* 0000800000001 *

Cambridge Assessment International Education

# Cambridge International AS & A Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

* 9 4 4 9 6 0 0 8 7 0 *

**COMPUTER SCIENCE** **9618/33**

Paper 3 Advanced Theory **May/June 2025**

**1 hour 30 minutes**

You must answer on the question paper.

No additional materials are needed.

**INSTRUCTIONS**

- Answer **all** questions.
- Use a black or dark blue pen.
- Write your name, centre number and candidate number in the boxes at the top of the page.
- Write your answer to each question in the space provided.
- Do **not** use an erasable pen or correction fluid.
- Do **not** write on any bar codes.
- You may use an HB pencil for any diagrams, graphs or rough working.
- Calculators must **not** be used in this paper.

**INFORMATION**

- The total mark for this paper is 75.
- The number of marks for each question or part question is shown in brackets [ ].
- No marks will be awarded for using brand names of software packages or hardware.

This document has **16** pages. Any blank pages are indicated.


© UCLES 2025

[Turn over

**1** A programmer is writing a program to manage a video library. They require a user-defined data type.

**(a)** Write **pseudocode** statements to declare the composite data type `VideoLibrary` to hold data about each video in the collection. This data includes:

- identity code (any combination of letters and numbers)
- title
- year released
- date purchased
- format (for example DVD, Blu-ray, 4K, MP4)
- running time (minutes)

Use the most appropriate data type in each case.

.....

.....

.....

.....

.....

.....

.....

..... [4]

**(b)** Identify **one** field in `VideoLibrary` that could be an efficient enumerated data type and give a reason for your choice.

Field .....

.....

Reason .....

.....

[2]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**2** Numbers are stored in a computer using binary floating-point representation with:

- 8 bits for the mantissa
- 8 bits for the exponent
- two's complement form for both the mantissa and the exponent.

**(a)** Give the largest normalised positive two's complement binary number that can be stored in this system **and** state its denary equivalent.

The denary answer should be expressed in terms of powers of 2.

**Mantissa** **Exponent**

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | |

Denary .....

[2]

**(b)** Calculate the normalised binary floating-point representation of –3.59375 in this system. Show your working.

**Mantissa** **Exponent**

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | |

Working .....

.....

.....

.....

.....

.....

.....

.....

[4]

DO NOT WRITE IN THIS MARGIN

**3** This truth table represents a logic circuit.

| INPUT | | | | OUTPUT |
|---|---|---|---|---|
| **A** | **B** | **C** | **D** | **Z** |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

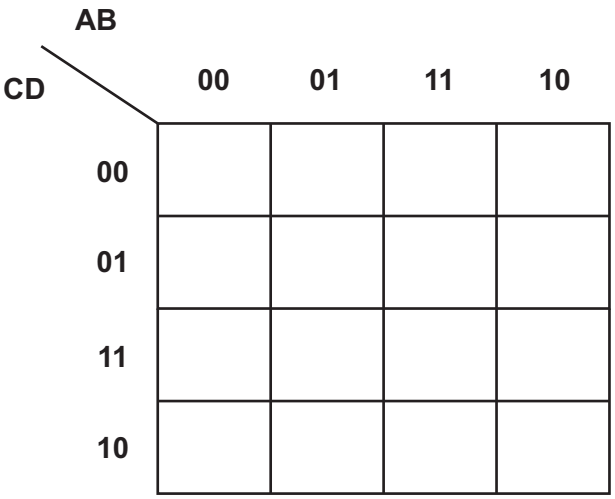
**(a)** Write the Boolean logic expression that corresponds to the given truth table as the sum-of-products.

**Z =** .....

..... [2]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**(b) (i)** Complete the Karnaugh map (K-map) for the given truth table.




[2]

**(ii)** Draw loop(s) around appropriate group(s) in the K-map to produce an optimal sum-of-products. [2]

**(iii)** Write the Boolean logic expression from your answer to part **(b)(ii)** as the simplified sum-of-products.

**Z =** .....

..... [2]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**4** **(a)** The Internet layer and Link layer are two layers of the TCP/IP protocol suite.

Describe the purpose of the Internet layer **and** the purpose of the Link layer.

Purpose of Internet layer .....

.....

.....

.....

.....

Purpose of Link layer .....

.....

.....

.....

.....

[5]

**(b)** Describe the function of a router in packet switching.

.....

.....

.....

.....

.....

.....

.....

..... [4]

DO NOT WRITE IN THIS MARGIN

DO NOT WRITE IN THIS MARGIN

DO NOT WRITE IN THIS MARGIN

DO NOT WRITE IN THIS MARGIN

DO NOT WRITE IN THIS MARGIN

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**5** Complete the table by filling in the missing object-oriented programming (OOP) terms and descriptions.

| OOP term | Description |
|---|---|
| ..... | A method that accesses the value of a property. |
| ..... | A method that changes the value of a property. |
| Object | ..... ..... ..... |
| Method | ..... ..... ..... |

[4]

**6** The management and scheduling of processes are tasks carried out by an operating system.

**(a)** Describe **one** reason why scheduling is necessary in process management.

.....

