Chhetri Academy

chhetriacademy.com

GCSE COMPUTER SCIENCE

1.2

Edexcel 1CP2 | Specification point 1.2.4

Be able to determine the correct output of an algorithm for a given set of data and use a trace table to...

Untiered

Answers and examiner comments

3	2019-2023	Untiered
question groups	available papers	tier

Using this booklet

1. Answer numbers match the companion question booklet.
2. Follow the original question number on marking tables.
3. Examiner comments are included where available.

Name _____ **Date** _____

Some entries use full source marking material in an appendix. Any unavailable mark schemes are identified beside the relevant answer.

Question Number	Answer	Additional Guidance	Mark
4(a)	Indicative content: ● This pseudo-code is not suitable for humans to read and understand / running all the individual instructions together on a single line makes this pseudo-code difficult for humans to read **but** algorithm is short and understandable in this form ● Use of English command words makes it easier to understand for humans ● To make this pseudo-code easier for humans to read and understand: o the variable name 'w' could be changed to something more meaningful, such as 'week' o comments could be used to explain what the code is doing / using # or // for comments o each instruction could be placed on a new line o indentation could be used to show the blocking of the constructs / ending words could show blocking of the code o additional white space / lines could be used to show blocks of code	Rewritten pseudo-code can be used as a supportive example	**6**

Level	Mark	Descriptor
	0	No rewardable content
Level 1	1-2	Basic, independent points are made showing elements of knowledge and understanding of key concepts/principles of computer science. The discussion will contain basic information with little linkage between points made.
Level 2	3-4	Demonstrates adequate knowledge and understanding of key concepts/principles of computer science. The discussion shows some linkages and lines of reasoning with some structure.
Level 3	5-6	Demonstrates comprehensive knowledge and understanding by selecting relevant knowledge and understanding of key concepts/principles of computer science to support the discussion being presented. The discussion shows a well-developed, sustained line of reasoning which is clear, coherent and logically structured.

<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>4(b)(i)</td><td>One mark for initialising all variables and one mark for each correct pass through the loop.

<table>
<tr><th>num</th><th>x</th><th>y</th><th>Display</th><th></th></tr>
<tr><td>0</td><td>999</td><td>0</td><td></td><td>(1)</td></tr>
<tr><td>355</td><td>355</td><td>355</td><td></td><td>(1)</td></tr>
<tr><td>554</td><td></td><td>554</td><td></td><td>(1)</td></tr>
<tr><td>199</td><td>199</td><td></td><td></td><td>(1)</td></tr>
<tr><td>409</td><td></td><td></td><td></td><td>(1)</td></tr>
<tr><td></td><td></td><td></td><td>199 554</td><td>(1)</td></tr>
</table></td><td>• 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)</td><td>6</td></tr>
</table>

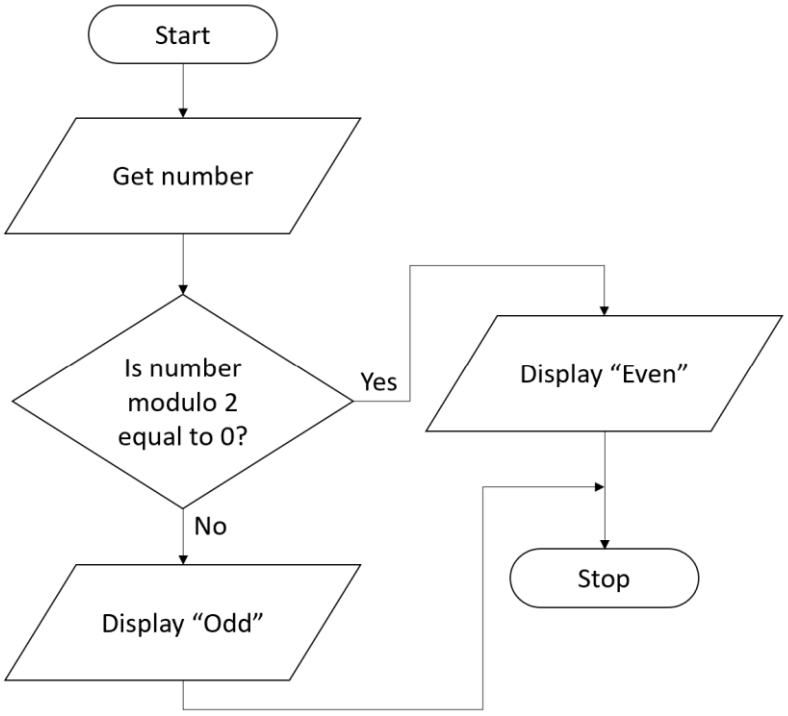
Question Number	Answer	Additional Guidance	Mark
4(b)(ii)	To identify the minimum/lowest number and the maximum/highest number (1)		**1**

Original question 5

The full source mark scheme begins on page 14. Use the original paper code, tier and question number shown above.

Question Number	Answer	Additional Guidance	Mark
4(a)	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)

Question Number	Answer	Additional Guidance	Mark
4(b)	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)

Question Number	Answer	Additional Guidance	Mark
4(c)	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): o outputs 'Display Even' **or** 'Display Odd', not both o no hanging symbols o 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)

Question Number	Answer	Additional Guidance	Mark
4(d)	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)

Question Number	Answer	Additional Guidance	Mark
4(e)	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:	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	(2)

S	M	S AND M
0	0	0
0	1	0
1	0	0
1	1	1

Question Number	Answer	Additional Guidance	Mark
4(f)	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 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		(6)

	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)

PMT



Mark Scheme (Results)

June 2022

Pearson Edexcel GCSE In Computer Science (1CP2/01) Paper 1: Principles of Computer Science

PMT

Edexcel and BTEC Qualifications

Edexcel and BTEC qualifications are awarded by Pearson, the UK's largest awarding body. We provide a wide range of qualifications including academic, vocational, occupational and specific programmes for employers. For further information visit our qualifications websites at www.edexcel.com or www.btec.co.uk. Alternatively, you can get in touch with us using the details on our contact us page at www.edexcel.com/contactus.

Pearson: helping people progress, everywhere

Pearson aspires to be the world's leading learning company. Our aim is to help everyone progress in their lives through education. We believe in every kind of learning, for all kinds of people, wherever they are in the world. We've been involved in education for over 150 years, and by working across 70 countries, in 100 languages, we have built an international reputation for our commitment to high standards and raising achievement through innovation in education. Find out more about how we can help you and your students at: www.pearson.com/uk

June 2022
Publications Code 1CP2_01_rms_20220825
All the material in this publication is copyright
© Pearson Education Ltd 2022

General Marking Guidance

- All candidates must receive the same treatment. Examiners must mark the first candidate in exactly the same way as they mark the last.
- Mark schemes should be applied positively. Candidates must be rewarded for what they have shown they can do rather than penalised for omissions.
- Examiners should mark according to the mark scheme not according to their perception of where the grade boundaries may lie.
- There is no ceiling on achievement. All marks on the mark scheme should be used appropriately.
- All the marks on the mark scheme are designed to be awarded. Examiners should always award full marks if deserved, i.e. if the answer matches the mark scheme. Examiners should also be prepared to award zero marks if the candidate's response is not worthy of credit according to the mark scheme.
- Where some judgement is required, mark schemes will provide the principles by which marks will be awarded and exemplification may be limited.
- When examiners are in doubt regarding the application of the mark scheme to a candidate's response, the team leader must be consulted.
- Crossed out work should be marked UNLESS the candidate has replaced it with an alternative response.

Question Number	Answer	Additional Guidance	Mark
1(a)(i)	Any **two** from: • Can share files/data • Can share applications/software • Can collaborate • Can share peripherals (printers, speakers) • Can share connectivity (Internet connection, hotspot) • Can access files **from any computer / hot desk** • Central control/management (security, software updates, backup, remote support, users, remote monitoring) • Allows communication (email, social media, video conferencing, online meeting, collaborative games)		**2**

Question Number	Answer	Additional Guidance	Mark
1(a)(ii)	A description such as: • A LAN covers a small (geographic) area / a WAN covers a large (geographic) area (2) • A LAN uses infrastructure owned by one company / a WAN uses third-party infrastructure (2) • A LAN connects individual devices / uses switches (1), whereas a WAN connects LANs / uses gateways (1)	Do not accept 'smaller/larger' in reference to the physical size of the network.	**2**

PMT

Question Number	Answer	Additional Guidance	Mark
1(a)(iii)	B 17 080 000 *A 17 080 is not correct because it is 17.08 multiplied by 1000* *C 17 080 000 000 is not correct because it is 17.08 multiplied by 1000^3* *D 17 080 000 000 000 is not correct because it is 17.08 multiplied by 1000^4*	17 080 000 is 17.08 multiplied by 1000^2	**1**

Question Number	Answer	Additional Guidance	Mark
1(a)(iv)	The delay/amount of time between data/signal being sent and it being received	Do not accept reference to delay/lag on its own.	**1**

Question Number	Answer	Additional Guidance	Mark
1(b)i **Clerical**	Any **one** from: • IP (address) • MAC (address)		**1**

Question Number	Answer	Additional Guidance	Mark
1(b)ii	A Data being sent *B is not correct because it is located in the header* *C is not correct because it is located in the header* *D is not correct because it is not located in a packet*		**1**

Question Number	Answer	Additional Guidance	Mark
1(c)	A description to include two from: • Monitors/checks traffic (1) using a set of rules / list of authorised/unauthorised computers/addresses/protocols (1) to decide if data is allowed into or out of the network (1)		**2**

Question Number	Answer	Additional Guidance	Mark
1(d) **Clerical**	Any **one** from: • POP3 • IMAP • SMTP	Accept POP	**1**

Question Number	Answer	Additional Guidance	Mark
1(e)	A linked explanation such as: • All communication could fail (1) because/if the central device fails (1) • It can be difficult/expensive to set up (1) because each device needs a cable to connect to the central device (1) • The number of devices that can be connected is limited (1) because the central device supports a fixed number of connections (1)	For 'central device': • Accept 'router' or 'hub' (as a term to refer to a device that contains a switch). • Accept 'switch' (as this is the correct term to refer to the central device) • Do not accept 'server'	**2**

Question Number	Answer	Additional Guidance	Mark
2(a)i	Any **one** from: • To represent an instruction/command • To make an instruction/command easy to read/write/learn/remember/understand • An instruction in a set of instructions	Accept generic definitions of mnemonic such as: 'To shorten a piece of information so it is easy to remember.'	**1**

Question Number	Answer	Additional Guidance	Mark
2(a)ii **Clerical**	Assembly language		**1**

Question Number	Answer	Additional Guidance	Mark
2(a)iii	A linked description such as: • They are microprocessor/CPU/machine specific (1) so they can manipulate the hardware directly (1) • They can be highly optimised (1) to make efficient use of the hardware/execute more quickly/use minimal memory (1) • Each line of code (1) is one instruction only (1) Award responses that refer to machine code as a low level language, such as: • **Machine code** is written in binary (1) so instructions can be processed **directly** by the CPU / do not need to be translated (1) Award **one** mark for: • Machine code does not need to be translated (1)	Accept: (Assembly language) instructions are assembled (1) to machine code (1)	**2**

Question Number	Answer	Additional Guidance	Mark
2(a)iv **Clerical**	Interpreter		**1**

Question Number	Answer	Additional Guidance	Mark
2(b)	• Data • Instructions		**2**

Question Number	Answer	Additional Guidance	Mark
2(c)i	A description to include **two** from: A laser heats/burns (areas of a disk) (1), which creates lands/pits (1) representing binary / 1s and 0s (1)		**2**

Question Number	Answer	Additional Guidance	Mark
2(c)ii **Clerical**	Solid state / flash *Accept 'SSD' for solid state*		**1**

Question Number	Answer	Additional Guidance	Mark
2(d)	Any **two** from: • Limited processing resources • Low power consumption • Simple/robust operating system • Low cost per unit • Smaller in size • Less storage • Limited user interface	Do not accept: 'Carries out a specific task': This is the opposite of 'general purpose', which is given in the question.	**2**

Question Number	Answer	Additional Guidance	Mark
2(e) **Clerical**	A Reduces data transfer times C Reduces required storage space *B is not correct because parts of a compressed file can still be written to several different places on a disk* *D is not correct as compression does not make data more secure like encryption does* *E is not correct because error detection/correction mechanism are still needed in network packets*		**2**

Question Number	Answer	Additional Guidance	Mark
2(f)	Any **two** from: • Programming language specific vulnerabilities (directly manipulating memory) • Security (vulnerability) • Weak design • Insufficient testing • Bad programming practices (not following standards, using unvalidated 3rd party libraries) • Inadequate validation (that fails to prevent incorrect/inappropriate data input) • Inadequate authentication (that enables hackers to gain unauthorised access) • Sensitive data not encrypted		**2**

Question Number	Answer	Additional Guidance	Mark
2(g)	A linked explanation such as: • Users can be added/deleted (1) so multiple people can use the same computer (1) • Edit user permissions (1) so only specific users can securely access their storage space (1) • Control the amount of resources/storage each user can access (1) so the limited storage on the machine can be shared (1) • Enforce user permissions (1) so only certain users are allowed to install programs / access certain files (1)		**2**

Question Number	Answer	Additional Guidance	Mark
3(a)(i) **Clerical**	1 mark for each nibble: 0011(1) 1001(1)		**2**

Question Number	Answer	Additional Guidance	Mark
3(a)(ii) **Clerical**	1 mark for each digit in the correct location: 2D		**2**

Question Number	Answer	Additional Guidance	Mark
3(a)(iii)	A linked description such as: • Flip all the bits (1) and add one (1) • Copy/keep all the 0s from the right/LSB, up to and including the first 1 (1), then flip the remaining bits (1) • Flip all the bits from the left/MSB (1) up to, but not including, the 1 nearest the right/LSB (1)		**2**

Question Number	Answer	Additional Guidance	Mark
3(a)(iv)	A linked explanation such as: • There are not enough bits (1) to store the result (1) • An overflow occurs (1) because 1 + 1 = 0 with a 1 carry / because the result is more than 255 (1)		**2**

Question Number	Answer	Additional Guidance	Mark
3(b) **Clerical**	Lossy		**1**

Question Number	Answer	Additional Guidance	Mark
3(c)	A description to include **two** from: • A standard • For representing text/characters • Allows data to be interchangeable between computers (from different manufacturers) • Each character is mapped to a unique number/binary equivalent • ASCII (encoding) uses 7/8 bits • There are 128 unique characters (allow 256 for 128)	Award a mark for an example of an ASCII character equivalence e.g. A = 65 as this can be awarded from mark point 4. (A = 01000001 could similarly be awarded.)	**2**

Question Number	Answer	Additional Guidance	Mark
3(d)(i)	Any **one** from: • The number of pixels **per** (square) **inch** is higher • The density of pixels is higher		**1**

Question Number	Answer	Additional Guidance	Mark
3(d)(ii)	Award **one** mark for each number in the correct location for a maximum of 4: • 15 in numerator (1) • 1028 x 640 in numerator (1) • 1024×1024 in denominator (1) • 8 in denominator (1) Example of expression that gains full marks: $\frac{1028 \times 640 \times 15}{1024 \times 1024 \times 8}$	Award equivalent representations of 1024 (2^{10})	**4**

Question Number	Answer	Additional Guidance	Mark
4(a)	A linked explanation such as: • It helps prevent loss of data (1) because malware could deliberately delete data / because data can be deleted/corrupted accidentally (1) • It helps minimise disruption following a system failure (1) because data can be restored (1) • Data can be restored (1) because physical hazards (fire/flood) (can damage storage devices) (1)		**2**

PMT

Question Number	Answer	Additional Guidance	Mark
4(b)	Any **two** from: • That they are giving consent • That they can say no / refuse • That they are not denied service, if they say no • What the data is being collected for • What processing will be done on their data/purpose of the processing/how data will be processed • That they can withdraw consent at any time • That the data will be shared (with third parties) • How long it will be stored • That the data will be stored securely		**2**

Question Number	Indicative content	Additional Guidance	Mark
4(c)	Discuss how algorithmic bias can affect the decisions police forces make. **Causes of algorithmic bias** • Machine learning having been trained using insufficient/inappropriate data • Human bias leading to discrimination and a lack of fairness • Poor design of the algorithm **Impacts on individuals and communities of algorithmic bias** • Deploying officers to the wrong areas • Individuals may feel as though they are being monitored/targeted/subjected to unfair treatment as a result of the output of a biased algorithm deploying too many officers / too few officers • Discrimination/equality (e.g. resourcing police more heavily in particular communities) • Feedback loops (if a model predicting crime rates based on historical arrest data is used to prioritise police resources, then arrests in high risk areas could increase further, reinforcing the imbalance) source **Methods available to reduce the risk of algorithmic bias** • Human oversight and 'sense checking' and confidence/error ratings of predictions • Governance (anticipating and managing risks and make sure legal requirements are adhered to) • Auditing to enable identification and mitigation of systematic bias • Open to scrutiny / transparent • Involvement of communities during development • Improve the training data		**6**

PMT

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)

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

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

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

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

PMT

Question Number	Answer	Additional Guidance	Mark
5(b)	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**

