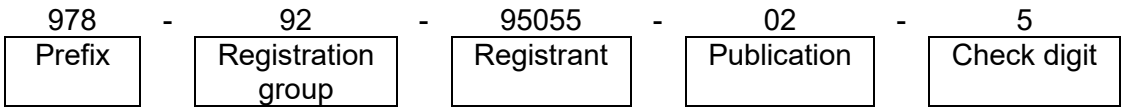
1 An ISBN is an International Standard Book Number and is a unique identifier for each book.

Each ISBN consists of 13 digits, which is made up of 5 elements with each element being separated by spaces or hyphens. Three of the five elements may be of varying length:

- **Prefix element** – currently this can only be either 978 or 979. It is always 3 digits in length
- **Registration group element** – this identifies the country, geographical region, or language area participating in the ISBN system. This element may be between 1 and 5 digits in length
- **Registrant element** - this identifies the publisher or imprint. This may be up to 7 digits in length
- **Publication element** – this identifies the edition and format of a specific title. This may be up to 6 digits in length
- **Check digit** – this is always the final single digit that mathematically validates the rest of the number. It is calculated using a Modulus 10 system with alternate weights of 1 and 3.

An example of an ISBN with check digit 5 is shown below:



(a) Describe a purpose of the check digit in an ISBN. [2]

Detect common input errors such as transcription error and transposition error [1], which helps to maintain data accuracy in book identification systems [1]

(b) Suggest another suitable validation technique for ISBN. [1]

Length check – check that ISBN consists of 13 digits [OR]
Format check – check that each of the elements fulfil the criteria specified

A school library system needs to store and search for information about books using their unique ISBNs. New books are added into the library inventory and damaged or older books are condemned from time to time.

The developers propose the use of a hash table to store and retrieve the book records.

(c) Explain how the library system would use a hash function and a hash table to store and search for books by their ISBNs. [3]

Each ISBN will be processed by the same hash function to generate an index/key [1], which corresponds to the position in the hash table where the book record will be stored [1]. When searching for a book using its ISBN, it uses the same hash function to generate the index/key to locate the record directly. [1]

(d) Give **three** features of an effective hashing algorithm. [3]

Any of the following 3:
1. The function should reduce the chances of different keys producing the same hash (collisions). This avoids clustering and reduces the chance of collisions, which keeps lookup, insert, and delete operations efficient (close to O(1)).
2. Equal probability of generating the indices or the indices generated are spread evenly across all the full range of the indices of the table. (Uniform Distribution)
3. The same input must always produce the same hash value. This ensures that a key can be found reliably each time it is accessed. (Deterministic)
4. Quick computation of the hash address.

(e) Explain the meaning of a collision in this context. [2]

A collision occurs when elements in a hash table share the same hash value. When this occurs, the book record cannot be stored in that space and an alternative space has to be sought out to store that book record. [1]
When searching for a book record, and another book record is found, it does not mean that the search item is not in the table. [1] A collision resolution algorithm is used to continue searching for the item at other locations before a conclusion is made.

(f) Describe **one** method that can be used to handle the consequence of a collision. [2]

Linear probing is a method to deal with collision. [1] This is done by probing sequentially the next space or using an interval (e.g. check every 3rd slot) until an empty slot is found. [1]
[OR]
Chaining can also be used to handle the consequence of a collision. In chaining, records are stored in a node of one of the many linked lists. Each array item of the hash table will store the address of one linked list. Whenever a key of a record gets hashed to an address where another record has already been hashed to, the new record will be added to the end of the list that its hashed address points to.

Students on an internship with the library ask their developer mentors why the book records, ordered by their ISBNs, were not stored in a linked list instead.

(g) Describe **one** disadvantage of storing and searching the records ordered by ISBNs using a linked list. [2]

Locating a book will require traversing through the linked list starting from the head of the linked list, which is slow process of O(n) time complexity [1]
This affects not just the searching process, but also the insertion and deletion process to maintain the order of the records by ISBN [1]

2 A program implements a Merge Sort algorithm to order values into ascending order.

The contents of an array are shown:

12	10	5	1	24	9	6

(a) Explain how a Merge Sort algorithm will sort the data in the array into ascending order. [3]

Divide the array into smaller subarray until each subarray has only one element [1]
Merge the subarrays back together in the correct (sorted) order. [1]
Illustrate using the array given [1]

(b) Using Big-O notation, state the worst-case time complexity of Merge Sort. [1]

O(log n)

(c) The Merge Sort algorithm uses recursion.

