



RAFFLES INSTITUTION

2024 YEAR 6 PRELIMINARY EXAM

CANDIDATE
NAME

CLASS

24

COMPUTING

9569/01

Paper 1 Written

September 2024

3 hours

READ THESE INSTRUCTIONS FIRST

An answer booklet will be provided with the question paper. You should follow the instructions on the front cover of the answer booklet. If you need additional answer paper ask the invigilator for a continuation booklet.

Answer **all** questions.

Approved calculators are allowed.

The number of marks is given in brackets [] at the end of each question or part question.

The total number of marks for this paper is 100.

This document consists of **8** printed pages.

RAFFLES INSTITUTION
Mathematics Department

- 1 An insurance company offers various plans, such as car, health, and home insurance. Customers can purchase one or more policies, with the assistance of agents who help them select and manage these policies. Each policy is associated with a specific agent, who is responsible for guiding the customer and handling any issues related to the policy. When an insured event occurs, customers can file claims to receive payouts.

Design a relational database to manage the following four entities:

- Policy: Each policy has a unique policy number, type (e.g., car, health, home), coverage amount, premium (cost), start date, and end date.
- Customer: Each customer has personal details like name, contact number, email address and home address.
- Claim: Each claim includes the claim date, amount, and status (e.g., approved, pending), and is linked to the corresponding policy.
- Agent: Agents assist customers with selecting and managing policies and claims

- (a) Draw an entity-relationship (ER) diagram showing the four entities and the relationships between them. [3]

- (b) A table description can be expressed as:

`TableName (Attribute1, Attribute2, Attribute3, ...)`

The primary key is indicated by underlining one or more attributes. Foreign keys are indicated by using a dashed underline.

Write table descriptions for the four entities so that they are in third normal form. [6]

- (c) (i) The company wants to find out which types of insurance policies are associated with the highest total claims. Write an SQL query to find the total amount of claims paid out for each policy type and sort them by total claim amount in descending order. [4]
- (ii) The company needs to update the premium amount for a specific policy. Write an SQL query to update the premium amount to '1500' for the policy with policy number 'P12345'. [2]
- (d) The insurance company is expanding its operations to include various types of information, such as customer reviews, social media interactions, and real-time claims data. It is considering using a NoSQL Database Management System (DBMS) instead of a relational database. State **three** reasons why the company may want to switch to a NoSQL DBMS. [3]
- (e) To prevent fraud, the insurance company plans to collaborate with a third-party vendor for fraud detection services. What actions must the company take before sharing customer data with the vendor? [2]

- 2 A local community centre organises various types of events. These events are categorised into three categories:

- Workshops: Interactive sessions where participants learn new skills.
- Seminars: Informative sessions led by experts on specific topics.
- Fundraisers: Events aimed to raise funds for a cause.

Each type of event shares some common attributes but also has specific features that distinguish them. The common attributes include:

- event ID,
- event name,
- date,
- duration,
- organiser.

The common methods for all events are:

- `scheduleEvent()`,
- `registerParticipant()`,
- `calculateCost()`.

The additional attributes for each type of event are:

- Workshop: materials provided, maximum number of participants.
- Seminar: speaker name, handouts provided.
- Fundraiser: target amount, donation received.

- (a) Draw a class diagram for the situation described, showing
- the superclass
 - any subclasses
 - inheritance
 - properties
 - appropriate methods
 - polymorphism
- [8]
- (b) Explain polymorphism using the example in your class diagram. [3]
- (c) Explain how encapsulation is implemented in the situation described. [2]

- 3 You are designing a music player application that needs to manage playlists efficiently. Each playlist consists of a collection of songs, and the application allows users to frequently perform operations such as adding new songs, removing existing songs, and accessing songs at specific positions in the playlist.

Consider these functions for the adding of new songs to the playlists, one implemented using an array, and the other implemented using a linked list.

```

FUNCTION AddSongToArray(playlistArray: ARRAY, newSong: STRING,
currentSize: INTEGER, maxSize:INTEGER)
  IF currentSize >= maxSize THEN
    // Resize array if necessary
    newMaxSize ← maxSize * 2
    newArray ← New Array of size newMaxSize
    FOR i = 0 to currentSize - 1
      newArray[i] ← playlistArray[i]
    ENDFOR
    playlistArray ← newArray
    maxSize ← newMaxSize
  ENDIF
  // Add new song
  playlistArray[currentSize] ← newSong
  currentSize ← currentSize + 1
  RETURN playlistArray, currentSize, maxSize
ENDFUNCTION

```

```

FUNCTION AddSongToLinkedList(head: POINTER, newSong: STRING)
  newNode ← New Node
  newNode.song ← newSong
  newNode.next ← NULL

  IF head = NULL THEN
    head ← newNode
  ELSE
    current ← head
    head ← newNode
    newNode.next ← current
  ENDIF

  RETURN head
ENDFUNCTION

```

- (a) Explain, in terms of time complexity, the performance of adding a new song to the playlist using an array and a linked list for this situation. [3]
- (b) Using pseudocode, write a function `RemoveSongFromArray` that will delete a song from the array. The algorithm should locate the song to delete in the array by iterating through the playlist. If the song is found, it should maintain the order of the playlist by shifting all the elements after the deleted song by one position to fill the gap.

The function specification is:

```

FUNCTION      RemoveSongFromArray      (playlistArray:      ARRAY,
songToDelete:  STRING,  currentSize:  INTEGER)  RETURNS  TUPLE
(playlistArray, currentSize) [5]

```

- (c) Using pseudocode, write a function `RemoveSongFromLinkedList` that will delete a song from the linked list. The algorithm should search for the song to delete by traversing the linked list. If the song is found, delete it from the linked list.