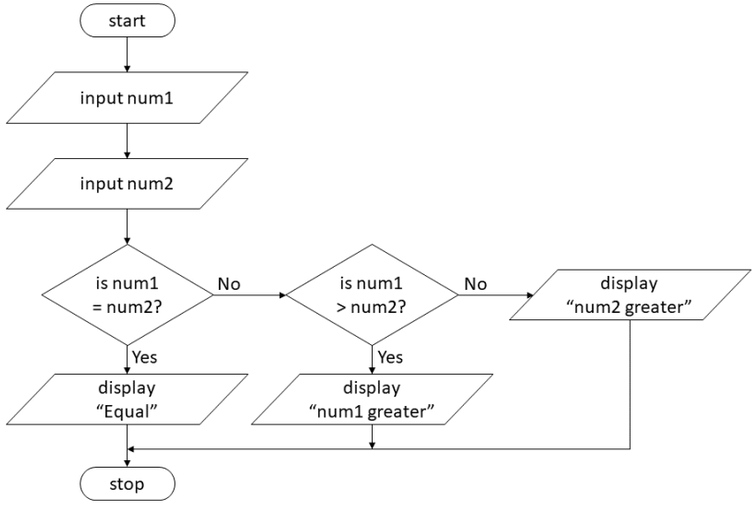
<table>
<tr><th>Question Number</th><th>Answer</th><th>Additional Guidance</th><th>Mark</th></tr>
<tr><td>5(c)</td><td>One mark for each correct row showing relationship of variable changes.
<table>
<tr><th>count</th><th>accept</th><th>reject</th><th>weight</th><th>Display</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>1</td><td>1</td><td></td><td>404</td><td></td></tr>
<tr><td>2</td><td></td><td>1</td><td>393</td><td></td></tr>
<tr><td>3</td><td>2</td><td></td><td>395</td><td></td></tr>
<tr><td>4</td><td>3</td><td></td><td>405</td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td>3 1</td></tr>
</table></td><td>Different versions of trace tables are acceptable.

Blanks can be replaced with contents of previous row.

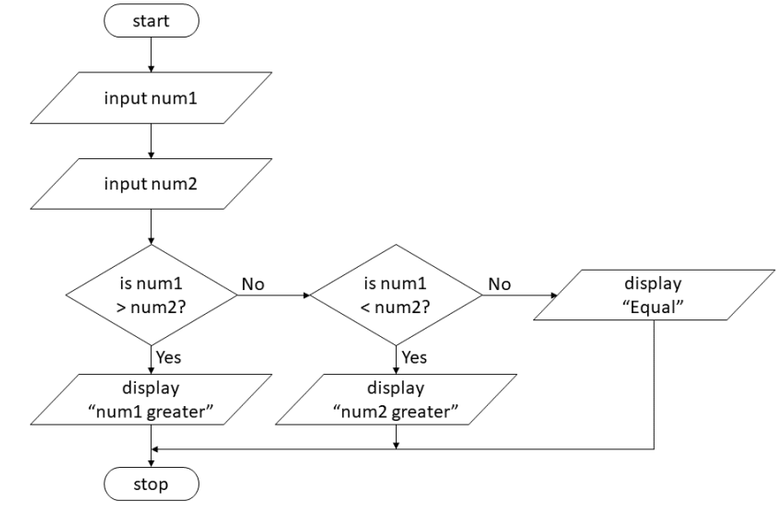
Final ‘3 1’ can be included on row above.

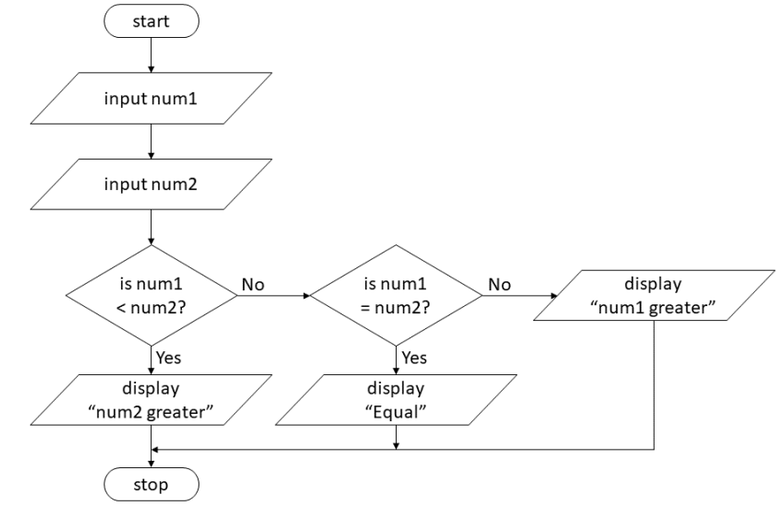
Ignore formatting of 3 1 in Display column (e.g. accept ‘3,1’)

Allow ‘Enter weight of box:’ in Display column until 3 1 should be displayed.</td><td>6</td></tr>
</table>

Question Number	Answer	Additional Guidance	Mark
5(d)	• 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**

PMT





20

1CP2-01

PMT

Question 5 (a)(i)

Two-thirds of candidates gained the mark for this question.

Question 5 (a)(ii)

Two-thirds of candidates gained the mark for this question.

Question 5 (a)(iii)

Two-thirds of candidates gained the mark for this question.

Question 5 (a)(iv)

Slightly more than half of candidates gained the mark for this question.

Question 5 (b)

Most candidates who gained a mark could state that the algorithm terminated due to the use of the passed variable.

Where marks were not awarded, candidates spent time discussing the found flag in place of the passed flag, repeated the question, and stated the target had been passed or simply reworded the algorithm.

(b) This algorithm searches a sorted array of numbers for a target value. The target value may or may not be in the array.

```
18  while ((index < len (theArray)) and (not found) and (not passed)):
19      if (theArray[index] == theTarget):
20          found = True
21          location = index
22      elif (theArray[index] > theTarget):
23          passed = True
24      index = index + 1
```

linear

The use of the found variable helps to make the algorithm efficient.

Describe how the use of the passed variable also helps to make the algorithm efficient.

(2)

If a certain amount of passes records the amount of passes, showing how much time it took to find the target value.

ResultsPlus Examiner Comments

Some responses did not address the requirements of the question.

Total: 0 Marks

(b) This algorithm searches a sorted array of numbers for a target value.
The target value may or may not be in the array.

```
18  while ((index < len (theArray)) and (not found) and (not passed)):
19      if (theArray[index] == theTarget):
20          found = True
21          location = index
22      elif (theArray[index] > theTarget):
23          passed = True
24      index = index + 1
```

The use of the `found` variable helps to make the algorithm efficient.

Describe how the use of the `passed` variable also helps to make the algorithm efficient.

(2)

If the array elements greater than the search algorithm passes the ~~found~~ theTarget then the algorithm stops searching and the program ends.

