Mark Scheme

Q1.

Question Number	Answer	Additional Guidance	Mark
i	5		**1**
ii	0		**1**
iii	Variable	Accept words that describe a use of variable in this case, e.g. flag	**1**
iv	1		**1**
v	Looping over every item in a data structure		**1**

Q2.

Question Number	Answer	Additional Guidance	Mark
(i) **Clerical**	Any **one** from: • Array • List		**1**

Question Number	Answer	Additional Guidance	Mark
(ii) **Clerical**	Yellow		**1**

Question Number	Answer	Additional Guidance	Mark
(iii) **Clerical**	Abstraction		**1**

Question Number	Answer	Additional Guidance	Mark
(iv) **Clerical**	Decomposition		**1**

Q3.

Question Number	Answer	Additional Guidance	Mark
(i)	Award **one** mark for any of the following: • 15 (1)		**(1)**

Question Number	Answer	Additional Guidance	Mark
(ii)	Award **one** mark for any of the following: • 11 (1)		**(1)**

Question Number	Answer	Additional Guidance	Mark
(iii)	Award **one** mark for any of the following: • 16, 17, 18, 19 (1) • 16 - 19 (1) • 16, 18 (1)	Do not accept 16 on its own.	**(1)**

Q4.

Question Number	Answer	Additional Guidance	Mark
(i)	Award **one** mark for each correct cell: **Algorithm type** / **Characteristic** / **Algorithm name** Search / Is a divide and conquer algorithm / Binary search (1) Sort / Is **not** a divide and conquer algorithm / Bubble sort (1)	Ignore exclusion of the words 'search' and 'sort' in the last column	**(2)**

Question Number	Answer	Additional Guidance	Mark
(ii)	Award up to **two** marks for a linked explanation, such as: • A sorting algorithm executes more quickly (1) because a small number of comparisons are made (1) • A sorting algorithm executes more slowly (1) because a large number of comparisons are made (1)	For both marks, the expansion must follow/associate with the statement.	**(2)**

Q5.

Question Number	Answer	Additional Guidance	Mark
(i)	Award **one** mark for each: • True (1) • False (1)	Do not award 0, 1, Yes, No Allow T/F for True/False	**(2)**

Question Number	Answer	Additional Guidance	Mark
(ii)	**The only correct answer is C** *A is not correct because 0.5 is the fractional part of division* *B is not correct because 1 is the remainder after integer division, i.e. the result of modulus* *D is not correct because 2.5 is the result of division*		**(1)**

Q6.

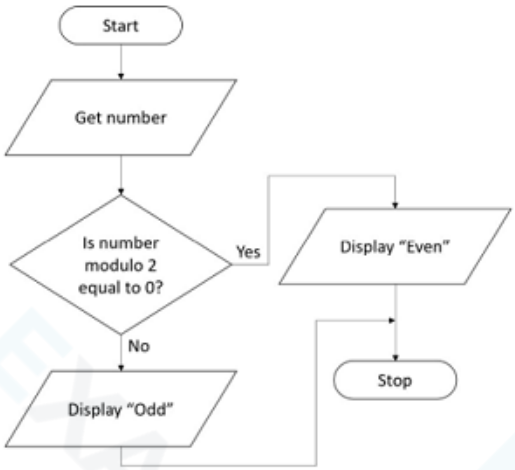
<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>(i)</td><td>Award one mark for each correct cell:

<table>
<tr><th></th><th>Line numbers</th></tr>
<tr><td>Traverses the records</td><td>17 (1)</td></tr>
<tr><td>Traverses the fields</td><td>19 (1)</td></tr>
</table>
</td><td></td><td>(2)</td></tr>
</table>

