

Please write clearly in	n block capitals.
Centre number	Candidate number
Surname	
Forename(s)	
Candidate signature	I declare this is my own work.

GCSE COMPUTER SCIENCE

Paper 1 Computational thinking and programming skills - Python

Wednesday 15 May 2024

Afternoon

Time allowed: 2 hours

Materials

- There are no additional materials required for this paper.
- You must not use a calculator.

Instructions

- Use black ink or black ball-point pen. Use pencil only for drawing.
- · Answer all questions.
- You must answer the questions in the spaces provided.
- If you need extra space for your answer(s), use the lined pages at the end of this book. Write the question number against your answer(s).
- Do all rough work in this book. Cross through any work you do not want to be marked.
- Questions that require a coded solution must be answered in Python.
- You should assume that all indexing in code starts at 0 unless stated otherwise.

Information

The total number of marks available for this paper is 90.



For Examiner's Use		
Question	Mark	
1		
2–3		
4–5		
6–7		
8–9		
10–11		
12		
13–14		
15		
TOTAL		

Advice

For the multiple-choice questions, completely fill in the lozenge alongside the appropriate answer.

CORRECT METHOD





If you want to change your answer you must cross out your original answer as shown.



If you wish to return to an answer previously crossed out, ring the answer you now wish to select as shown.



Do not write
outside the
box

Allowel all udeoliolis	Answer	all	questions
------------------------	--------	-----	-----------

0 1 Figure 1 shows an algorithm, represented using pseudo-code.

The algorithm assigns different values to two variables, then asks the user to input a letter.

Figure 1

0 1.1 Which pseudo-code statement assigns the length of the string film to a variable called value?

Shade one lozenge.

[1 mark]

- A film ← LEN(value)
- B film ← film + value ○
- C value ← film ○
- D value ← LEN(film)

0 1. 2 The POSITION subroutine returns the position of the first occurrence of a character in a string.

For example:

- POSITION("Godzilla vs. Kong", "o") would return 1
- POSITION("Godzilla vs. Kong", "z") would return 3

letter and film are variables used in the algorithm in Figure 1.

Complete the pseudo-code statement to find the position of the first occurrence of the contents of letter in film and store this position in the variable location

You **must** use the POSITION subroutine in your answer.

[1 mark]

location ← _____



				-
0 1.3	outs			Do not write outside the box
	Shad	le one lozenge.		[1 mark]
	Α	Boolean	0	
	В	character		
	С	integer	0	
	D	real	0	
0 1.4	Desc	ribe what is meant by an a	ssignment statement in a program.	[1 mark]
		Question 1 con	tinues on the next page	

outside the

box

0 1.5 Write a Python program that:

- gets the user to enter the name of a film
- displays You entered followed by the name of the film entered by the user.

The output from the program **must** be on one line.

You **should** use indentation as appropriate, meaningful variable name(s) and Python syntax in your answer.

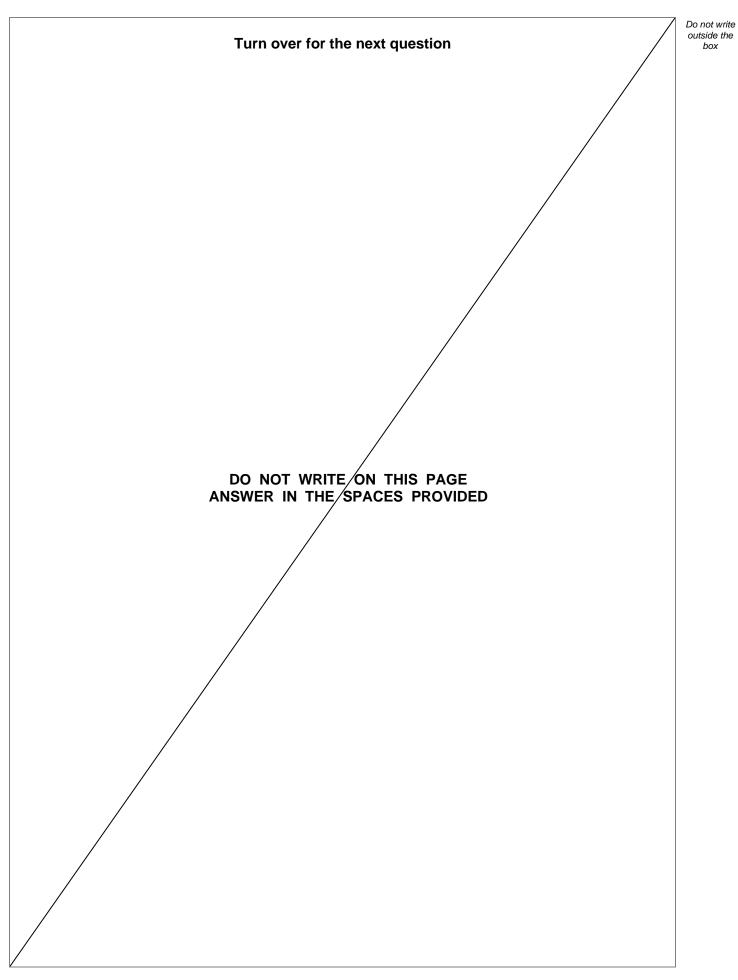
The answer grid below contains vertical lines to help you indent your code accurately.

[2 marks]



IB/G/Jun24/8525/1B

6







Do not write
outside the
box

0	2	Figure 2 shows an algorithm	n, represented using pseudo-code.
---	---	-----------------------------	-----------------------------------

• Line numbers are included but are not part of the algorithm.

Figure 2

```
1
      num ← USERINPUT
2
      IF NOT(num > 1) OR num > 20 THEN
         OUTPUT "False"
3
4
      ELSEIF num > 1 AND num < 15 THEN
5
         OUTPUT "Almost"
6
      ELSEIF num MOD 5 = 0 THEN
7
         OUTPUT "True"
8
      ELSE
9
         OUTPUT "Unknown"
10
      ENDIF
```

The modulus operator is used to calculate the remainder after dividing one integer by another.

For example:

- 14 MOD 3 evaluates to 2
- 24 MOD 5 evaluates to 4
- 0 2. 1 Where is a relational operator first used in the algorithm in Figure 2?

Shade **one** lozenge.

[1 mark]

Α	Line number 1	0
В	Line number 2	0
С	Line number 3	0
D	Line number 6	0



0 2 . 2	In the	e algorithm in Figure 2 , what will be t	he output when the user input is 5?	?	Do not write outside the box
	Shad	le one lozenge.		[1 mark]	
	Α	Almost	0		
	В	False	0		
	С	True	0		
	D	Unknown	0		
0 2.3	Whic Figu i	h value input by the user would resu re 2 ?	It in True being output by the algor	rithm in	
	Shad	le one lozenge.		[1 mark]	
	Α	-1	0		
	В	10	0		
	С	20	0		
	D	21	0		
0 2 . 4	Rewr	ite line 2 from the algorithm in Figur	e 2 without using the NOT operato	or.	
	The a	algorithm must still have the same fu	nctionality.	[1 mark]	
0 2 . 5		er inputs a value into the algorithm in		len oven	
	Siate	e one value that the user could input	ınat would result in an output of ∪n.	[1 mark]	



0 3 Figure 3 shows an incomplete Python program for a number guessing game.

Line numbers are included but are not part of the program.

Figure 3

```
1
      import random
2
3
      print("Enter a number")
4
      userNumber = int(input())
      while userNumber < 1 or userNumber > 100:
5
         print("Invalid number")
6
         userNumber = int(input())
7
8
      print("Valid number entered")
9
      if randomNumber == userNumber:
10
         print("Number guessed correctly")
```

The program should generate a random number between 1 and 100 (including 1 and 100). This will be the number the user has to guess.

Write the Python code that should be used on line 2 in Figure 3 to:

- generate a random number between 1 and 100 inclusive
- assign this number to the appropriate variable from the program.

You **must** use random.randrange(a, b) in your Python code.

random.randrange(a, b) generates a random integer in the range a to b starting at a but finishing one before b

[2 marks]



0 3.2

Complete the test plan in **Table 1** to test the validation of userNumber in the program in **Figure 3**. [2 marks]

Table 1

Test number	Test type	Test data	Expected result
1	Erroneous	150	
2	Boundary		
3	Normal		Valid number entered

0 3. In an earlier version of the program in **Figure 3**, **line 5** contained one syntax error and one logic error:

whil userNumber < 1 or userNumber >= 100:

Complete the table to describe the errors in the program on line 5.

[2 marks]

Error type	Description
Syntax error	
Logic error	

11



0 4.1	Define the term abstraction. [1 mark]	Do not write outside the box
0 4.2	State the name for the process of breaking a problem down into sub-problems. [1 mark]	



0 5

Figure 4 shows an algorithm, represented using pseudo-code.

Do not write outside the box

The algorithm calculates the total cost of hiring a hotel for a wedding.

Figure 4

```
numberOfRooms ← USERINPUT
charge ← 25
IF numberOfGuests > 50 THEN
  totalCost ← numberOfGuests * 2
ELSE
  IF numberOfGuests ≥ 25 THEN
     totalCost ← numberOfGuests * 4
  ELSE
     totalCost ← numberOfGuests * 5
  ENDIF
ENDIF
totalCost ← totalCost + (numberOfRooms * 100)
IF totalCost < 1400 THEN
  totalCost ← totalCost + charge
ENDIF
OUTPUT totalCost
```

Complete the table below using the algorithm in **Figure 4**.

[3 marks]

Input value for numberOfGuests	Input value for numberOfRooms	Output
50	30	
20	10	
500	5	





hox

0	6

A university is writing a program to calculate a student's total mark for three essays.

If any essays are handed in late, the total mark is reduced.

Write a Python program to calculate the total mark.

You should assume there are three integer variables called e1, e2 and e3 which have already been given values to represent the marks of the three essays.

The program should:

- get the user to enter the number of essays handed in late and store the number in a variable
- calculate the total mark for the three essays
 - o if only one essay is handed in late, the total mark is reduced by 10
 - o if more than one essay is handed in late, the total mark should be halved
 - o the total mark should **not** be less than 0
- output the total mark.

You **should** use indentation as appropriate, meaningful variable name(s) and Python syntax in your answer.

The answer grid below contains vertical lines to help you indent your code.

[7 marks]



l	l	1	

Turn over for the next question



0 7

A shop owner wants to create stock codes for each type of sweet they sell.

Do not write outside the

Figure 5 shows some of the sweets.

Figure 5

sweetID	sweetName	brand
S1	WINE GUMS	MAYNARDS
S2	COLA CUBES	BERRYMANS
S3	STARBURST	WRIGLEY

A stock code is made up of the:

- sweetID
- first letter and the second letter in sweetName
- first letter of the brand

For example:

- the stock code for WINE GUMS would be S1WIM
- the stock code for STARBURST would be S3STW

Write a Python program to create the stock code for a sweet.

The program should:

- get the user to enter the sweetID, sweetName and brand
- · create the stock code
- assign the stock code to a variable called code

You **should** use indentation as appropriate, meaningful variable name(s) and Python syntax in your answer.

The answer grid below contains vertical lines to help you indent your code.

[4 marks]





		Do not outside
		box
		11

Turn over for the next question



0 8 Figure 6 shows an algorithm, represented using pseudo-code.

Figure 6

The DIV operator is used for integer division.

Complete the trace table for the algorithm in **Figure 6**.

Part of the table has already been filled in.