ResultsPlus
Examiner Comments

This is typical of responses that gained both marks.

Total: 2 Marks

Question 5 (c)

Many candidates were awarded the full six marks. Some lost a mark by not including the 0-count line.

There was a variety in the style and format of trace tables. It was common to see candidates writing in the display column with the accept, reject values or not using the count column.

The inputs are 404, 393, 395, 405.

Complete the trace table showing the execution of the program with these inputs.

You may not need to fill in all the rows in the table.

(6)

count	accept	reject	weight	Display
0	1	0	404	0, 0
1	1	1 & 0	395	1, 0
2	2	1	395	1, 0
3	3	1	405	2, 1
4	4	4	404	3, 1

ResultsPlus
Examiner Comments

No 0 row (no MP1)

First given row: Count value is off by one (no MP2) (Display values are also present for all rows, which is incorrect)

Second given row: follow through (FT) was applied on the count column, but reject should be 0 not 1 for MP3 (no MP3)(Display values are also present for all rows, which is incorrect)

Third given row: FT on count. Reject is correct... but accept value should be 1 (no MP4) (Display values are also present for all rows which is incorrect)

Fourth given row: FT on count, FT on accept, reject is still correct.... but weight is incorrect (no MP5)

Final given row: FT for count, FT for accept, reject still correct, weight correct Display column correct (awarded MP6)

Total: 1 Mark

The inputs are 404, 393, 395, 405.

Complete the trace table showing the execution of the program with these inputs.

You may not need to fill in all the rows in the table.

(6)

count	accept	reject	weight	Display
0	0	0	0	
0	1		404	
1		1	393	
2	2		395	
3	3		405	
				3, 1

ResultsPlus
Examiner Comments

All zeroes present. Awarded MP1 (1)

MP2 lost as count value is off by one (0)

The rest of the data is correct, aside from a repeated count error. Allow follow-through marks (third to sixth mark)

The comma was ignored in Display.

Total: 5 Marks

Question 5 (d)

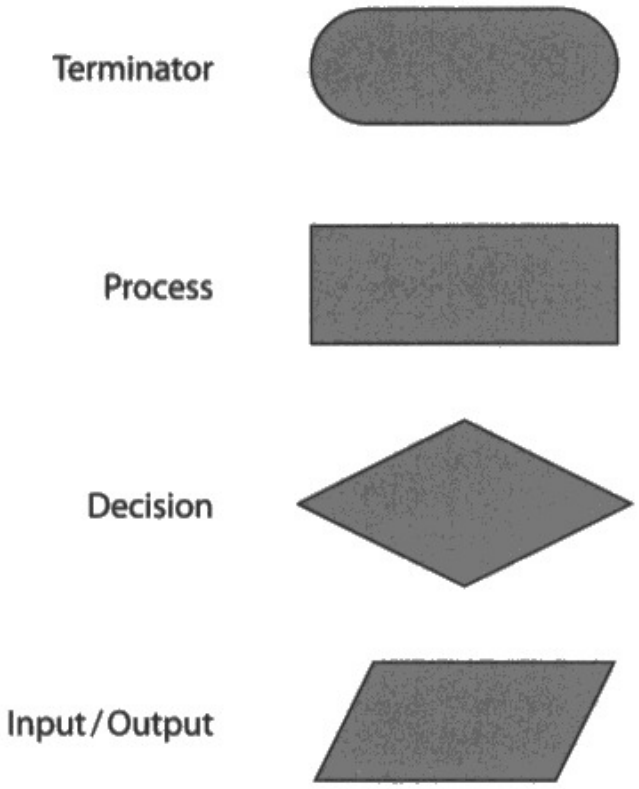
Candidates receiving four marks or fewer usually did not use the correct symbol for all flow chart symbols.

The common misconceptions are using processes for decisions and input/outputs. Also, multiple terminators for the END, which outputs did not link to.

Those candidates achieving more than four marks could link the flow lines to the correct shape.

(d) An algorithm is required that allows a user to enter two numbers. The algorithm then informs the user which number is greater, or whether the two numbers are equal. The algorithm is expressed in a flowchart.

Here are some flowchart symbols:





A typical full-mark response.

Total: 6 Marks

Paper Summary

Based on their performance on this paper, candidates are offered the following advice:

- Expanding and explaining answers using examples and reasons is required, especially where more than a simple statement or list is requested
- Where a context or scenario is provided, respond with the context of the question in mind
- Do not repeat responses when more than one example/reason is required
- Ensure responses match the requirements of the 'command' word
- Identify key words in the question to ensure that responses reflect the question that is asked

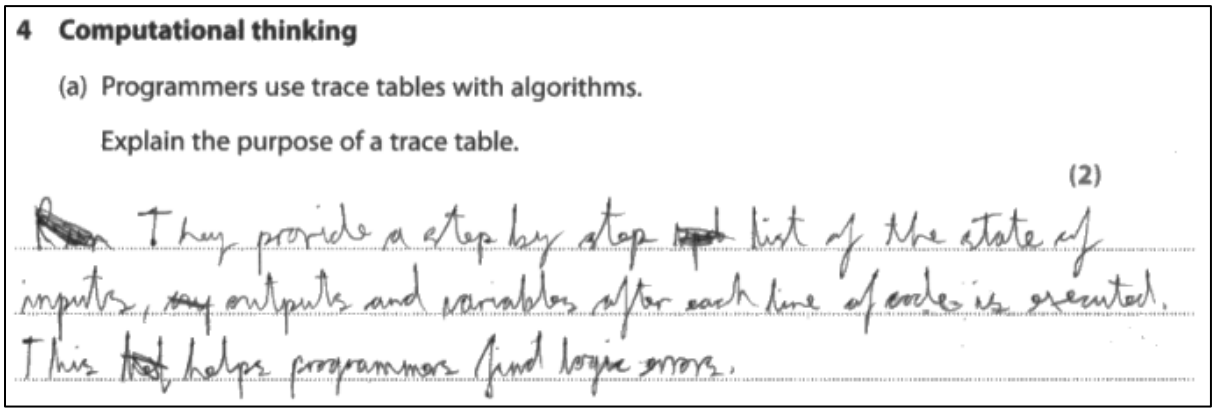
Grade boundaries

Grade boundaries for this, and all other papers, can be found on the website on this link:

https://qualifications.pearson.com/en/support/support-topics/results-certification/grade-boundaries.html

Question 4a

While some responses achieved both marks for a well-linked explanation, there were many that got trace tables confused with truth tables. Most correct responses mentioned that they are used to check the correct output, or test for bugs, but failed often to explain how, omitting the fact that a trace table tracks variables.