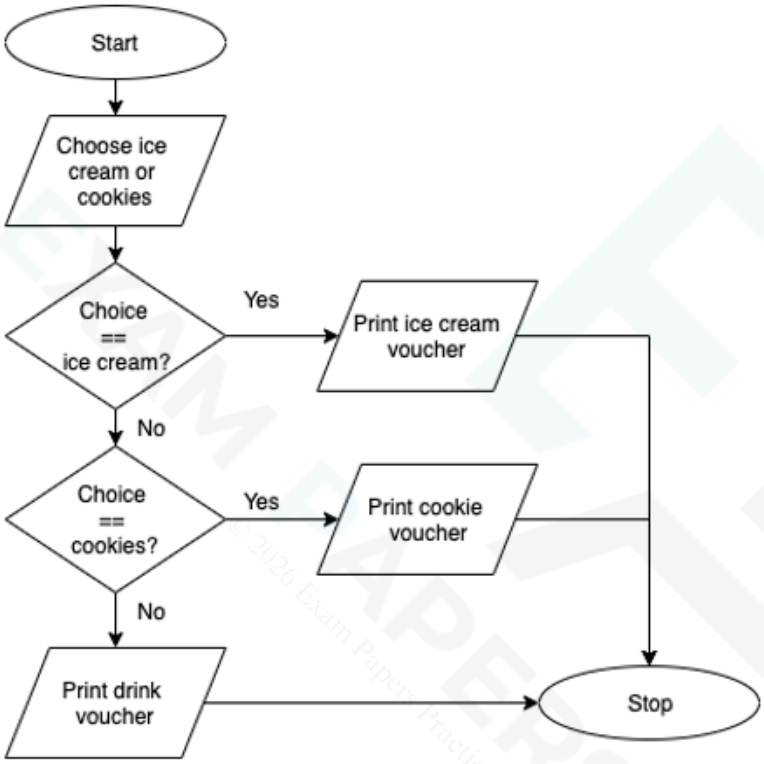
Question Number	Answer	Additional Guidance	Mark
(ii)	Award **one** mark for any of the following: • if (item > currMax): (1) • item > currMax (1)	Do not award relational symbol only, as there is no indication which line the error is on Allow equivalent expressions: item >= currMax currMax < item currMax <= item	**(1)**

Q7.

Question Number	Answer	Additional Guidance	Mark

	Award **one** mark for each of:
	
	• Input symbol to get number from user (1)
	• Decision symbol (disregard test inside), with exactly two arrows and labels for yes and no (1)
	• Correct relational test matches labels and output messages (1)
	• Accurately connected (1):
	  ○ outputs 'Display Even' **or** 'Display Odd', not both
	  ○ no hanging symbols
	  ○ connection to terminator
	
	 
	
		Arrows must match symbols for use of symbol to be awarded
		
		Each symbol should have only a single input arrow
		
		Input/output symbols should have only a single output arrow
		
		The decision symbol must have exactly two output arrows
		
		Relational tests:
		• X % 2 == 0
		• X MOD 2 == 0
		• X modulus 2 == 0
		
		Allow = for == in relational test
			(4)

Q8.

Question number	Answer	Additional guidance	Mark
	• Correct message in output box acting as a prompt for the user (1). • Correct diamond symbol for decision (1). • Correct test 'Choice == cookies?' for decision (1). • Correct label 'Yes' on right arrow AND Correct label 'No' on bottom arrow (1). • Correct output symbol with suitable message (1). • Correct ellipse symbol and 'stop' for terminator (1). 	• Symbol and contents are awarded independently. • Award 'End', 'Stop', 'Start' and 'Begin' as text for terminator symbols. • Award '==' and '=' used for equivalence inside decision symbol, but not in process symbol. • Accept 'Input choice' as an alternative response in the top process symbol	**(6)**

Q9.

Question Number	Answer	Additional Guidance	Mark
	One for each correct cell. **Input** / **Output** 200 – Too high (1) 33 – Good choice (1) 100 – Nice round number (1) 0 – Too low (1)	• Ignore spelling	**4**

Q10.

Question number	Answer	Additional guidance	Mark
	One mark for each correct cell **Input** / **Output** 2 – 7:30am 7 – 8am 8 – 7am		**3**

Q11.

Question Number	Answer	Additional Guidance	Mark
	Award **one** mark for each point for a maximum of **six** marks: • 0 and 10 in first row (1) • Evidence of a calculation for midpoint that would result in 5 shown in first row (1) • N and lower in first row (1) • 6, 10, calculation/8, N, higher in second row (1) Rounding down method • 6, 7, calculation/6, N, lower in third row (1) • 7, 7, calculation/7, N, higher/none/<blank> in fourth row (1) Rounding up method • 6, 7, calculation/7, N, higher in third row (1) • 6, 6, calculation/6, N, lower/none/<blank> in fourth row (1)	Penalise only once on bullet 2 if mid-point value given instead of calculation shown For BP2 there needs to be some evidence of a calculation that would yield the result 5 for the first midpoint. Allow midpoint calculation method to be either rounding down or rounding up.	**(6)**

Method 1: DIV 2 rounding down

Start index	End index	Calculation of mid-point	Found Y or N	Discard lower, higher or none
0	10	(0 + 10) // 2 = 5	N	lower
6	10	(6 + 10) // 2 = 8	N	higher
6	7	(6 + 7) // 2 = 6	N	lower
7	7	(7 + 7) // 2 = 7	N	(higher) none/<blank>
7	6			

Method 2: rounding up midpoint calculation

Start index	End index	Calculation of mid-point	Found Y or N	Discard lower, higher or none
0	10	(0 + 10) / 2 = 5	N	lower
6	10	(6 + 10) / 2 = 8	N	higher
6	7	(6 + 7) / 2 = 6.5 → 7	N	higher
6	6	(6 + 6) / 2 = 6	N	(lower) none/<blank>
7	6			

Q12.

Question number	Answer	Additional guidance	Mark
	Award **1** mark for each correct cell.		**(3)**

First	f
Second	c
Third	a
Fourth	b

Q13.

Question Number	Answer	Additional Guidance	Mark
	Award **one** mark for each of the following: • Number column is correct (goes from 1-6) (1) • Rows where number is 1, 2, 3 are correct (1) • Rows where number is 4, 5, 6 are correct (1) • Single accurate output only (1) **number** / **remainder** / **total** / **output** 0 1 / 0 / 1 2 / 0 / 3 3 / 0 / 6 4 / 2 / 6 5 / 1 / 6 6 / 0 / 12 12	Interpret empty cells as unchanged values Ignore spacing / empty cells on first bullet as long as numbers in sequence Other formats of trace table are acceptable	**(4)**

Q14.

Question Number	Answer	Additional Guidance	Mark
	One mark for each correct row showing relationship of variable changes. **count** / **accept** / **reject** / **weight** / **Display** 0 / 0 / 0 / 0 / 1 / 1 / / 404 / 2 / / 1 / 393 / 3 / 2 / / 395 / 4 / 3 / / 405 / / / / / 3 1	Different versions of trace tables are acceptable. Blanks can be replaced with contents of previous row. Final '3 1' can be included on row above. Ignore formatting of 3 1 in Display column (e.g. accept '3,1') Allow 'Enter weight of box:' in Display column until 3 1 should be displayed.	**6**

Q15.

Question number	Answer	Additional guidance	Mark
	1 mark for initialising all variables and **1** mark for each correct pass through the loop.	• Award alternative versions of the trace table if correct. For example, copying of values that do not change. • Passes are incorrect if display is indicated. • Display must be after the final pass (on a separate line in the table).	**(6)**

num	x	y	Display	Marks
0	999	0		(1)
355	355	355		(1)
554		554		(1)
199	199			(1)
409				(1)
			199 554	(1)

Q16.

Question Number	Answer	Additional Guidance	Mark
	One mark for each correct column	Allow follow through	**3**

A	B	C	(A AND B)	(NOT C)	(A AND B) OR (NOT C)
0	0	0	0	1	1
0	0	1	0	0	0
0	1	0	0	1	1
0	1	1	0	0	0
1	0	0	0	1	1
1	0	1	0	0	0
1	1	0	1	1	1
1	1	1	1	0	1

Q17.

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td></td><td>Award one mark for each of:

• Four unique rows for the values of S and M, indicating binary representation of 0, 1, 2, and 3 (1)
• Four correct results of applying AND to the values in S and M (1)
Example:
<table><tr><td></td><td></td><td></td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table></td><td>Ignore order of rows

Award both marks only if completely correct

Award carry through in final column, if error in S or M columns

Only award:
- 1/T/True/On
- 0/F/False/Off</td><td>(2)</td></tr>
</table>

Q18.

<table>
<tr><th>Question number</th><th>Answer</th><th>Additional guidance</th><th>Mark</th></tr>
<tr><td></td><td>One mark for each correct column
<table><tr><th>A</th><th>B</th><th>A OR B</th><th>NOT (A OR B)</th></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr></table></td><td></td><td>2</td></tr>
</table>

Q19.

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td></td><td>Award one mark for each of:

• S = A AND C (1)
• T = B OR C (1)
• Correct values in column Q (1)

<table>
<tr><th>A</th><th>B</th><th>C</th><th>S</th><th>T</th><th>Q</th></tr>
<tr><td></td><td></td><td></td><td>A AND C</td><td>B OR C</td><td>NOT S AND T</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
</table>
</td><td>Ignore capitalisation</td><td>(3)</td></tr>
</table>

Q20.

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>(i)</td><td>Award one mark for any of the following:

• Code that breaks/violates the rules/grammar of the programming language (1)</td><td>Do not award credit for an example in isolation such as missing colon, there must be a definition.</td><td>(1)</td></tr>
</table>

Question Number	Answer	Additional Guidance	Mark
(ii)	Award up to **two** marks for a linked explanation, such as: • The program crashes/stops (1) because the operation the computer is asked to do is impossible (1) • The program crashes (1) because the CPU cannot execute one of the instructions in the code (1)	For both marks, the expansion must follow/associate with the statement. Do not award credit for an example in isolation such as division by zero. Be careful not to award marks for syntax/logic error	**(2)**

Q21.

Question Number	Answer	Additional Guidance	Mark
	A definition to include: • The process of removing or hiding unnecessary (1) details so that only the important points remain (1) • Programmers can focus only on the important details of a problem (1) because abstraction allows them to ignore (1) any detail that is not relevant.		**2**

Q22.

Question Number	Answer	Additional Guidance	Mark
	Breaking down a problem/solution/system/algorithm		**1**

Q23.

Question number	Answer	Additional guidance	Mark
	One mark each for a maximum of 2 • A subprogram is a self-contained block (1) of code. • A subprogram performs a specific/ dedicated task (1). • It can be 'called' by the main program or other subprograms, when it is needed (1).		**2**

Q24.

Question Number	Answer	Additional Guidance	Mark
	Award up to **two** marks for a linked description, such as: • A list / data structure / collection of elements/fields/values (1) where each element/field/value may be a different data type / is related to the others (1)	For both marks, the expansion must associate with the statement.	**(2)**

Q25.

Question Number	Answer	Additional Guidance	Mark
	Award up to **four** marks for a linked description, such as: • Start at the first position / Iterate/Traverse (through the array) (1), compare the item with the target (1), stop when the target is matched (1), or stop when the end of the list is reached (and the item is not matched) (1)		**(4)**

Q26.

Question Number	Answer	Additional Guidance	Mark
	A linked description such as: • If the value of the item at the current index position in the array is more than the value of the target (1), no more passes of the loop are required / the loop will exit (1) • If the pass through the loop goes past the expected location of the target value (1), the third condition for the loop to execute will not be met (1) • If the value (of 'theTarget') is exceeded without reaching the end of the array (1) the algorithm will end (1)	Do not accept 'index' in place of 'the value at the index'.	**2**

Q27.

Question Number	Answer	Additional Guidance	Mark
	A description to include: • A syntax error is caused by using the grammar/form/words of the programming language incorrectly (1), whereas a logic error is caused by an error in the design of the algorithm (1) • A syntax error causes the program not to interpret/compile/translate, whereas a logic error produces an incorrect/wrong result (1)		**2**

Q28.