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

[6 marks]

i	daysTotal	weeks		weeksTotal	
		[0]	[1]	[2]	weeksiotai
		0	0	0	



0 9.1	0 9. 1 Which of the following best describes a data structure?			
	Shac	le one lozenge.	[1 mark]	
	Α	A number with a fractional part	0	
	В	A value such as a whole number		
	С	All of the data used and stored within a program		
	D	An organised collection of values		
		Question 9 continues on the next page		



0 9 . 2

Figure 7 shows an incomplete algorithm, represented using pseudo-code.

The algorithm is used to store and manage books using records.

The algorithm should do the following:

- create a record definition called Book with the fields bookName, author and price
- create a variable for each book using the record definition.

Complete Figure 7 by filling in the gaps using the items in Table 2.

- You may need to use some of the items in **Table 2** more than once.
- You will not need to use all the items in Table 2.

[3 marks]

Table 2

1	2	author
В1	В2	Book
bookName	i	Real
OUTPUT	String	Boolean

Figure 7

RECORD
bookName : String
: String
price :
ENDRECORD
B1 ← Book("The Book Thief", "M Zusak", 9.99)
B2



0 9.3	Write an algorithm using pseudo-code to display the name of the most expensive book.	Do not write outside the box
	The algorithm should: • compare the price of B1 and the price of B2 • output the book name of the most expensive book • output Neither if the books are the same price.	
	The algorithm should work for any values stored in B1 and B2 [3 marks]	
		13

Turn over for the next question



1 0 Figure 8 shows a Python program.

Figure 8

State what will be displayed by the print statement when the subroutine First is called with the values 3, 4 and 4 for the parameters p1, p2 and p3

[1 mark]

1 0.2 State what will be displayed by the print statement when the subroutine First is called with the values 3, 4 and 8 for the parameters p1, p2 and p3

[1 mark]

Do not write outside the box Turn over for the next question DO NOT WRITE ON THIS PAGE ANSWER IN THE SPACES PROVIDED



1 1

A program is to be written to authenticate a username and password entered by the user.

Do not write outside the

Figure 9 shows the only two pairs of valid usernames and passwords.

Figure 9

Username	Password
Yusuf5	33kk
Mary80	af5r

Write a Python program to authenticate a username and password.

The program should:

- get the user to enter a username
- get the user to enter a password
- display the message Access denied if the username and password pair entered is not valid
- display the message Access granted if the username and password pair entered is valid
- repeat until a valid username and password pair is entered.

You **should** use indentation as appropriate, meaningful variable name(s) and Python syntax in your answer.

The answer grid below contains vertical lines to help you indent your code accurately.

[7 marks]

	1	





1 2

A program is being written to solve a sliding puzzle.

- The sliding puzzle uses a 3 x 3 board.
- The board contains eight tiles and one blank space.
- Each tile is numbered from 1 to 8
- On each turn, a tile can only move one position up, down, left, or right.
- A tile can only be moved into the blank space if it is next to the blank space.
- The puzzle is solved when the tiles are in the correct final positions.

Figure 10 shows an example of how the tiles might be arranged on the board at the start of the game with the blank space in the position (0, 1).

Figure 11 shows the correct final positions for the tiles when the puzzle is solved.

The blank space (shown in black) is represented in the program as number 0

Figure 10

column
0 1 2
0 4 2
row 1 1 7 6
2 5 3 8

Figure 11

		column		
		0	1	2
	0	1	2	3
row	1	4	5	6
	2	7	8	



Table 3 describes the purpose of three subroutines the program uses.