4 Computational thinking

(a) Programmers use trace tables with algorithms.

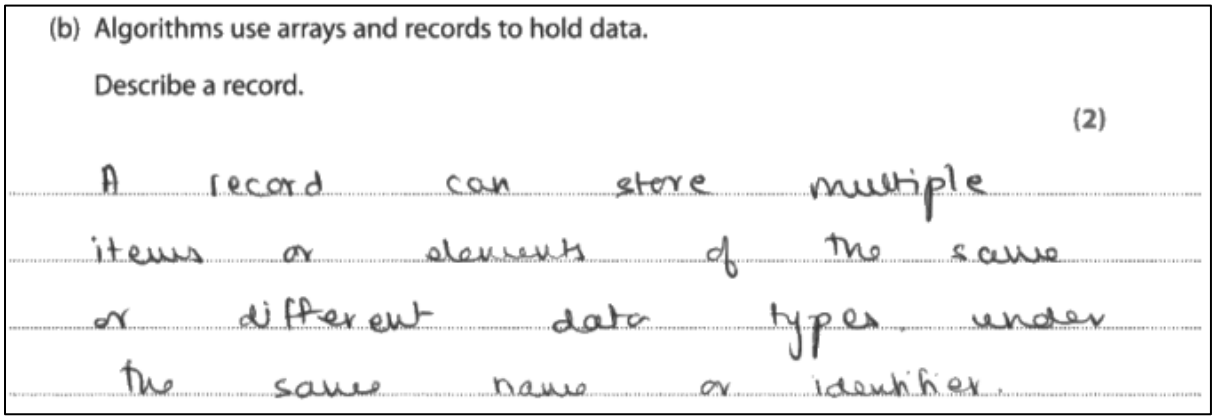
Explain the purpose of a trace table.

(2)

They provide a step by step list of the state of inputs, outputs and variables after each line of code is executed. This helps programmers find logic errors.

This response scored 2 marks. 'They provide …. the state of … variables' gained the first mark and ' helps programmers find logic errors' gained the second mark.

Lots of responses described storage or a storage method, rather than a data structure.


(b) Algorithms use arrays and records to hold data.

Describe a record.

(2)

A record can store multiple items or elements of the same or different data types, under the same name or identifier.

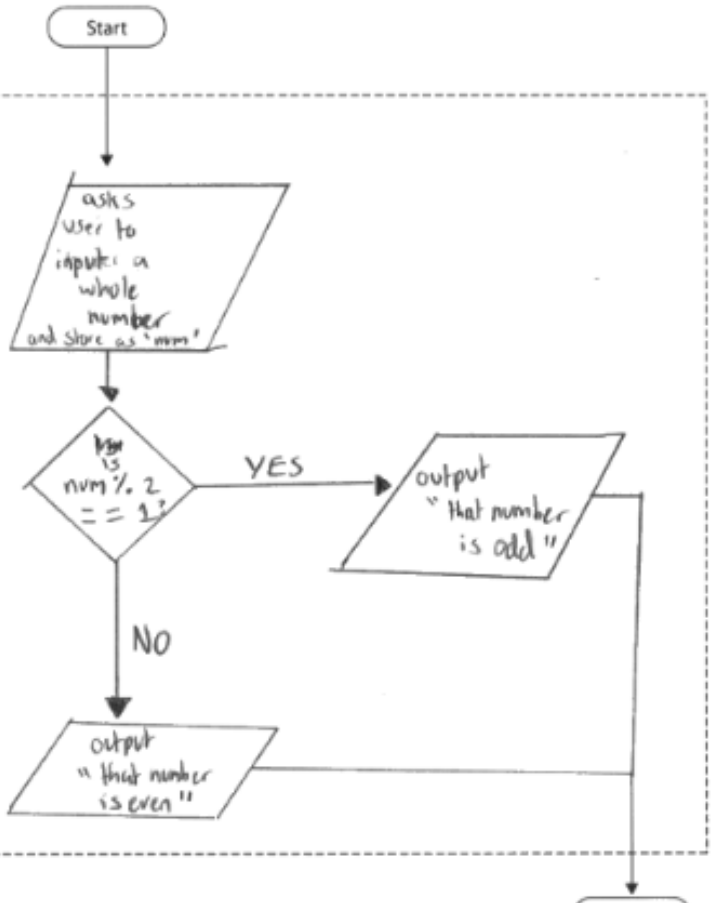
This response scored 2 marks; 1 mark for 'can store multiple items or elements' and 1 mark for 'different data types'.

Question 4c

Most responses correctly used the appropriate input symbol and had two outcomes from the decision. They understood the necessity of evaluating conditions and making choices based on them. Some students struggled to comprehend the relational test involving MOD, instead focusing on DIV, or whether the value divided by 2 had a remainder. Despite this, many students could make appropriate connections to the terminator and include at least one output message relevant to the decision outcome.
A few students used a decision shape instead of an output shape.

You may not need to use all the flowchart symbols.

(4)



This response scored all 4 marks.

Question 4d

Some responses methodically outlined the steps of the algorithm. However, the final mark point awarded for when the item is not found so the algorithm stops and the explicit point of comparing the item with the target were the most frequently missed marks. Many responses deviated from the intended focus by discussing sort algorithms.

(d) A linear search algorithm can be used on both a sorted and an unsorted array.

Describe how a linear search algorithm operates on an **unsorted** array.

