AQA

A-level
COMPUTER SCIENCE

Paper 1

Monday 12 June 2023 Afternoon Time allowed: 2 hours 30 minutes

Materials

For this paper you must have:
- a computer
- a printer
- appropriate software
- the Electronic Answer Document
- an electronic version and a hard copy of the Skeleton Program
- an electronic version and a hard copy of the Preliminary Material.

You must **not** use a calculator.

Instructions
- Type the information required on the front of your Electronic Answer Document.
- Before the start of the examination make sure your **Centre Number**, **Candidate Name** and **Candidate Number** are shown clearly **in the footer** of every page (also at the top of the front cover) of your Electronic Answer Document.
- Enter your answers into the Electronic Answer Document.
- Answer **all** questions.
- Save your work at regular intervals.

Information
- The marks for questions are shown in brackets.
- The maximum mark for this paper is 100.
- No extra time is allowed for printing and collating.
- The question paper is divided into **four** sections.

Advice

You are advised to allocate time to each section as follows:
Section A – 40 minutes; **Section B** – 20 minutes; **Section C** – 20 minutes; **Section D** – 70 minutes.

At the end of the examination

Tie together all your printed Electronic Answer Document pages and hand them to the Invigilator.

Warning

It may not be possible to issue a result for this paper if your details are not on every page of your Electronic Answer Document.

Section A

You are advised to spend no longer than **40 minutes** on this section.

Enter your answers for **Section A** in your Electronic Answer Document.

You **must save** this document at regular intervals.

0 1 Describe the process that should be followed to add an item to a circular queue implemented as a static data structure using an array.

Your method should deal appropriately with any issues which could arise.

[5 marks]

0 2 **Figure 1** shows a logic puzzle.

Figure 1

The following five coloured shapes are placed on a table.

Image of coloured shapes not reproduced here due to third party copyright restrictions.

Tabitha secretly chooses one of the coloured shapes and:

- tells Walter the colour of the shape she has chosen (pink, yellow or blue)
- tells Lionel the type of shape she has chosen (triangle, circle or square).

Lionel and Walter both know what coloured shapes are on the table.

Lionel knows that Walter has been told the colour chosen by Tabitha.
Walter knows that Lionel has been told the type of shape chosen by Tabitha.
They do not know what the other has been told.

Tabitha first asks Walter and Lionel if they know which coloured shape she has chosen. They both answer at the same time and say “No”.

Tabitha then asks them again if they know which coloured shape she has chosen. They both answer at the same time and say “No” again.

Tabitha asks them a third time if they know which coloured shape she has chosen and they both answer at the same time and say “Yes”.

0 2 . 1 After they have both replied to Tabitha’s **first** question, what does Lionel now know about Tabitha’s choice because Walter said “No”?

[1 mark]

0 2 . 2 After they have both replied to Tabitha's **first** question, what does Walter now know about Tabitha's choice because Lionel said "No"?

[1 mark]

0 2 . 3 Which coloured shape had Tabitha chosen?

[1 mark]

Turn over for the next question

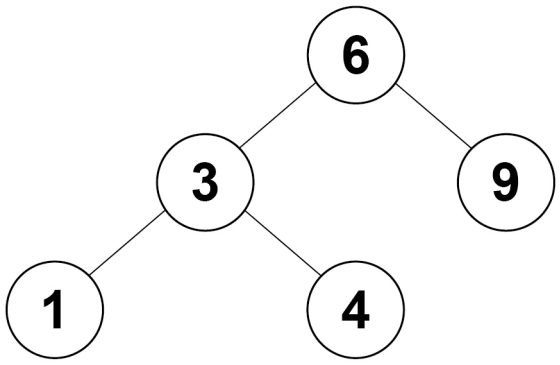
0 3 A binary tree is a type of data structure.

0 3 . 1 State **two** characteristics that make a tree a binary tree.

[2 marks]

0 3 . 2 **Figure 2** shows a binary tree and its representation using an array of records called `Tree`. Each record consists of three fields, `Data`, `Left` and `Right`.