(i) State **three** features of a successful recursive function. [3]

- A recursive algorithm must call itself, recursively.
- Has a base case or terminating condition where the function stops calling itself
- A recursive algorithm must change its state and move toward the base case

(ii) State the purpose of the given pseudocode function:

```
FUNCTION iterative (string1)
    left = 1
    right = LENGTH(string1)
    WHILE left < right
        IF string1[left] <> string1[right]
            RETURN FALSE
        END IF
        left = left + 1
        right = right - 1
    END WHILE
    RETURN TRUE
END FUNCTION
```

[1]

This function checks whether `string1` is a palindrome.

(iii) Rewrite the pseudocode function in **(c)(ii)** using recursion. [3]

```
FUNCTION recursive(string1)
    IF LENGTH(string1) <= 1   [Base case: 1]
        RETURN TRUE
    END IF

    IF string1[1] <> string1[LENGTH(string1)]   [1]
```

```
        RETURN FALSE
    END IF

    RETURN recursive(SUBSTRING(string1, 2, LENGTH(string1) -
    2)) [1]
END FUNCTION
```

3 A software company is writing a program for a vehicle hire business. Both cars and vans are available for hire.

For all vehicles, the data that will be stored include:

Vehicle Registration Number (VRN)
Total distance travelled (km)
Date hired
Date of return
Cost per day
Availability for hire

For cars, the additional data stored include:

Fuel type (petrol, diesel, electric, hybrid)

For vans, the additional data stored include:

Maximum load (kg)

The odometer in the vehicle displays the total distance the vehicle has travelled since manufacture.

When a vehicle is hired:

- total distance travelled is set to the odometer's value
- date hired is set to the current date
- date of return is set to the date the vehicle is expected to be returned
- availability for hire is set to FALSE.

When a vehicle is returned:

- hire cost is returned as the cost per day multiplied by the number of days the vehicle was hired
- total distance travelled is set to the odometer value
- date of return is set to the current date
- availability for hire is set to TRUE.

Object-oriented programming will be used to model this situation.

(a) Draw a class diagram that shows the following for the situation described above.

- the superclass
- any subclasses
- inheritance
- properties
- appropriate methods. [8]

(b) Explain how encapsulation helps to prevent inadvertent data modification using examples from this situation. [2]

(c) Explain how inheritance promotes code reuse using examples from this situation. [2]

The programmer uses a database to store vehicle hire data. Each row in the table represents a vehicle hire record. A vehicle must be hired for a minimum of one day.

Vehicle hire data is stored in the following table:

VRN	TotalDistance	DateHired	DateReturned	CostPerDay	Availability	FuelType	MaxLoad

(d) **(i)** Identify the primary key for the above table. [1]

(ii) Identify suitable SQL data types for the following columns:

1. DateHired [1]

2. Availability [1]

(e) Explain why the above table is not in Third Normal Form (3NF). [3]

A database consultant rewrites the above table into the following table descriptions:

Vehicle (VRN, Type, TotalDistance, CostPerDay, Availability)
Hire (VRN, DateHired, DateReturned)
Car (VRN, FuelType)
Van (VRN, MaxLoad)

The vehicle type is stored as either `'Car'` or `'Van'`.

(f) Identify the foreign key(s) of the tables Hire, Car and Van. [1]

(g) Draw an entity-relationship (ER) diagram showing the four tables and the relationships between them. [4]

(h) Write an SQL query to retrieve from the above tables the VRNs of hybrid cars and their latest date of hire. [6]

3	**Data Management, Object-Oriented Programming**
3a	Vehicle: private attributes present
	Vehicle: public getter methods and appropriate setter methods
	Car and Van subclasses
	correct inheritance relationship between Car/Van and Vehicle
	additional private attributes and public getter/setter for Car
	additional private attributes and public getter/setter for Van
	`hire_vehicle()` method
	`return_vehicle()` method
	Polymorphism: all Vehicle instances have same method names
3b	**private attributes cannot be directly accessed or modified** outside of the class, only through public methods, to prevent the object falling into an inconsistent state
	e.g. hire_vehicle() method ensures the total distance, date hired, return date, and availability are updated in a consistent way
3c	inheritance allows subclasses to access public methods of the superclass
	Car and Van can access the public getter methods of Vehicle e.g. getVRN(), and do not need to duplicate the same code.

