryDistance</th><th>Output</th></tr>
<tr><td>55.5</td><td>2</td><td>1.5;</td></tr>
<tr><td>35.0</td><td>5</td><td>7.0;
A. 7</td></tr>
</table>
</td><td>2</td></tr>
</table>

<table>
<tr><th>Question</th><th>Part</th><th>Marking guidance</th><th>Total marks</th></tr>
<tr><td>06</td><td>2</td><td>Mark is for AO2 (apply)

2 // two;</td><td>1</td></tr>
</table>

<table>
<tr><th>Question</th><th>Part</th><th>Marking guidance</th><th>Total marks</th></tr>
<tr><td>06</td><td>3</td><td>2 marks for AO2 (apply)

<table>
<tr><th>Variable identifier</th><th>Data type</th></tr>
<tr><td>deliveryCost</td><td>Float // Real // Decimal</td></tr>
<tr><td>messageOne</td><td>String // str</td></tr>
</table>

I. Case
A. Programming language specific data types eg Single in VB.NET</td><td>2</td></tr>
</table>

<table>
<tr><th>Question</th><th>Part</th><th>Marking guidance</th><th>Total marks</th></tr>
<tr><td>06</td><td>4</td><td>Mark is for AO1 (recall)

Boolean // Bool;
Int // Integer;
Date/Time;
Character;

R. Any answer that was given in 02.3

I. Case
A. Any reasonable data type</td><td>1</td></tr>
</table>

Question	Part	Marking guidance	Total marks
07	1	**Mark is for AO2 (apply)** **A** 2; **R.** if more than one lozenge shaded	1

Question	Part	Marking guidance	Total marks
08	1	**Mark is for AO2 (apply)** Boolean//bool; **I.** Case	1

Question	Part	Marking guidance	Total marks
09	1	**Mark is for AO2 (apply)** **D** `value ← LEN(film);` **R.** If more than one lozenge shaded	1

Question	Part	Marking guidance	Total marks
09	2	**Mark is for AO2 (apply)** `POSITION(film, letter);` **I.** Case **R.** Quotes	1

Question	Part	Marking guidance	Total marks
09	3	**Mark is for AO2 (apply)** **C** integer; **R.** If more than one lozenge shaded	1

Question	Part	Marking guidance	Total marks
09	4	**Mark is for AO1 (understanding)** When a value is given to a variable; // When a variable is assigned a value;	1

Question	Part	Marking guidance	Total marks
09	5	**2 marks for AO3 (program)** **Program Logic** **Mark A** for using user input and storing the result in a variable; **Mark B** for displaying `You entered` followed by the name of the film entered by the user in the appropriate place; **I.** Case **I.** Indentation **I.** Messages or no messages with input statements **I.** Gaps/spaces throughout the code, except where to do so would explicitly alter the logic of the code in a way that makes it incorrect **Maximum 1 mark** if any errors in code. **Note to examiners** In C#/VB.NET examples, explicit variable declarations are not shown. Refer to the specific language type issues section of the appropriate Marking guidance document. Any correct variable declarations in student answers should be accepted. **C# Example 1 (fully correct)** `film = Console.ReadLine();` **(A)** `Console.WriteLine("You entered " + film);` **(B)** **A.** `Write` in place of `WriteLine` **C# Example 2 (fully correct)** `film = Console.ReadLine();` **(A)** `Console.Write("You entered ");` **(Part B)** `Console.WriteLine(film);` **(Part B)** **Python Example 1 (fully correct)** `film = input()` **(A)** `print("You entered", film)` **(B)** **Python Example 2 (fully correct)** `film = input()` **(A)** `print("You entered " + film)` **(B)**	2

Python Example 3 (fully correct)

```
film = input()                          (A)
print(f"You entered {film}")            (B)
```

VB.NET Example 1 (fully correct)

```
film = Console.ReadLine()               (A)
Console.WriteLine("You entered " & film)  (B)
```

A. `Write` in place of `WriteLine`

VB.NET Example 2 (fully correct)

```
film = Console.ReadLine()               (A)
Console.WriteLine("You entered " + film)  (B)
```

A. `Write` in place of `WriteLine`

VB.NET Example 3 (fully correct)

```
film = Console.ReadLine()               (A)
Console.Write("You entered ")           (Part B)
Console.WriteLine(film)                 (Part B)
```

A. `Write` in place of `WriteLine`