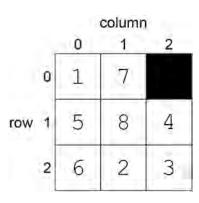
Table 3

	T
Subroutine	Purpose
<pre>getTile(row, column)</pre>	Returns the number of the tile on the board in the position (row, column)
	For example:
	• getTile(1, 0) will return the value 5 if it is used on the board in Figure 12
	• getTile(1, 2) will return the value 0 if it is used on the board in Figure 12 .
move(row, column)	Moves the tile in position (row, column) to the blank space, if the blank space is next to that tile.
	If the position (row, column) is not next to the blank space, no move will be made.
	For example:
	 move (0, 2) would change the board shown in Figure 12 to the board shown in Figure 13 move (2, 0) would not make a move if used on the board shown in Figure 12.
displayBoard()	Displays the board showing the current position of each tile.

Figure 12

column
0 1 2
0 1 7 4
row 1 5 8
2 6 2 3

Figure 13



Question 12 continues on the next page



1 2.1

The Python program shown in **Figure 14** uses the subroutines in **Table 3**, on page 25.

The program is used with the board shown in Figure 15.

Figure 14

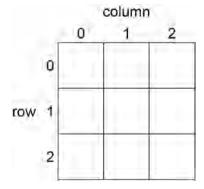
```
if getTile(1, 0) == 0:
    move(2, 0)
if getTile(2, 0) == 0:
    move(2, 1)
displayBoard()
```

Figure 15

		column		
	-	0	1	2
	0	1	8	3
row	1		7	5
	2	4	2	6

Complete the board to show the new positions of the tiles after the program in **Figure 14** is run.

[2 marks]





IB/G/Jun24/8525/1B

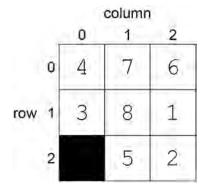
Figure 16 shows part of a Python program that uses the getTile subroutine from **Table 3**, on page 25.

The program is used with the board shown in Figure 17.

Figure 16

```
for i in range(3):
    for j in range(3):
        if getTile(i, j) == 0:
        ref1 = i
        ref2 = j
```

Figure 17



1 2.2 Which **two** of the following statements about the program in **Figure 16** are **true** when it is used with the board in **Figure 17**?

Shade two lozenges.

[2 marks]

- A Nested iteration is used.
- **B** The final value of ref1 will be 0
- C The number of comparisons made between getTile(i, j) and 0 will be nine.
- **D** The outer loop, for i in range(3), will execute nine times.
- E The values of i and j do not change when the program is executed.

Question 12 continues on the next page

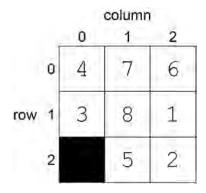


Figure 16 and Figure 17 are repeated below.

Figure 16

```
for i in range(3):
    for j in range(3):
        if getTile(i, j) == 0:
        ref1 = i
        ref2 = j
```

Figure 17



1 2.3	Explain the purpose of the first iteration structure in the program in Figure 16	[1 mark]
1 2.4	Explain the purpose of the second iteration structure in the program in Figure	• 16. [1 mark]
1 2 . 5	State the purpose of the program in Figure 16 .	[1 mark]



Do not write outside the box Question 12 continues on the next page DO NOT WRITE ON THIS PAGE ANSWER IN THE SPACES PROVIDED





1 2 . 6

Table 4 shows a description of the getTile subroutine previously described in more detail in **Table 3**, on page 25.

Do not write outside the

Table 4

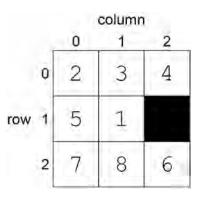
Subroutine	Purpose
<pre>getTile(row, column)</pre>	Returns the number of the tile on the board in the position (row, column)

Figure 18 and Figure 19 show example boards.

Figure 18

column
0 1 2
0 5 2
row 1 1 3 4
2 6 7 8

Figure 19



Write a Python program to:

- check that in the first row:
 - o the second tile number is one more than the first tile number
 - $\circ\,$ the third tile number is one more than the second tile number
- display Yes when the row meets both conditions above
- display No when the row does not meet both conditions above.