3di	(VRN, DateHired)
3dii	DateHired: DATE / STRING
	Availability: BOOLEAN / INTEGER
3e	Not in 3NF
	3NF requires 2NF
	non-key attributes CostPerDay, FuelType, MaxLoad **depend only on VRN** (name at least one appropriate non-key attribute)
	and not on the entire PK (**partial dependence**)
3f	VRN
3g	All entities represented: Vehicle, Hire, Car, Van
	Vehicle <--1--1--> Car
	Vehicle <--1--1--> Van
	Vehicle <--1--n--> Hire
3h	`SELECT Vehicle.VRN, MAX(DateHired)`

	`FROM Vehicle`
	`  INNER JOIN Hire ON Vehicle.VRN = Hire.VRN`
	`  INNER JOIN Car ON Vehicle.VRN = Car.VRN`

	`WHERE Type = 'Car'`
	`  AND FuelType = 'hybrid'`
	`GROUP BY Vehicle.VRN`
	Correct SQL syntax

4 A software company is developing a print server that handles print jobs from multiple users. Each print job has a job ID and the number of pages.

(a) The print server manages the print jobs with a linear queue.
Explain why a linear queue would be appropriate for regular print jobs. [2]

Definition of queue as FIFO structure [1]
Explanation of appropriateness for print jobs (order preservation) [1]

A queue is a First-In-First-Out (FIFO) data structure where elements are added at one end (rear) and removed from the other end (front).
This is appropriate for print jobs as it ensures jobs are processed in the order they were submitted (effectively manage multiple print jobs without the risk of mixing them up), maintaining fairness in the printing system.

The company decides to implement the print queue as a circular queue using a static array of size 5.

(b) Explain **one** advantage and **one** disadvantage of using a circular queue instead of a linear queue for this situation, where both queues are implemented using a static array of the same size. [4]

Valid advantage [1]
Sufficient explanation in the context [1]

A linear queue stores print jobs sequentially and can become inefficient when it reaches capacity, as it cannot accept new print job until all the print jobs have been processed, potentially wasting space. In contrast, a circular queue wraps around and efficiently utilizes space by allowing a new print job to be added at the front if there is space due to dequeued job, thereby preventing wasted positions.

Valid disadvantage [1]
Sufficient explanation in the context [1]

A circular queue has a more complex implementation logic that requires additional calculations to handle the wraparound of the rear pointer and to determine when the queue is full.
For example, when adding a new print job, the program needs to check if (rear + 1) mod size equals front to determine if the queue is full, rather than simply checking if rear has reached the end of the array as in a linear queue. This increased complexity makes the code more difficult to maintain and debug if issues arise with the print job system.

(c) Draw a diagram showing the state of a circular queue after the following operations, showing the positions of front and rear pointers:

1. Enqueue jobs: 101, 102, 103, 104
2. Dequeue two jobs
3. Enqueue jobs: 105, 106 [3]

Correct array representation and element placement [1]
Correct front pointer position [1]
Correct rear pointer position [1]

[106] [ ] [103] [104] [105]
^Front (at index 2)

[106] [ ] [103] [104] [105]
^Rear (at index 0)

(d) The print server keeps a history of completed print jobs in a stack data structure.

(i) Explain how a stack data structure would be appropriate for implementing a "restore" feature that allows the system administrator to restore recently completed print jobs back to the queue. [2]

Explanation of stack's LIFO nature being suitable for restore/history [1]
Clear connection to how this benefits system administrator's restore operations [1]

A stack is appropriate for implementing an undo feature because its Last-In-First-Out (LIFO) structure naturally matches the way undo operations work. When a print job is completed, it is pushed onto the stack, allowing the most recently completed job to be easily accessed and restored first. This means that if the system administrator decides to undo the last completed job, they can simply pop it from the stack and restore it back to the print queue. This matches the typical user expectation that undo operations will reverse the most recent actions.

(ii) Write pseudocode for a procedure `restoreJobs` that takes three parameters: the history stack, the print queue and an integer `N`. The function should restore the `N` most recently completed jobs from the history stack back to the print queue, maintaining their original print order. [6]

Correct function header with parameters [1]

Appropriate use of temporary storage to maintain order [1]
Correct loop to handle N items [1]
Proper stack operations (pop from history and push to temp) [1]
Proper queue operations (pop from temp and enqueue to queue) [1]
Handling edge cases (empty stack or N larger than stack size) [1]

```
PROCEDURE restoreJobs(historyStack : STACK, printQueue : QUEUE, N :
   INTEGER)
    // Declare temporary storage to maintain order
    DECLARE tempStack : STACK
    DECLARE count : INTEGER

    count ← 0

    // Check if N is valid
    IF N <= 0 THEN
        RETURN
    ENDIF

    // Move up to N jobs to temporary stack to reverse order
    WHILE NOT historyStack.isEmpty() AND count < N
        tempStack.push(historyStack.pop())
        count ← count + 1
    ENDWHILE

    // Move jobs from temporary stack to queue to restore original
   order
    WHILE NOT tempStack.isEmpty()
        printQueue.enqueue(tempStack.pop())
    ENDWHILE
ENDPROCEDURE
```

5 A company, ImageMaps, wishes to develop an application that scans a user's device for photographs, detect the associated location, and display the images on a map using the associated locations.