The function specification is:

```
FUNCTION RemoveSongFromLinkedList (head: POINTER, songToDelete:
STRING) RETURNS head [5]
```

- (d) If a hash table were used to manage the songs in each playlist instead of arrays or linked lists, which operation would improve significantly? Explain why. [3]

- 4 A company has branch offices across the country and needs to ensure secure communication and data access for its employees. The company wants a network solution for secure remote access and unified data management.

- (a) Describe how combining an Intranet with a Virtual Private Network (VPN) can provide secure remote access to the company's internal resources. [3]
- (b) Explain why a client-server architecture is effective for managing and storing the company's data and applications. [3]
- (c) How can a firewall protect the company's network from unauthorised access and ensure data integrity? [2]
- (d) State **three** limitations of using a firewall in protecting the company's network. [3]

- 5 Merge sort is an algorithm used to sort data items into ascending or descending order.

- (a) Explain why merge sort is a divide-and-conquer algorithm. [3]
- (b) Explain why merge sort is preferred over insertion sort for handling large datasets. [2]
- (c) Merge sort may be implemented recursively and iteratively. What are the advantages of each implementation? [2]

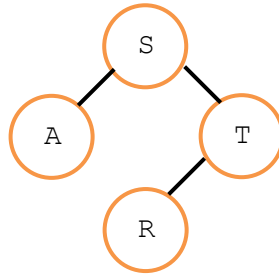
6 The algorithm performs a traversal on a binary tree.

```

PROCEDURE Process(root)
    stack ← CreateStack() // Initialise an empty stack
    node ← root
    WHILE node IS NOT null OR stack IS NOT empty
        WHILE node IS NOT null
            Push node to stack
            node ← node.left
        ENDWHILE
        node ← Pop from stack
        Output(node.value)
        node ← node.right
    ENDWHILE
ENDPROCEDURE

```

- (a) Copy and complete the trace table for the function call `Process(tree.root)`, where `tree.root` refers to the root node of a binary tree.



Step	node	stack	Output
1	S	Empty	
2	A	[S]	
3	null	[S, A]	
.....			

Insert rows to complete the trace table. [4]

- (b) Rewrite the given algorithm using recursion instead of using a stack. [4]
- (c) State **two** differences between a binary tree and a binary search tree. [2]
- (d) What are the consequences of having an unbalanced binary search tree? [2]

7 A function is written to validate an array of codes using the ISBN-13 check digit algorithm.

```

01 FUNCTION ValidateCodes(Codes : ARRAY OF STRING, NumberOfCodes :
    INTEGER) RETURNS INTEGER
02     DECLARE ValidCount : INTEGER
03     DECLARE i, j, TotalSum, Digit, CheckDigit : INTEGER
04     DECLARE Code : STRING
05     ValidCount ← 0
06     FOR i = 0 TO NumberOfCodes
07         Code ← Codes[i]
08         IF LENGTH(Code) <> 13 OR NOT IsNumeric(Code) THEN
09             CONTINUE
10         ENDIF
11         TotalSum ← 0
12         FOR j = 0 TO 11
13             Digit ← INTEGER(Code[j])
14             IF j MOD 2 = 0 THEN
15                 TotalSum ← TotalSum + Digit
16             ELSE
17                 TotalSum ← TotalSum + Digit * 3
18             ENDIF
19         ENDFOR
20         CheckDigit ← INTEGER(Code[12])
21         IF (TotalSum + CheckDigit) MOD 10 = 0 THEN
22             ValidCount ← ValidCount + 1
23         ENDIF
24     ENDFOR
25     RETURN ValidCount
26 ENDFUNCTION

```

(a) Line 6 generates an error. Explain why the error occurs and how you would correct it. [2]

(b) Given the function call with the following arguments:

- Codes = ["1234567890128", "123456789012a", "1234567890", "123456789012", "1234567890124"]
- NumberOfCodes = 5

What is the return value from the function? Explain your answer. [4]

(c) Invalid codes in the Codes array are not processed.

Describe how you can modify the function to report these invalid codes. You are not required to rewrite the entire pseudocode. [3]

- 8 A smart home lighting system is programmed to automatically control the lights based on the following conditions:

- If it is nighttime, the lights should be turned on.
- If the room is occupied and it is daytime, the lights should be turned on.
- If the room is unoccupied, the lights should be turned off regardless of the time of day.
- The user can manually override the system to keep the lights off. The lights should remain off even if the other conditions suggest they should be on.
- Additionally, the lights can be set to a specific colour if it is nighttime, and the room is occupied.

(a) Create a decision table showing all the possible conditions and actions. [4]

(b) Simplify your decision table by removing redundancies. [2]

Another smart home air conditioning system uses various data inputs to control its operations. The data components are:

- Cooling status: 1 bit

0	Off
1	On

- Mode setting: 2 bits

00	Cooling
01	Dehumidifying
10	Fan
11	Auto

- Temperature setting: 4 bits (to represent a range of temperature settings)

The binary format is 7 bits (1+2+4) in total. When using a byte, it is organised as follows:

Bit position	7	6	5	4	3	2	1	0
Purpose	Unused bit	Cooling status	Mode setting		Temperature setting			

For example, the binary data 0100 1000 represents the following operations:

Bit value	0	1	0	0	1	0	0	0
Purpose	Unused bit	Cooling status = 'On'	Mode setting = 'Cooling' (binary 00)		Temperature setting = 8 (binary 1000)			

(c) How many different temperature settings are possible? [1]

(d) Convert the binary data '0100 1000' to its hexadecimal representation. [1]

(e) If the hexadecimal data is '6E', convert it back to binary and interpret the data according to the data components listed above. [2]

(f) If this system is to be implemented for up to five different rooms, what changes need to be made to the data representation? [2]