For example:

- for the board in Figure 18, the program would display No
- for the board in Figure 19, the program would display Yes

You **must** use the getTile subroutine in your Python code.

You **should** use indentation as appropriate, meaningful variable name(s) and Python syntax in your answer.

The answer grid below contains vertical lines to help you indent your code accurately.

[4 marks]



	Т	

Question 12 continues on the next page





1 2 . 7 Table 5 describes the purpose of another two subroutines the program uses.

outside the hox

Table 5

Subroutine	Purpose
solved()	Returns True if the puzzle has been solved. Otherwise returns False
<pre>checkSpace(row, column)</pre>	Returns True if there is a blank space next to the tile on the board in the position (row, column) Otherwise returns False

Table 6 shows a description of the move subroutine previously described in more detail in Table 3, on page 25.

Table 6

Subroutine	Purpose
move(row, column)	Moves the tile in position (row, column) to the blank space, if the blank space is next to that tile.
	If the position (row, column) is not next to the blank space, no move will be made.

Write a Python program to help the user solve the puzzle.

The program should:

- get the user to enter the row number of a tile to move
- get the user to enter the column number of a tile to move
- check if the tile in the position entered is next to the blank space
 - o if it is, move that tile to the position of the blank space
 - o if it is not, output Invalid move
- repeat these steps until the puzzle is solved.

You must use the subroutines in Table 5 and Table 6.

You should use indentation as appropriate, meaningful variable name(s) and Python syntax in your answer.

The answer grid opposite contains vertical lines to help you indent your code accurately.

[6 marks]



17



1 3	Explain how the linear search algorithm works.	[3 marks]	Do not write outside the box



1 4.1	State one property of local variables that is not true for all variables. [1 mark]	Do not write outside the box
1 4.2	Using Python, write a subroutine to help a museum review the number of visitors in a month.	
	The subroutine must: • have the identifier countDays • have the number of days a museum was open in the last month as a parameter • get the user to enter the number of visitors to the museum for each of those days • count how many of those days the museum had more than 200 visitors • return the count.	
	You should use indentation as appropriate, meaningful variable name(s) and Python syntax in your answer. The answer grid below contains vertical lines to help you indent your code. [6 marks]	





10

outside the

hox

1 5

A programmer is writing a game.

The game uses a row of cells represented as a list. Figure 20 shows an example.

Figure 20

0	1	2	3	4	5	6	7
			Х			Х	

Figure 21 describes how the game is to be played.

Figure 21

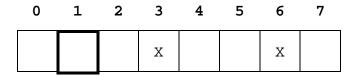
- The player starts at position 0 in a row of cells.
- The aim of the game is for the player to reach the end of the row.
- At each turn the player must enter either 1 or 2
 - o if the player enters 1, the player's position increases by 1
 - o if the player enters 2, the player's position increases by 2
- If the player's position goes beyond the end of the row or contains an X:
 - o the message Bad move is displayed
 - o the player goes back to position 0
- These steps are repeated until the player reaches the end of the row.
- If the player reaches the end of the row the game is finished.

For example, using the list in **Figure 20**:

• the player starts in position 0

0	1	2	3	4	5	6	7
			Х			Х	

• if the player enters a 1, then they move to position 1



Question 15 continues on the next page



Do not write
outside the
box

• if the player then enters a 2, Bad Move is displayed as position 3 contains an X

0	1	2	3	4	5	6	7
			X			Х	

Bad move

• the player then goes back to position 0

0	1	2	3	4	5	6	7
			Х			Х	

• if the player then enters a 2, they move to position 2

0	1	2	3	4	5	6	7
			Х			Х	

• if the player then enters a 2, they move to position 4

0	2				
		X		Х	

• if the player then enters a 1, they move to position 5

0	1	2	3	4	5	6	7
			Х			Х	

• if the player then enters a 2, the game finishes.

0	1	2	3	4	5	6	7
			Х			X	



Figure 22 shows part of a Python program that will be used for the game.