(a) State **two** actions the company must take regarding the collection and use of personal user data to comply with prevailing personal data protection laws. [4]

Notification: ImageMaps must inform customers of the purposes of data use through a privacy policy

Consent: ImageMaps must obtain customers' informed consent (e.g. through app permissions or webapp dialog box)

Purpose Limitation: ImageMaps must ensure employees do not use customer personal data for purposes other than the stated (e.g. cannot be used to train AI)

Access and Correction: ImageMaps must allow customers to see their stored personal data, and correct it if necessary.

(other appropriate obligations relating to collection and use, with appropriate action specified)

(b) ImageMaps needs to store a large quantity of data that will not be frequently accessed. State how they can reduce costs while doing so. [2]

Data not frequently accessed can be archived [1] by moving it to cheaper storage [1]

(c) State whether a native or web application is more suitable for ImageMaps's application. Explain your answer. [3]

Native application [1]

The application needs to access device storage to scan for photographs [1]

Web apps do not have the same capabilities as native apps, such as the capability to access device storage [1]

(d) When data is transmitted between user's device and ImageMaps' server, it is divided into packets that travel through the network. Describe how networking protocols address the following problems:

(i) Data packets may arrive out of order and require reassembly [2]

(ii) Data packets may be corrupted in transit. [2]

(i) Each packet is tagged with a sequence number [1]
The sequence number is included in the header to allow sequential reassembly [1]

(ii) The packet data is hashed to produce a checksum [1]
The recipient verifies the checksum against its hashed data [1]

6 A commercial bank is strengthening its cybersecurity framework after experiencing multiple security breaches.

(a) **(i)** Explain how a Trojan horse differs from a worm in terms of propagation and payload delivery. [2]

Trojan: Masquerades as legitimate software and tricks users to execute it, does not replicate itself (1 mark)

Worm: Self-replicating, spreads autonomously through networks (1 mark)

(ii) Describe two characteristics that make ransomware particularly dangerous to financial institutions. [2]

Encrypts critical financial data (1 mark)

Causes operational disruption during attacks (1 mark)

High ransom demands target valuable data (1 mark)

(Any 2 points, 1 mark each)

(b) The bank has deployed a firewall and Intrusion Prevention System (IPS).
Explain why implementing both a firewall and IPS provides better security than using either system alone. [2]

- Firewall filters basic malicious traffic (1 mark)
- IPS detects sophisticated attacks in payloads (1 mark)
- Defense-in-depth approach (1 mark)

(Any 2 points, 1 mark each)

(c) The bank must choose between biometric authentication and security tokens for employee access to sensitive systems. Evaluate which method would provide better security, giving **two** reasons for your choice. [3]

Biometric Authentication Preferred Because:

1. Unique physiological traits (1 mark)
2. Cannot be lost/stolen like tokens (1 mark)
3. Harder to replicate than token codes (1 mark)

OR

Security Tokens Preferred Because:

1. Not affected by physical changes (1 mark)
2. Easier to revoke/reissue (1 mark)
3. No biometric data storage risks (1 mark)

(Accept either position with valid reasoning, 1 mark per reason)

(d) Describe the complete process of generating and verifying a digital signature for an electronic funds transfer request. [4]

1. Hash document using a hashing algorithm to produce a hash digest (1 mark)
2. Encrypt hash digest with sender's private key and send message with cipher to recipient (1 mark)
3. Recipient decrypts cipher using sender's public key to obtain hash digest and verify the sender's identity (1 mark)
4. Recipient hashes document using the same hashing algorithm, compares it with the hash digest to ensure that the original document has not been tampered with. (1 mark)

(e) A bank employee wishes to email a contract document to a customer. To prevent tampering with the contract document, he attaches his digital certificate to the email alongside with the document.
Explain why this is insufficient to ensure the integrity of the document. [2]

- A digital is a method of ensuring that an encrypted message is from a trusted source as they have a certificate from a Certification Authority that confirms that the individual is who they claim to be in an online communication. (1 mark)
- However, it is unable to verify the authenticity of message content and whether the message has been tampered with. (1 mark)

- End of Paper -