(4)

It iterates on every value of an unsorted array and for every value checks if the value is equal to the value being searched for. If not, it will continue to the next variable until it does. If it reaches the end of the list and the value has not been found, it will declare that there is no such value

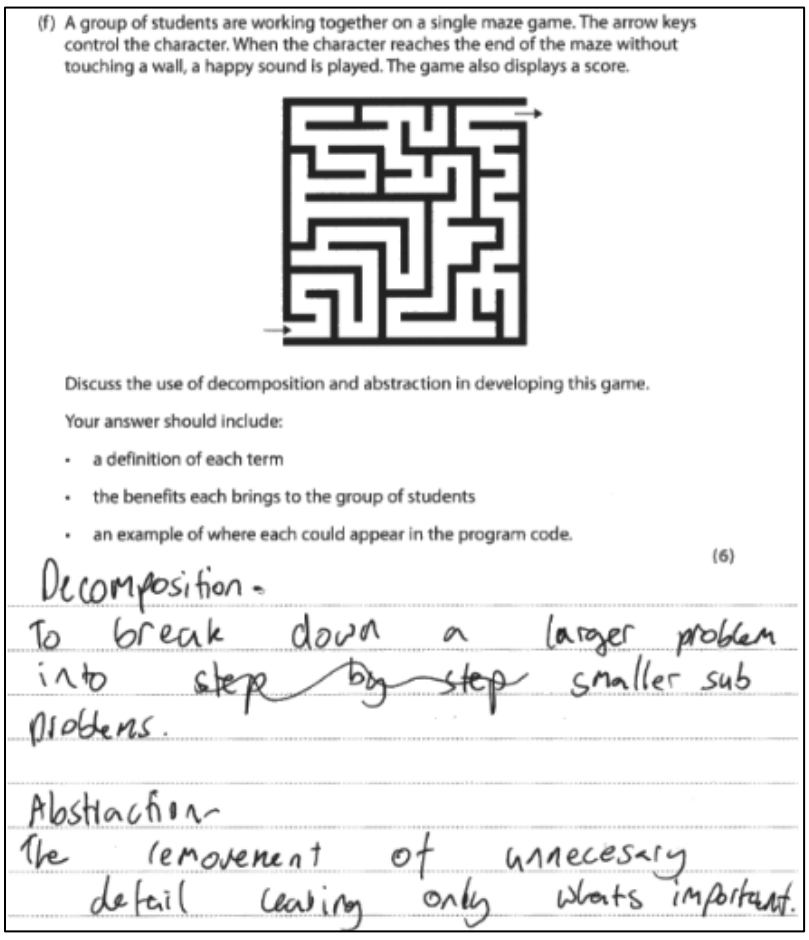
This response scored 4 marks. 'iterates' 1 mark, 'checks if the value is equal to the value being searched for' 1 mark, 'if not, it will continue until it [is equal to the value being searched for]' 1 mark and 'If it reaches the end of the list and the value has not been found' 1 mark.

Question 4e

A number of responses used ticks and crosses instead of 0, 1 etc. In general, students fully understood the logic process.

Question 4f

Despite being given bullet points to cover, many students did not use the structure of the question to compile their answer. Many cited vague benefits and although the definition for abstraction was often given, students were not clear on its benefits or its use. For example: some students thought abstraction would save on file size if a simplified character was used, while others talked about using or simplified maze, because they would be easier to navigate.



(f) A group of students are working together on a single maze game. The arrow keys control the character. When the character reaches the end of the maze without touching a wall, a happy sound is played. The game also displays a score.

Discuss the use of decomposition and abstraction in developing this game.

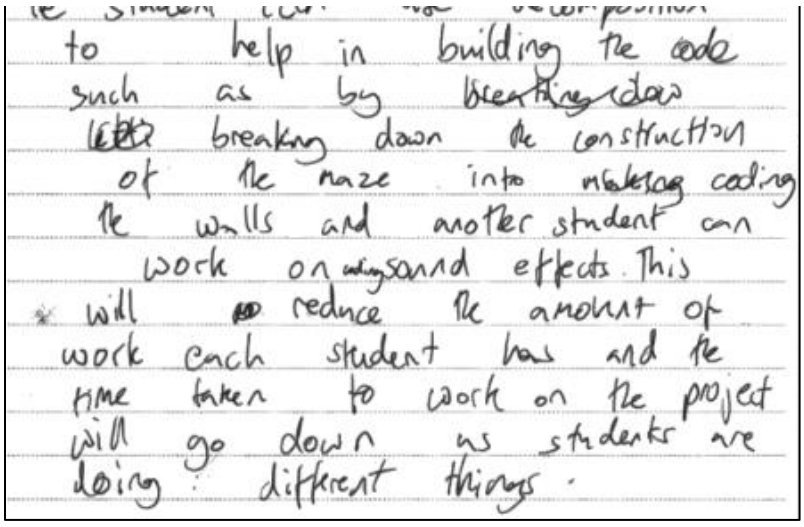
Your answer should include:

- a definition of each term
- the benefits each brings to the group of students
- an example of where each could appear in the program code.

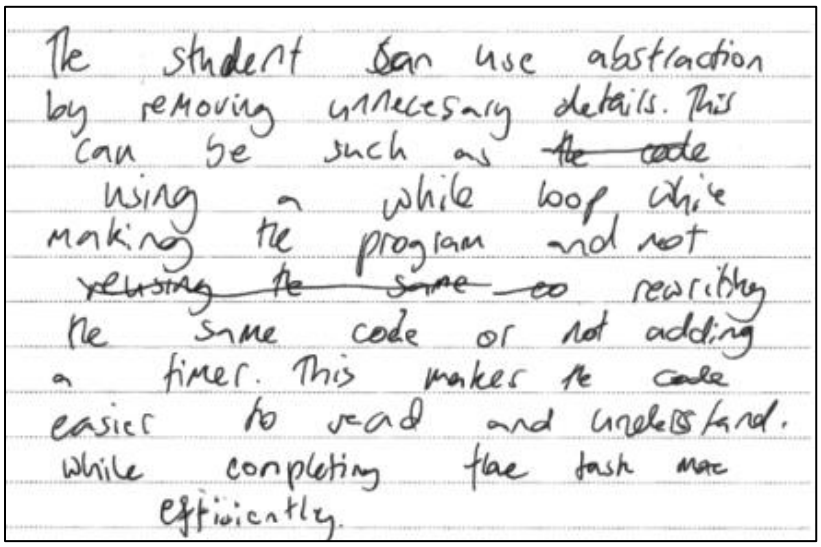
(6)

Decomposition -
To break down a larger problem into step by step smaller sub problems.