Do not write outside the

Figure 22

```
pos = 0
lastPos = len(row) - 1
while pos < lastPos:</pre>
```

pos is a variable that contains the player's current position.

Extend the program from **Figure 22** so that the game works as described in **Figure 21**, on page 37.

When writing your program you should assume:

- there is a list called row
- the number of X characters in row can vary
- the position of the X characters in row can vary
- the X characters have already been added to the list called row
- the row list can be of any length.

You **should** use indentation as appropriate, meaningful variable name(s) and Python syntax in your answer.

The answer grid below contains vertical lines to help you indent your code.

[8 marks]

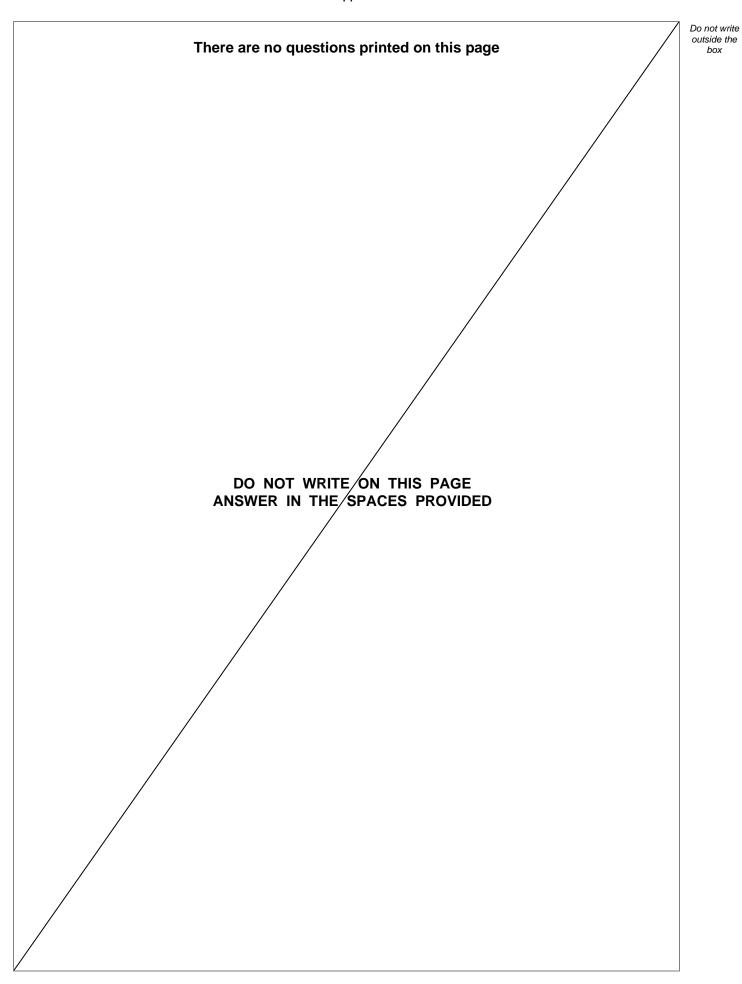
<pre>pos = 0 lastPos = len(row) - 1 while pos < lastPos:</pre>						
	<u>-</u>					



END OF QUESTIONS



8





Question number	Additional page, if required. Write the question numbers in the left-hand margin.



Question number	Additional page, if required. Write the question numbers in the left-hand margin.



Question number	Additional page, if required. Write the question numbers in the left-hand margin.
	3
	Copyright information
	For confidentiality purposes, all acknowledgements of third-party copyright material are published in a separate booklet. This booklet is published after each live examination series and is available for free download from www.aqa.org.uk.
	Permission to reproduce all copyright material has been applied for. In some cases, efforts to contact copyright-holders may have been unsuccessful and AQA will be happy to rectify any omissions of acknowledgements. If you have any queries please contact the Copyright Team.
	Copyright © 2024 AQA and its licensors. All rights reserved.