Figure 2



	Data	Left	Right
[0]	6	1	4
[1]	3	2	3
[2]	1	-1	-1
[3]	4	-1	-1
[4]	9	-1	-1

Figure 3 shows a subroutine that implements a binary tree search algorithm using the array `Tree`. The subroutine parameter, `k`, is the data item being searched for. The subroutine returns a Boolean value indicating if the data item being searched for is in the binary tree or not.

Parts of the algorithm are missing.

Figure 3

```
SUBROUTINE BTS(k)
    Current ← ____1____
    WHILE Current > ____2____
        IF Tree[Current].Data = k THEN
            RETURN ____3____
        ELSEIF Tree[Current].Data < k THEN
            ____4____
        ELSE
            ____5____
        ENDIF
    ENDWHILE
    RETURN ____6____
ENDSUBROUTINE
```

Complete each row in **Table 1**, to show what the labels indicating missing parts of the algorithm in **Figure 3** should be replaced by.

Table 1

1	
2	
3	
4	
5	
6	

Copy the contents of the unshaded cells in **Table 1** into the table in your Electronic Answer Document.

[4 marks]

0 3 . 3 There are similarities in how the binary tree search and the binary search algorithms work.

State the big-O time complexity of the **binary search** algorithm.

[1 mark]

0 3 . 4 Explain why the binary search algorithm has the time complexity stated in your answer for **03.3**.

[1 mark]

0 3 . 5 Explain why searching for an item in a list or tree is a tractable problem.

[1 mark]

0 3 . 6 Heuristics can be used when working with an intractable problem.

Explain what heuristics are.

[2 marks]

0 3 . 7 Explain what is meant by constant time complexity.

[2 marks]

0 4 A regular language is one that can be defined using a regular expression.

Figure 4 shows definitions for six different languages.

Figure 4

Language A

```
<string> ::= <term> <op> <term>
<string> ::= ( <string> )
<op>     ::= ÷ | -
<term>   ::= a | b
```

Language B

```
(a|b)+ab*
```

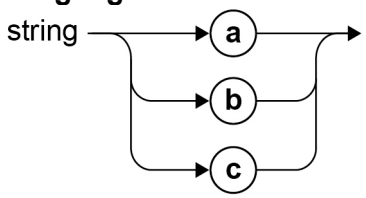
Language C

```
<string> ::= a | b | <num>
<string> ::= <string> a | <string> b
<digit>  ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<num>    ::= <digit> <num> | <digit>
```

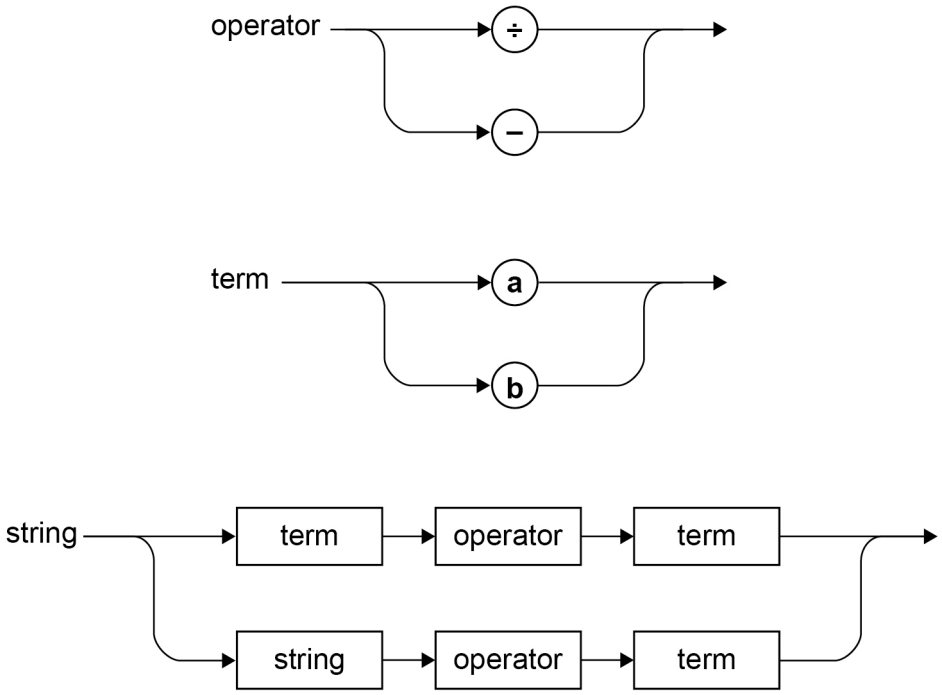
Language D

$\{a^n b \mid n \geq 1\}$

Language E



Language F



0 4 . 1 Complete **Table 2** to show which of the languages in **Figure 4** are regular languages.

Table 2

Language	Regular language (Y/N)?
Language A	
Language B	
Language C	
Language D	
Language E	
Language F	

Copy the contents of the unshaded cells in **Table 2** into the table in your Electronic Answer Document.

[3 marks]

0 4 . 2 Show that the language defined by the union of the sets $\{b^n \mid n > 0\}$ and $\{a, ab\}$ is a regular language by writing a regular expression for the language.

[2 marks]

0 4 . 3 Explain what is meant by the cardinality of a set.

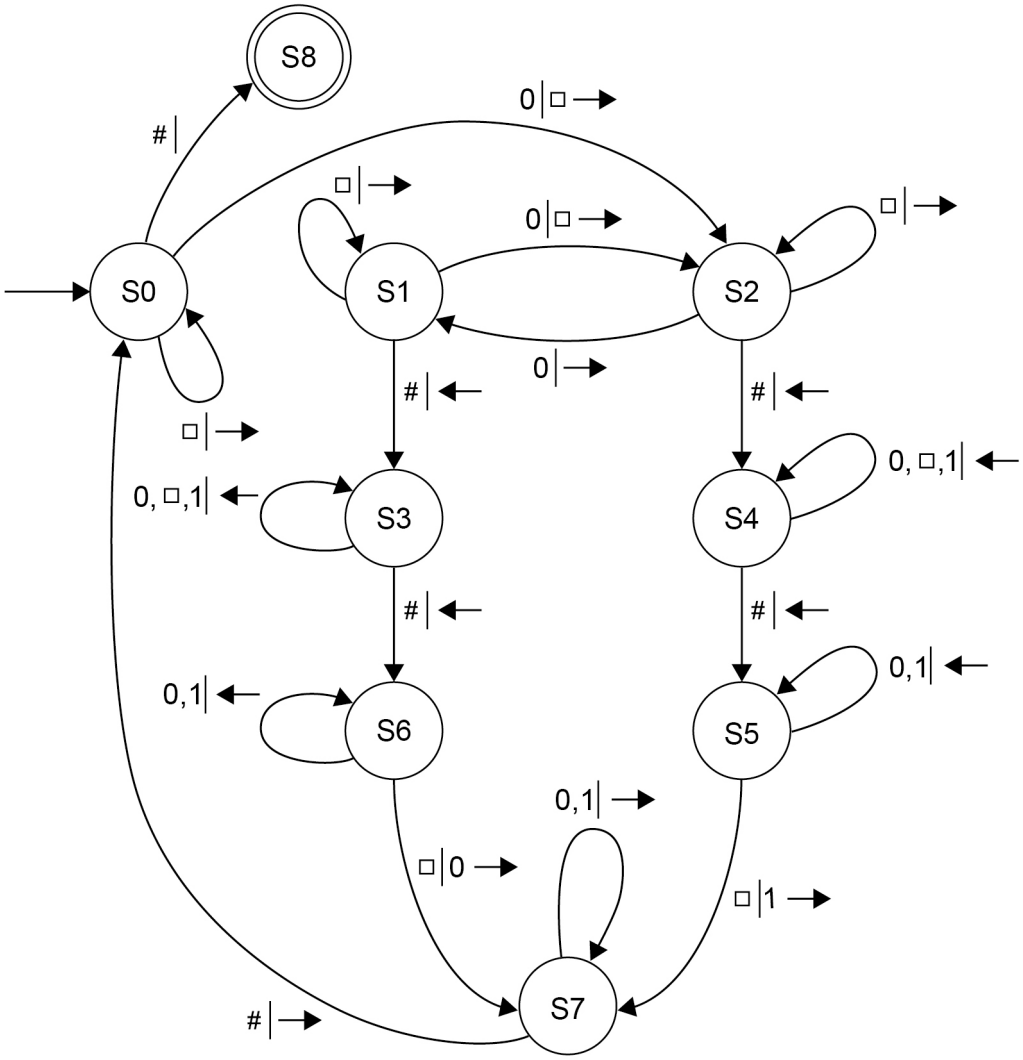
[1 mark]

Turn over for the next question

0 5 **Figure 5** shows the transition function (represented as a state transition diagram) and part of the tape for a Turing machine designed to count the number of zeros between two # symbols on the tape.

The current position of the read/write head is indicated by the * symbol.

Figure 5



…	#	0*	0	0	#	…

The □ symbol is used to denote an empty cell on the tape.

The label □|0 → on the transition from S6 to S7 means if the machine is in state S6 and a □ is read from the tape, then a 0 should be written to the tape and then the read/write head is moved one cell to the right.

The label 0, □,1| ← on the transition from S3 to S3 means if the machine is in state S3 and a □, a 0 or a 1 is read from the tape, then the read/write head is moved one cell to the left; the contents of the tape are not changed.

0 5 . 1 Complete the unshaded cells of **Table 3** to show the result of tracing the computation of this Turing machine **for the first 20 steps only**.

Show the contents of the tape, the current position of the read/write head and the current state as the input is processed. The top row of the table shows the initial state, the initial position of the read/write head and the starting contents of the tape.

Table 3

Tape										Current state
…			#	0*	0	0	#		…	S0
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	
…									…	

Copy the contents of the unshaded cells in **Table 3** into the table in your Electronic Answer Document.

[5 marks]

0 5 . 2 Describe **two** circumstances when the transition function for this Turing machine will **not** correctly count the number of zeros between two # symbols on the tape.

You should assume that:

- the read/write head starts immediately to the right of a cell containing a #
- there is a cell containing a # symbol somewhere on the tape to the right of the read/write head's initial position
- there are exactly two # symbols on the tape.

[2 marks]

Turn over ►

0 5 . 3 Explain what a Universal Turing machine is.

[2 marks]

0 5 . 4 State **one** reason why there are some problems that no real computer can solve that the Universal Turing Machine could solve.

[1 mark]

Section B

You are advised to spend no more than **20 minutes** on this section.

Enter your answers to **Section B** in your Electronic Answer Document.

You **must save** this document at regular intervals.

The question in this section asks you to write program code **starting from a new program/project/file**.

You are advised to **save** your program at regular intervals.

0 6 Write a program that gets the user to enter a string. It should keep getting the user to enter a string until they enter a valid string. Each time they enter a string an appropriate message should be displayed telling them whether the string they entered is valid or not.

For a string to be valid:

- it must be between five and seven characters in length (inclusive)
- it must consist only of uppercase letters
- it must contain only unique characters (ie no character appears in the string more than once)
- the sum of the ASCII codes of the characters in the string must be between 420 and 600 (inclusive).

Evidence that you need to provide
Include the following evidence in your Electronic Answer Document.

0 6 . 1 Your PROGRAM SOURCE CODE

[12 marks]

0 6 . 2 SCREEN CAPTURE(S) showing the results of testing the program by entering the strings `BOIL`, `BRAisE`, `ROAST`, `BLANCH` and `PRESSURECOOK`. You will need to execute your program more than once to test all of the strings.

[1 mark]

Section C

You are advised to spend no more than **20 minutes** on this section.

Enter your answers to **Section C** in your Electronic Answer Document.

You **must save** this document at regular intervals.

These questions refer to the **Preliminary Material** and the **Skeleton Program**, but **do not** require any additional programming.

Refer **either** to the **Preliminary Material** issued with this question paper **or** your electronic copy.

0 7 This question is about the `PlayGame` method in the `Dastan` class.

0 7 . 1 Complete the unshaded cells of **Table 4** to show which statements are true and which statements are false about the `PlayGame` method.

Table 4

Statement	True/False
Uses definite iteration	
Uses nested iteration	
Uses nested selection	
Uses one or more global variables	
Uses one or more local variables	
Uses one or more named constants	

Copy the contents of the unshaded cells in **Table 4** into the table in your Electronic Answer Document.

[2 marks]

0 7 . 2 If the `PlayGame` method was altered to allow the user to select any move option from their queue, what is the maximum amount their score could decrease by after making a move?

[1 mark]

0 8 The **Skeleton Program** uses overriding. When a derived class overrides a method from the base class, the base class method could be either a virtual method or an abstract method.

Describe **two** differences between a virtual method and an abstract method.

[2 marks]

0 9 This question is about the `MoveOptionQueue` class.

0 9 . 1 Explain how this class uses information hiding.

[1 mark]

The data structure `Queue` has **not** been implemented as a first-in first-out (FIFO) queue.

0 9 . 2 Explain why the `MoveItemToBack` method in the `MoveOptionQueue` class does **not** use the `Queue` data structure as a FIFO queue.

[1 mark]

0 9 . 3 State the identifier of a method in the `MoveOptionQueue` class that **does** use the `Queue` data structure as a FIFO queue.

[1 mark]

1 0 This question is about the `CheckIfGameOver` method in the `Dastan` class.

1 0 . 1 Explain why the first selection structure is needed.

[1 mark]

1 0 . 2 **Figure 6** shows a pseudo-code representation of part of the `CheckIfGameOver` method.

Figure 6

```
NOT (Player1HasMirza AND Player2HasMirza)
```

Rewrite the code in **Figure 6** so that it has the same functionality but uses `OR` and `NOT` Boolean operations only, instead of the `AND` and `NOT` Boolean operations.

[1 mark]

1 1

1 1 . 1 State the name of an identifier for a user-defined method that uses string concatenation.

[1 mark]

1 1 . 2 State the name of an identifier for a class that has a constructor that takes three integer parameters.

[1 mark]

1 2 Explain why a value of `-1` is used in the `CreateMoveOptions` method in the `Dastan` class.

[1 mark]

Turn over ►

Section D

You are advised to spend no more than **70 minutes** on this section.

Enter your answers to **Section D** in your Electronic Answer Document.

You **must save** this document at regular intervals.

These questions require you to load the **Skeleton Program** and to make programming changes to it.

1 3 This question refers to the method `UseMoveOptionOffer` in the `Dastan` class.

The program is to be changed so that it checks the choice made by the player about which move option to replace is valid. A choice is valid if it is between 1 and 5 inclusive.

The program should keep getting the player to enter a value until a valid choice has been made. Each time an invalid value is entered an appropriate error message should be displayed.

What you need to do

Task 1
Modify the method `UseMoveOptionOffer` so it checks that the value entered by the player is valid. An appropriate error message should be displayed if an invalid value is entered and the user should be made to enter another value.

You may assume that the user will only enter a single numeric digit.

Task 2
Test that the changes you have made work:

- run the Skeleton Program
- enter `9`
- enter `6`
- enter `4`

Evidence that you need to provide
Include the following evidence in your Electronic Answer Document.

1 3 . 1 Your PROGRAM SOURCE CODE for the amended method `UseMoveOptionOffer`.

[4 marks]

1 3 . 2 SCREEN CAPTURE(S) showing the results of the requested test.

[1 mark]

1 4 This question extends the Skeleton Program by allowing a player to choose a bhukampa (earthquake).

After choosing a bhukampa the player can then choose a move option, choose to use the move option offer or choose a bhukampa again.

When a bhukampa is chosen the following steps are repeated **five** times:

- two **different** squares are randomly selected
- the positions on the board of those two squares are swapped.

The player's score is then reduced by 15 and the new state of the game is displayed.

What you need to do

Task 1
Create a new method in the `Dastan` class called `ProcessBhukampa` that swaps the squares in the way described.

Task 2
Modify the `PlayGame` method in the `Dastan` class so that if the player enters `8` when asked for their move option, the new method `ProcessBhukampa` is called, the current player's score is reduced by 15 and the new game state is displayed.

Task 3
Test that the changes you have made work:

- run the Skeleton Program
- enter `8`

The results of your test should show at least one piece or kotla now being in a different location. If your test does not show this (because all the squares randomly selected by your program were empty), then keep doing this test until that is not the case.

Evidence that you need to provide
Include the following evidence in your Electronic Answer Document.

1 4 . 1 Your PROGRAM SOURCE CODE for the new method `ProcessBhukampa` and the amended method `PlayGame`.

[9 marks]

1 4 . 2 SCREEN CAPTURE(S) showing the requested test.

[1 mark]

1 5 This question extends the Skeleton Program so that there is a new type of square – a gacaka (trap).

When a player makes a move that means a piece moves into a gacaka, the value of `PointsIfCaptured` of the piece is increased by 2 and a message saying "`Trap!`" is displayed. A player who has a piece in a gacaka has their score decreased by 3 each turn until the piece is moved out of the gacaka.

The location of the gacaka is not shown on the board.

What you need to do

Task 1
Create a new class called `Gacaka` that is a subclass of the `Square` class.

Create two methods `SetPiece` and `GetPointsForOccupancy` in the `Gacaka` class that allow a gacaka to work in the way described.

The method `SetPiece` in the `Gacaka` class should override the method from the `Square` class.

The method `GetPointsForOccupancy` in the `Gacaka` class should override the method from the `Square` class.

Task 2
Modify the `CreateBoard` method in the `Dastan` class so that row four, column four is a gacaka instead of a square.

Task 3
Test that the changes you have made work:

- run the Skeleton Program
- enter `3`
- enter `24`
- enter `44`
- enter `1`
- enter `54`
- enter `44`

Evidence that you need to provide
Include the following evidence in your Electronic Answer Document.

1 5 . 1 Your PROGRAM SOURCE CODE for the new class `Gacaka`, the amended `CreateBoard` method and any other methods you have modified or created when answering this question.

[8 marks]

1 5 . 2 SCREEN CAPTURE(S) showing the requested test.

[1 mark]

1 6 This question extends the program by telling the player how many legal moves they have in the current state of the game.

A legal move is when, using one of the first three move options in their move option queue, a player's piece will move to a square that contains an opponent's piece or to a square that does not contain a piece. There are 45 legal moves at the start of the game for the first player.

When answering this question, you should ignore the option to use the move option offer and the option to use a bhukampa (from Question **14**) as these do not count as moves.

What you need to do

Task 1
Create a new method called `GetNoOfPossibleMoves` in the `Player` class that takes `Board` as a parameter and returns the total number of possible legal moves for that player based on the contents of the squares in the `Board` list.

Task 2
Modify the `DisplayState` method in the `Dastan` class so that it displays the value returned by a call to `GetNoOfPossibleMoves` for the current player.

Task 3
Test that the changes you have made work:

- run the Skeleton Program
- enter `1`
- enter `22`
- enter `12`

Evidence that you need to provide
Include the following evidence in your Electronic Answer Document.

1 6 . 1 Your PROGRAM SOURCE CODE for the new method `GetNoOfPossibleMoves` and the amended method `DisplayState`.

[12 marks]

1 6 . 2 SCREEN CAPTURE(S) showing the requested test with the number of legal moves for the second player displayed.

[1 mark]

END OF QUESTIONS

There are no questions printed on this page