Question Number	Answer	Additional Guidance	Mark
	Indicative content: **Definition** • Decomposition is breaking down into smaller parts. Problems, solutions, and algorithms can be decomposed. • Abstraction is the process of removing or hiding unnecessary detail. **Benefits** • It is usually easier to solve several smaller problems, such as checking if touching a wall or updating the score display, than solve one big problem, such as making a game. • Different parts of the program can be shared between the class members to speed up development, for example, one group could work on the code to control the character, while another works on creating and playing the sounds. • Once all the pieces, like sounds, movement, and score are working correctly, the smaller solutions can be combined to make a larger solution, with fewer errors. • The individual parts of the program, such as updating the score can be		**(6)**

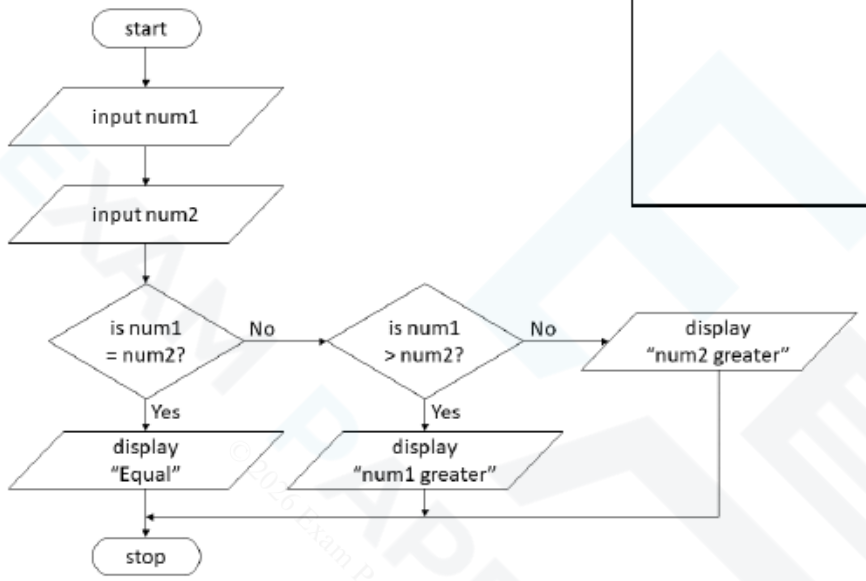
	ignored by the group of students writing the code for moving the character with the arrows / allowing each group of students to focus only on the small problem they have been given means time is not wasted on analysis not relevant to the solution. **Appear in program code** • Different blocks of code logic show decomposition, such as		

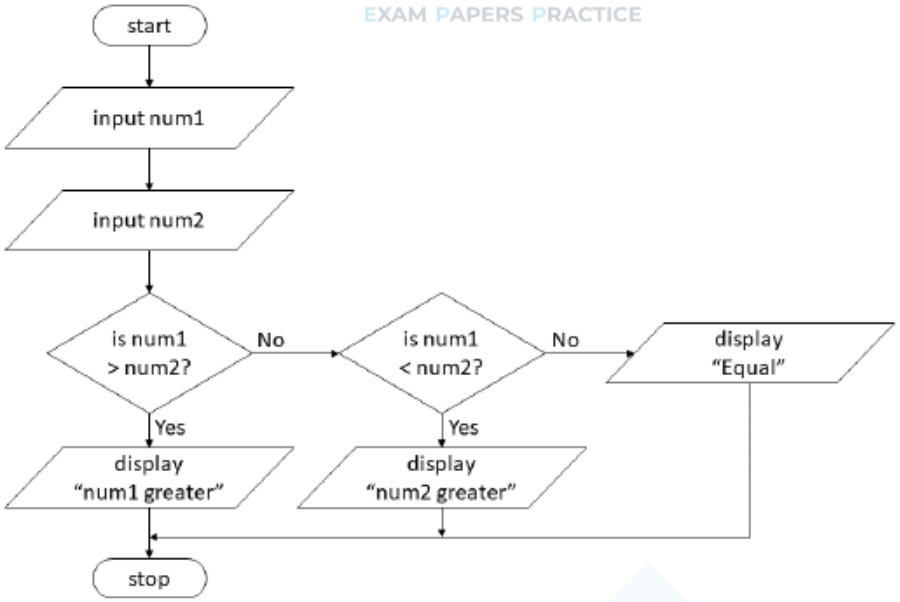
	moving the character and updating the score. These blocks could be shown separated by white space. • Subprograms are decompositions because they're blocks of code logic, separated from the main program. Subprograms could be used for updating the score and resetting the character back to the starting position. • Abstraction is used each time a subprogram, either built-in, in a library, or in the code file is called. Library routines in the game would include one to play the sounds and to get the keyboard input. • Subprograms are abstractions because their names hide how they work internally, even if their name is not descriptive. A subprogram to update the score should have a descriptive name, such as updateScore, but could just be called X. Either way, how it works internally is still hidden from the caller.		

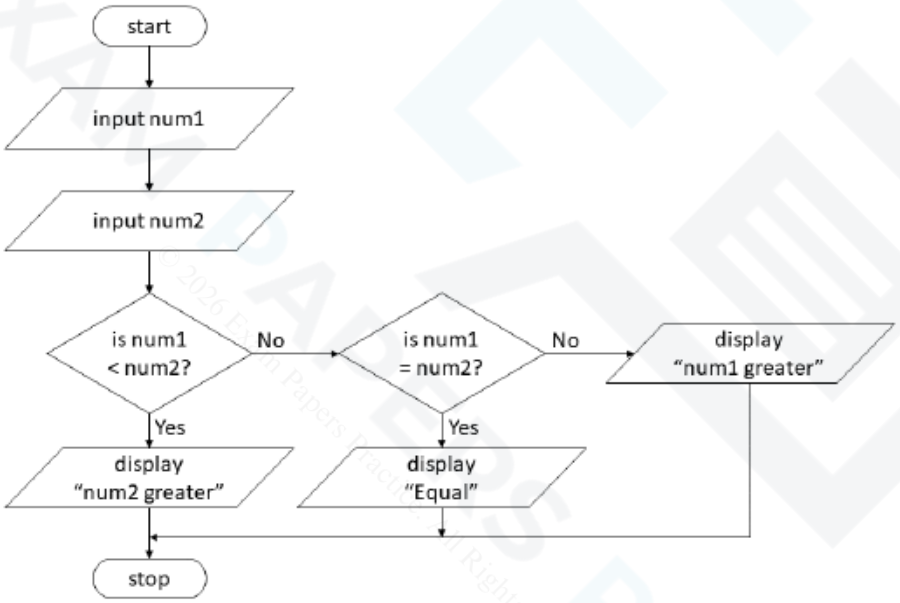
Level	Mark	Descriptor
	0	No rewardable content.
Level 1	1–2	Basic, independent points are made, showing elements of understanding of key concepts/principles of computer science. (AO1) The discussion will contain basic information with little linkage between points made or application to the context. (AO2)
Level 2	3–4	Demonstrates adequate understanding of key concepts/principles of computer science. (AO1) The discussion shows some linkages and lines of reasoning with some structure and application to the context. (AO2)
Level 3	5–6	Demonstrates comprehensive understanding of key concepts/principles of computer science to support the discussion being presented. (AO1) The discussion is well developed, with sustained lines of reasoning that are coherent and logically structured, and which clearly apply to the context. (AO2)

Q29.

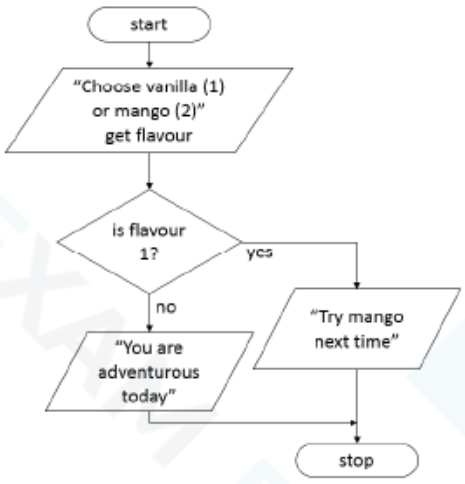
Question Number	Answer	Additional Guidance	Mark
	• Both start and stop terminators present **and** algorithm terminates correctly at the stop terminator (1) • Two separate inputs for two integers (1) [can be in the same input symbol] • Test for equivalence, greater than or less than in one decision box (1) • Different test (for equivalence, greater than, or less than) in separate decision box (1) • Two outputs, labelled Yes and No, for each decision box (1) • Logic of flowchart links to **three** logically valid outputs (1)	• Award 'End', 'Stop', 'Start' and 'Begin' as text for terminator symbols. • Award '==' and '=' used for equivalence inside decision symbol. • Accept 'Print' or 'Output' as an alternative to 'Display' in the output symbols. No quotes required around output string. • Accept True/False for Yes/No labels	**6**







Q30.

Question number	Answer	Additional guidance	Mark
	• Start and stop terminators in correct positions (1) • Decision symbol has two outputs only, i.e. the two output messages (1) • Yes/no labels on decision match output messages (1) • Fully connected to function correctly (1) 		4

Q31.

Question Number	Answer	Additional Guidance	Mark
	An explanation to include **two** from: • The subprogram may be used more than once in a program (1) so that writing, debugging, testing will save time (1) • The subprogram performs one specific/contained task (1) so it can be moved away from the main program code (1) • Subprograms for common tasks can be stored in libraries and reused in other programs (1) so that they don't have to be re- written (1)		**2**

Q32.

Question Number	Answer	Additional Guidance	Mark
	An explanation to include **two** from: An integer is returned with `//` (1) because • `len(plants)` could be an odd number (1) • the value for position has to be an integer / not a decimal (1) e.g. If the length of the array is an odd number, normal division will return a real number (1) (which is not acceptable because) index values are integers (1)		**2**

Q33.

Question Number	Answer	Additional Guidance	Mark
	Award up to **two** marks for a linked explanation, such as: • It helps the programmer visualise the steps in a program / find errors/bugs / check the algorithm gives the correct output (1) because they can see the value a variable holds at a given point in an algorithm (1)	For both marks, the expansion must associate with the statement.	**(2)**

Q34.

Question Number	Answer	Additional Guidance	Mark
(i)	Award up to **two** marks for a linked explanation, such as: • When the target is the last item/is not in the array (1) because the number of comparisons equals the number of items in the array / every item in the array must be compared to the target (1)	For both marks, the expansion must follow/associate with the statement.	**(2)**

Question Number	Answer	Additional Guidance	Mark
(ii)	Award up to **two** marks for a linked explanation, such as: • Only one pass of the array is needed (1) because the array is already in order (1) • A single pass will be done (1) because after one pass it can detect if the array is already sorted (1) • No swaps are needed/detected (1) because the array is already sorted (1)	For both marks, the expansion must follow/associate with the statement.	**(2)**

Q35.

Question number	Answer	Additional guidance	Mark
	Award **1** mark for the identification of a reason (1) with a linked justification/exemplification (1), up to a maximum of **2** marks. • Constants (shown in all capitals) are less likely to be changed by accident or error (1), so algorithms that use them should be more robust (1). • If the value of a constant does have to be altered (1), only one change is required (on the line where it is created and set) (1). • Constants allow values to be replaced with a name/identifier (1), so code is easier to read/maintain (1).		**(2)**

Q36.

Question Number	Answer	Additional Guidance	Mark
	The only correct answer is C *A is not correct because abstraction is hiding or removing detail* *B is not correct because computation is the act of computing D is not correct because evaluation is reaching a conclusion*		**(1)**

Q37.

Question Number	Answer	Additional Guidance	Mark
	One mark for each in the correct location: • Relational (1) • Arithmetic (1) • Boolean/Logical (1)	• Ignore spelling	**3**

Q38.

Question number	Answer	Additional guidance	Mark
(i)	• The number of sides on the single dice/a number between 6 and 12 (1) • The user's guess of the roll (1)	• Award variable names provided they are meaningful and uniquely identify the inputs, e.g. theGuess, sides.	**2**

Question number	Answer	Additional guidance	Mark
(ii)	D roll <= 3 • A roll = 3 is not a relational operator, but an assignment • B roll == 3 is an equivalence check, which would not allow a guess for 1 and 2 • C roll > 3 is greater than, which would not allow a guess for 1, 2, or 3		1

Q39.

Question Number	Answer	Additional Guidance	Mark
	Any **two** from: • Number of legs • Number of antennae • Length / width / size • Number of wings • Size of head • Shape of body • Number of eyes • Size of mouth • Habitat	Accept any appropriate feature. Award two marks from the same bullet point for the identification of two different features e.g. number of legs (1), legs jointed/unjointed (1)	2

Q40.

Question number	Answer	Additional guidance	Mark
	Logic		(1)