.....

.....

..... [2]

**(b)** Explain the function of the round robin scheduling routine **and** give a benefit of this routine.

Function .....

.....

.....

.....

.....

.....

Benefit .....

.....

[4]

**7** Secure Socket Layer (SSL) and Transport Layer Security (TLS) are two protocols.

Explain how SSL/TLS is used when client-server communication is initiated.

.....

.....

.....

.....

.....

.....

.....

..... [4]

**8** Explain the process of **syntax analysis** during program compilation.

.....

.....

.....

.....

.....

.....

.....

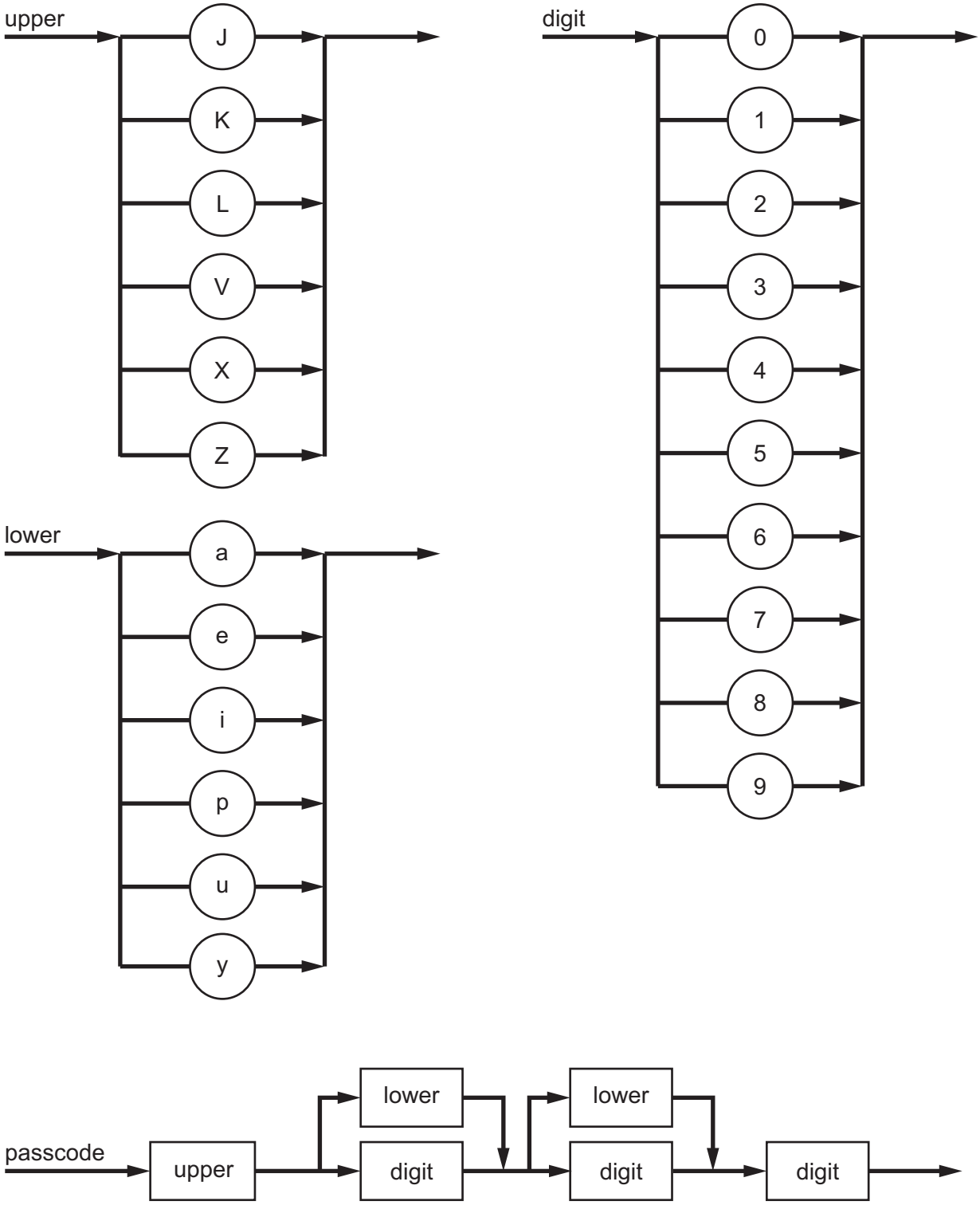
..... [4]

DO NOT WRITE IN THIS MARGIN

**BLANK PAGE**

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**9** Several syntax diagrams are shown.




**(a)** State why `JJ90` is **not** a valid passcode for the given syntax diagrams.

.....

..... [1]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**(b)** Complete the Backus-Naur Form (BNF) for `<upper>` and `<passcode>`.

`<upper> ::=` .....

.....

`<passcode> ::=` .....

.....

.....