Abstraction -
The removement of unnecesary detail leaving only whats important.



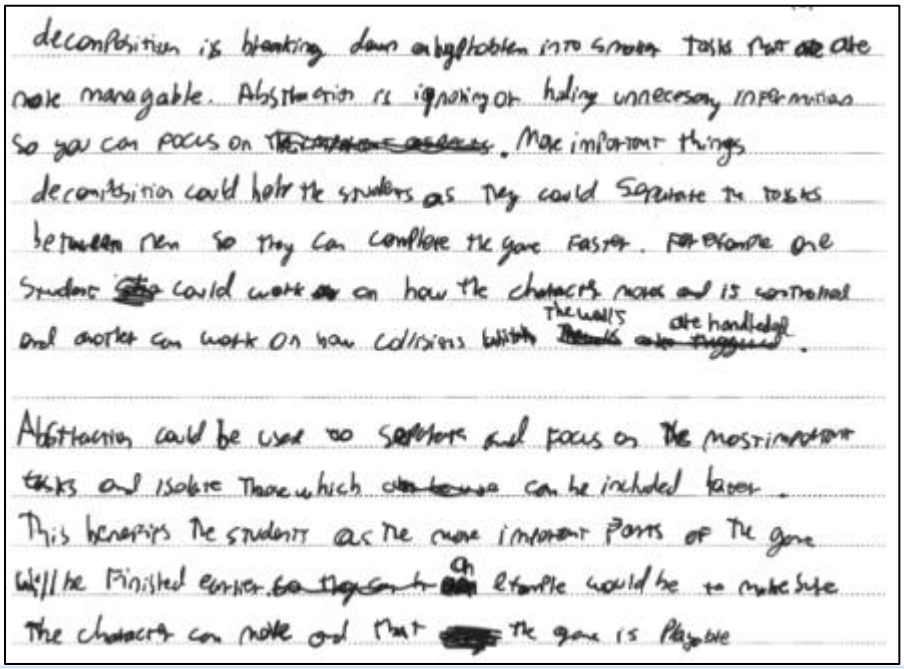
to help in building the code such as by ~~breaking down~~ breaking down the construction of the maze into ~~making~~ coding the walls and another student can work on ~~adding~~ sound effects. This will reduce the amount of work each student has and the time taken to work on the project will go down as students are doing different things.



The student can use abstraction by removing unnecesary details. This can be such as ~~the code~~ using a while loop while making the program and not ~~reusing the same co~~ rewriting the same code or not adding a timer. This makes the code easier to read and understand. While completing the task more efficiently.

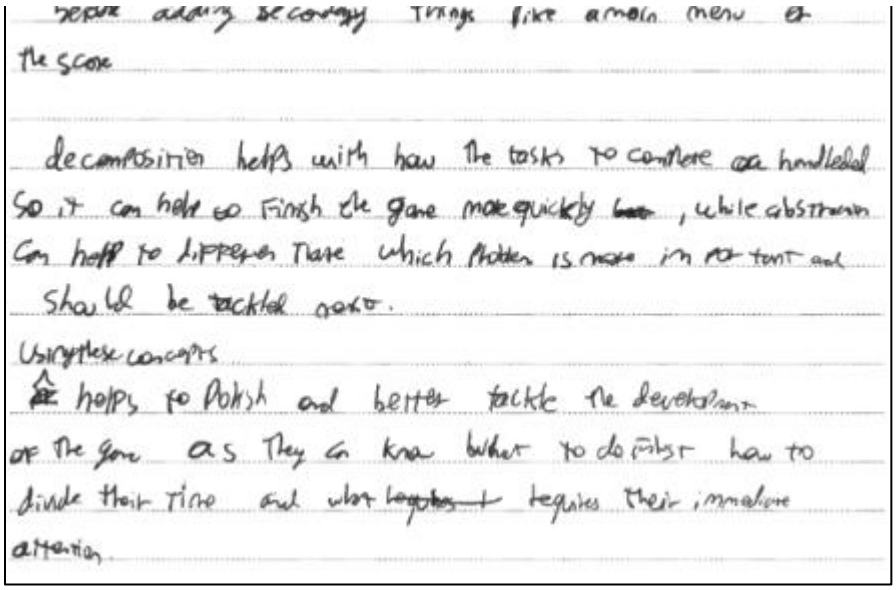
This response scored 3 marks. Both aspects are defined, and links are made to the benefits of decomposition. The benefits of abstraction are unclear.

Responses did not have to give the benefits of both, but if only one, the benefits of that one needed to be well-developed.



decomposition is breaking down a byproblem into smaller tasks that are more managable. Abstraction is ignoring or hiding unnecesary information so you can focus on ~~the important aspects~~. More important things decomposition could help the students as they could separate the tasks between them so they can complete the game faster. For example one student could work on how the characters moves and is controlled and another can work on how collisions with the walls are handledgd.

Abstraction could be used to separate and focus on the most important tasks and isolate those which can be included later. This benefits the students as the more important parts of the game will be finished earlier. An example would be to make sure the character can move and that the game is playable



before adding secondary things like a main menu or the score

decomposition helps with how the tasks to complete are handleded so it can help to finish the game more quickly ~~but~~, while abstraction can help to differen Tiate which problem is more important and should be tackled next.

Using these concepts helps to polish and better tackle the development of the game as they can know what to do first how to divide their time and what ~~requires~~ requires their immediate attention.

This response scored 4 marks. The response defines decomposition and abstraction and mentions some benefits to the group of students. However, there is some repetition of points made and no reference to the program code.

If a response only defined the terms, it can be no more than Level 1. For Level 2, the response had to provide benefits to the students - the definitions may have been implied.

For Level 3, a response had to provide at least one example of where decomposition or abstraction can appear in the program code. It is likely that a good example of either was enough to gain full marks, as long as the [possibly implied] definitions and benefits were also there.