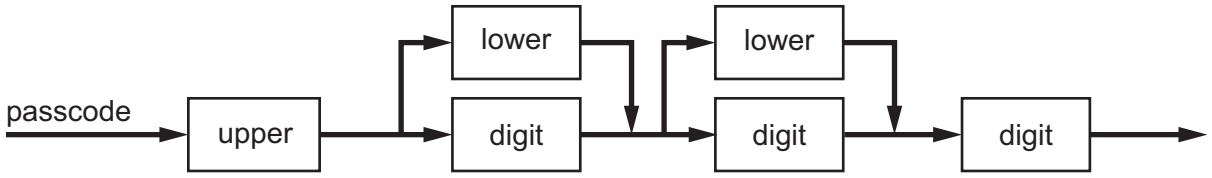
.....

[3]

**(c)** A character can be an upper, a lower or a digit.

The rules for passcode have been changed so that the third character may also be selected from upper and the final character may be repeated one or more times.

Complete the syntax diagram for passcode to show these changes.




[2]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**10** **(a)** State the purpose of the A* and Dijkstra's algorithms.

.....

..... [1]

**(b)** Outline the difference between the A* and Dijkstra's algorithms.

.....

.....

.....

..... [2]

**(c)** Explain how unsupervised learning takes place in machine learning.

.....

.....

.....

.....

.....

..... [3]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**11** The pseudocode algorithm below allows a user to input a new stock item. The random file is searched for the next empty location in the file and the new item is inserted there.
A suitable message is displayed if the file is full.

Complete this pseudocode.

```
DECLARE Location : INTEGER
DECLARE NewStock : STRING
DECLARE CurrentStock : STRING
DECLARE Stored : BOOLEAN
DECLARE Max : INTEGER
Max ← 100000
Stored ← FALSE
Location ← 1

.....
OUTPUT "Enter the new item you wish to store: "
INPUT NewStock
WHILE NOT Stored AND Location <= Max

   .....

   GETRECORD "StockList.dat", .....
   IF CurrentStock = "" THEN

      ..... "StockList.dat", NewStock
      Stored ← TRUE
   ELSE
      Location ← Location + 1
   ENDIF
ENDWHILE

..... THEN
   OUTPUT "The new item has not been stored as the file was full."
ENDIF
CLOSEFILE "StockList.dat"
```

[5]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**12 (a)** An array is an Abstract Data Type (ADT).

Identify **two** other ADTs.

1 .....

2 .....

[1]

**(b)** A 1D array `DataArray` holds up to 1000 elements of type integer and needs to be sorted in ascending order.

Write the **pseudocode** for an insertion sort to sort the array into ascending order.

Use the identifiers from the table in your algorithm.

You do **not** need to declare any arrays or variables for this algorithm. You may assume this has already been done.

| Identifier | Data type | Description |
|---|---|---|
| `Index` | `INTEGER` | counter for outer loop |
| `Position` | `INTEGER` | counter for inner loop – insertion position |
| `DataArray` | `INTEGER` | 1D array to store up to 1000 integers |
| `Value` | `INTEGER` | value to insert |

The first line has been written for you.

```
FOR Index ← 2 to 1000
```

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

DO NOT WRITE IN THIS MARGIN

**(c)** Describe **two** ways in which the performance of a sort routine is affected by the data to be sorted.

1 .....

.....

2 .....

.....

[2]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

**13** The recursive procedure `Delete()` is defined as follows:

```
PROCEDURE Delete(Index, Target)
   IF Numbers[Index] > 0 THEN
      IF Numbers[Index] >= Target THEN
         Numbers[Index] ⟵  Numbers[Index + 1]
      ENDIF
      Index ⟵ Index + 1
      CALL Delete(Index, Target)
   ENDIF
ENDPROCEDURE
```

An array `Numbers` is used to store a sorted data set of non-zero positive integers. Unused cells contain zero.

The contents of the array at the start of the algorithm are:

| **Numbers** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **[1]** | **[2]** | **[3]** | **[4]** | **[5]** | **[6]** | **[7]** | **[8]** | **[9]** | **[10]** |
| 2 | 3 | 7 | 11 | 15 | 17 | 19 | 23 | 0 | 0 |

Complete the trace table for the algorithm for the procedure call:

```
CALL Delete(1, 15)
```

| | | **Numbers** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Index** | **Target** | **[1]** | **[2]** | **[3]** | **[4]** | **[5]** | **[6]** | **[7]** | **[8]** | **[9]** | **[10]** |
| | | 2 | 3 | 7 | 11 | 15 | 17 | 19 | 23 | 0 | 0 |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |

[4]

Permission to reproduce items where third-party owned material protected by copyright is included has been sought and cleared where possible. Every reasonable effort has been made by the publisher (UCLES) to trace copyright holders, but if any items requiring clearance have unwittingly been included, the publisher will be pleased to make amends at the earliest possible opportunity.

To avoid the issue of disclosure of answer-related information to candidates, all copyright acknowledgements are reproduced online in the Cambridge Assessment International Education Copyright Acknowledgements Booklet. This is produced for each series of examinations and is freely available to download at www.cambridgeinternational.org after the live examination series.

Cambridge Assessment International Education is part of Cambridge Assessment. Cambridge Assessment is the brand name of the University of Cambridge Local Examinations Syndicate (UCLES), which is a department of the University of Cambridge.



DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN