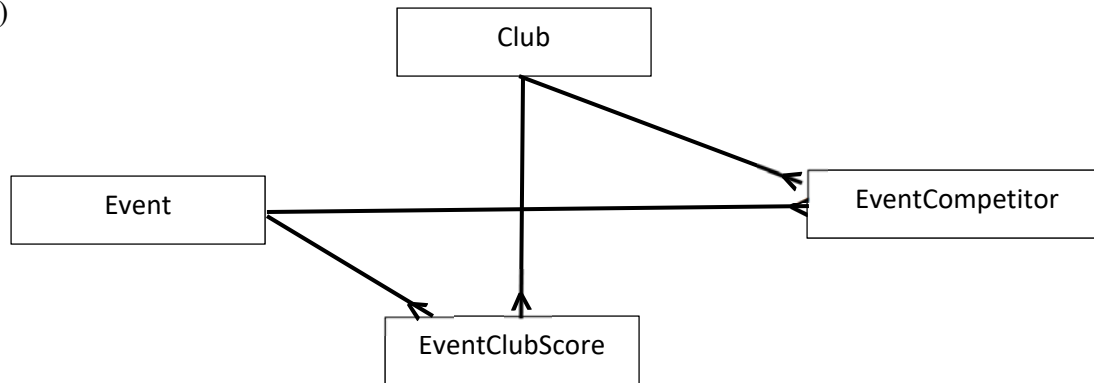


1(a)



(b)

Club (ClubName, SecretaryName, SecretaryEmail)

Event(EventNumber, Date, StartTime, Location)

EventCompetitor(CompetitorNumber, CompetitorName, FinalPosition, ClubName, EventNumber)

EventClubScore(ClubName, EventNumber, ClubScore)

(c)

```

SELECT C.ClubName, S.ClubScore, C.SecretaryName, C.SecretaryEmail
FROM Club C INNERJOIN EventClubScore S ON C.ClubName=S.ClubName
WHERE S.EventNumber=23
ORDER BY S.ClubScore DESC
    
```

(d i) 2 aims of normalisation – reduce data redundancy, prevent inconsistency data

(d ii) 2 requirements for table to be in 3NF

-all non-key attributes are fully functional dependent on the primary key. If the table has a composite key as the primary key, then each non-key attribute must be fully dependent on the entire composite primary key and not the subset of the primary key.

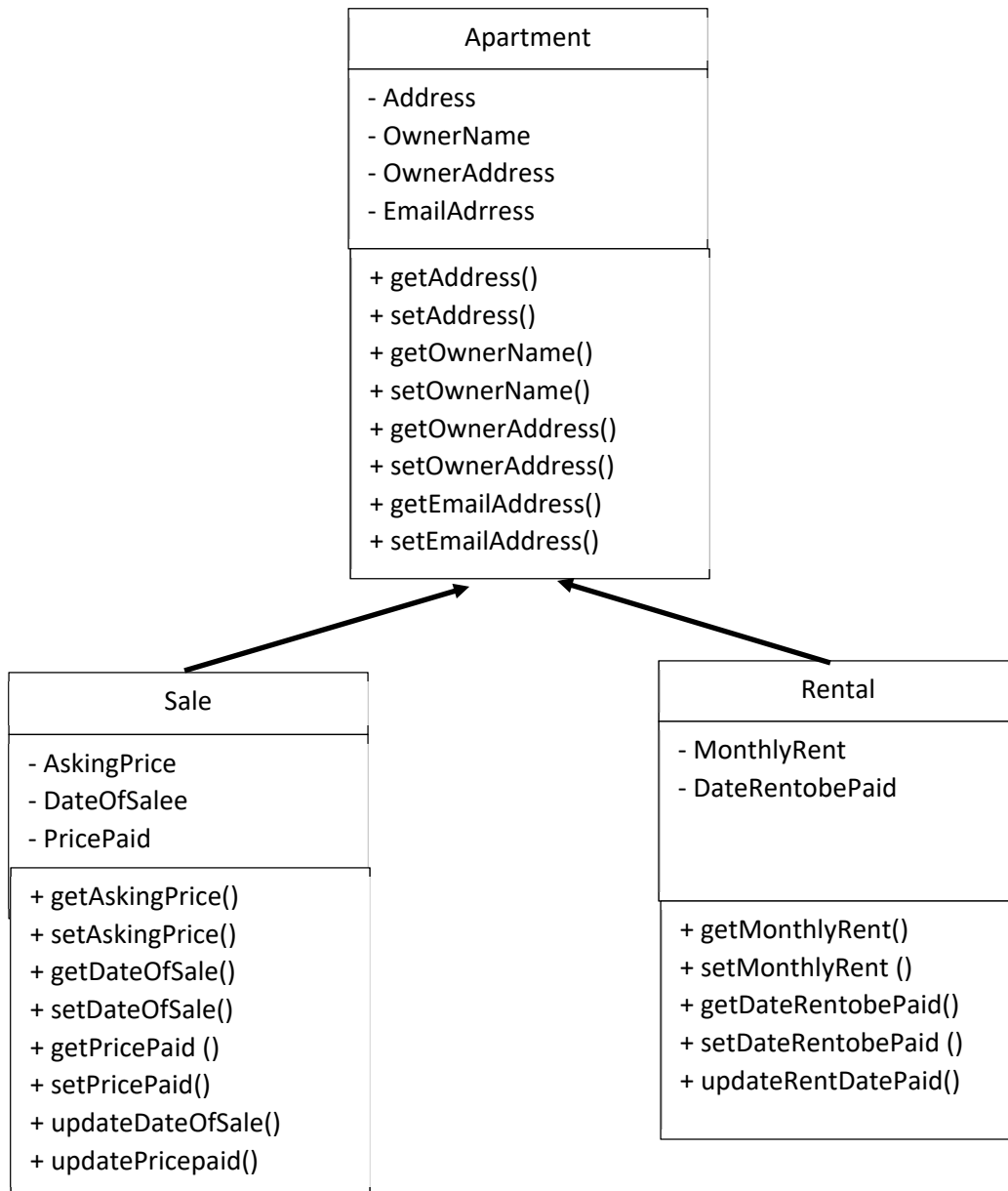
- all transitive dependencies must be removed. The table will contain only columns that are non-transitively dependent on the primary key.

(e) Range test. The score must be between 0 and 25 inclusive.

2 (a). A class is a blueprint or a template that defines the properties and methods for creating an object.

An object is an instance of a class that has specific values for its properties.

(b)



(c) Private properties provide a way to control access to the data of an object. By restricting access to the properties, the class designer can ensure that the data is only modified in a controlled way. This can help to prevent other parts of the program from accidentally modifying the state of the object.

(d) Code Reusability software developers can create a new class by inheriting from an existing class, they can reuse the code that already exists in the parent class and need not code all the code from scratch.

3(a) A message can be made meaningless to anyone other than the intended recipient by having the message encrypted using a cipher to convert the message into a unreadable format. The recipient can convert the message back into its original form by using decryption algorithm.

(b) Digital signatures are created by encrypting a hash value of the message using the sender's private key. The recipient can then use the sender's public key to decrypt the digital signature and obtain the hash value. The recipient can then regenerate the hash value of the received message and compare it with the decrypted hash value. If they match, the message has not been tampered with during transmission.

(c)

- Authentication is required to ensure only authorised personnel can access the network or the data kept in the server. This can be done by issuing the authorised personnel an userid and password for access.
- Authentication is important to ensure the true identity of a person in situations when personal data needs to be accessed. This can be done by performing a biometric verification with a fingerprint, an iris or retina scan, or facial recognition.

4(a).

- Arrays provide *constant time or $O(1)$ time access* to any element in the array. This is because the elements in an array are stored in contiguous memory locations.
- Arrays are more *memory efficient* than linked lists, because they do not require any extra memory for storing pointers to the next node in the list.

(b)

- Linked lists are better than arrays when it comes to *inserting or deleting elements, because they only require changes to the pointers of adjacent nodes*, while arrays require shifting all elements after the insertion or deletion point.
- Linked lists allow for *efficient use of memory allocation*, since they can allocate memory for each node as needed, whereas arrays require contiguous memory allocation at the time of creation.

(c)

(i) Function Z is a recursive function

(ii) Line 2 and 3 represent the base case for the recursive function

(iii) The purpose of function Z is to count the number of items in the linked list.

(d)

```

FUNCTION REV(CurrPt)
    IF(CurrPt <> -1) THEN // Not End of List
        NextPt ← LL(CurrPt).Pointer
        IF(CurrPt = HEAD) THEN //Start
            LL(CurrPt).Pointer ← -1
            REV(NextPt)
        ELSE
            LL(CurrPt).Pointer ← Head
            Head ← CurrPt
            REV(NextPt)
        ENDIF
    ENDIF
ENDFUNCTION

```

(c)

- Merge sort has a worst-case time complexity of $O(n \log n)$. Quick sort has a worst-case time complexity of $O(n^2)$.
- For large arrays, the difference in worst-case time complexity between the two algorithms becomes significant. Quick sort, in the worst case, may take much longer to sort the array. This is because Quick sort relies heavily on the choice of pivot element. If the pivot element is chosen poorly, it may lead to a large number of comparisons and swaps. Merge sort guarantees that the worst-case time complexity will always be $O(n \log n)$, regardless of the input data.

5(a). The algorithm did not release the memory used to store the floor number when the condition, the elevator is already at the chosen floor, is true. As the memory is not released, after a long period of time, situation when there are not enough memory for storing the floor number will cause the elevator to cease operation.

(b)

Get some memory that will be used to store the chosen floor number

Store the chosen floor number into the memory

Is the elevator already at the chosen floor?

Yes: Finished

Release the memory used to store the floor number

No: Close the doors

Move to the required floor

Open the doors

Release the memory used to store the floor number

(c)

- The error is a runtime error that will happen when the program is not able to get enough memory for storing the chosen floor number. Since the amount of memory required might not be a large amount, it will take a long time for the error to surface.
- The condition which did not release the memory is a test case which only require the user to test once and it is not an error which is being tested normally.

6(a) i.

- The purpose of creating a backup is to ensure that important data, files, or information can be recovered in case of loss, corruption, or damage.
- Backups serve as a form of insurance against data loss and provide a way to restore critical data to its previous state.

ii.

- The purpose of archiving is to preserve and protect information and records that are no longer actively in use but still have historical, legal, or cultural value.
- Archiving involves transferring the records from their primary storage location to a secondary storage location, where they can be accessed and retrieved when needed.

(b)

- Storing backup copies of data off-site is important because it provides an additional layer of protection against data loss.
- In the event of a disaster, having a backup stored off-site ensures that the data is still available and can be recovered.

(c)

- Data loss: If a business does not back up its data, it is at risk of losing it in the event of a hardware failure, software corruption, or cyber-attack. This can lead to the loss of critical business data, including financial records, customer information, and other important documents.
- Downtime: If a business's data is lost, it may take time to recover or recreate it. This can lead to significant downtime, during which the business may not be able to operate normally or provide services to customers.
- Financial loss: Data loss or downtime can result in financial losses for a business, including lost revenue, missed opportunities, and expenses related to recovery and repair.
- Reputation damage: Data loss or downtime can also damage a business's reputation, particularly if it results in the loss of sensitive customer

7(a)

(i)

- Efficient use of network resources: Dividing data into packets allows for more efficient use of network resources. Because data is transmitted in discrete packets, the network can more easily allocate resources to different data streams and prioritize traffic based on its importance.

- Error detection and correction: Packets contain additional information such as checksums, which allow for error detection and correction. If a packet is corrupted or lost during transmission, the receiver can use the checksum to determine that an error has occurred and request that the packet be retransmitted.

(ii)

- Tracking packet delivery: By assigning a unique sequence number to each packet, the sender and receiver can track which packets have been successfully delivered and which ones are missing. If a packet is lost in transit, the receiver can request that the sender retransmit the missing packet(s).
- Ensuring correct packet order: Because packets can be transmitted out of order and still be reassembled correctly, assigning sequence numbers allows the receiver to ensure that the packets are reassembled in the correct order.
- Handling retransmissions: If a packet is lost in transit or arrives at the receiver with errors, the sender can use the sequence number to retransmit only the missing or corrupted packets, rather than retransmitting the entire data stream.
- Network congestion control: Packet sequence numbers can also be used to control network congestion. By assigning a unique sequence number to each packet, the sender can limit the number of packets that are sent at any one time, which helps prevent network congestion and improves overall network performance.

(iii)

- Source and destination addresses: The packet header contains the source and destination addresses of the sender and receiver, respectively. These addresses are used to ensure that the packet is delivered to the correct destination.
- Protocol information: The packet header identifies the protocol used for the transmission, such as TCP, UDP, or ICMP.

7(b)

Protocols are required to enable reliable communication over the internet for several reasons:

- Standardization: The internet is a global network that connects millions of devices across the world. Protocols provide a standardized way for these devices to communicate with each other, regardless of the manufacturer, operating system, or location.
- Error detection and correction: Protocols provide mechanisms for error detection and correction. For example, the Transmission Control Protocol (TCP) uses sequence numbers, acknowledgments, and checksums to ensure reliable data transmission and detect errors during transmission.
- Flow control: Protocols provide mechanisms for flow control, which is the process of regulating the rate at which data is transmitted over the network. This is important to prevent congestion and ensure that data is transmitted efficiently.
- Security: Protocols provide mechanisms for secure communication over the internet. For example, the Secure Sockets Layer (SSL) protocol is used to encrypt data transmitted over the network, preventing unauthorized access and ensuring data privacy.
- Interoperability: Protocols enable devices and software from different vendors to communicate with each other seamlessly. This is important for ensuring that the internet remains an open and accessible platform for all users.

7(c).

Routers direct arriving data packets to the correct device on the LAN (Local Area Network) using a process called routing. When a data packet arrives at a router, the router examines the packet header to determine the destination IP address. Based on the **destination IP address**, the router looks up the routing table to find the **best path to the destination**.

7(d)

The firewall serves as a barrier that filters incoming and outgoing network traffic based on a set of predefined security rules.

- Access control: A firewall can be configured to block unauthorized access attempts to the LAN. It does this by examining network traffic and comparing it against a set of rules that define what types of traffic are allowed and what types are not. For example, a firewall might be configured to block all incoming traffic except for traffic on specific ports that are required for normal network operations.
- Traffic filtering: A firewall can also filter network traffic based on its source, destination, type, and content. For example, a firewall might be configured to block all incoming traffic that contains specific keywords or malicious code. This helps to prevent malware from entering the LAN and causing damage to network devices and data.
- Intrusion prevention: A firewall can be configured to detect and prevent intrusion attempts from malicious actors. This is done by analyzing network traffic for patterns that indicate an intrusion attempt and then taking action to block or quarantine the traffic.

8(a)

- Linear search has a time complexity of $O(n)$. This means that the time required to find a specific record grows linearly with the size of the data set.
- Binary search has a time complexity of $O(\log n)$. This means that the time required to find a specific record grows logarithmically with the size of the data set.
- A hash table can provide constant time search, insert, and delete operations, on average, with a time complexity of $O(1)$. This is because the hash function maps the key to an index in the array, allowing for direct access to the record. In the best case, the hash function can map every key to a unique index, resulting in constant time operations. However, in the worst case, the hash function can map multiple keys to the same index, resulting in collisions and potentially degrading the performance of the hash table.

(b) A collision occurs when two or more keys in the search map to the same location in the hash table.

(c) In chaining, the hash table maintains a linked list of all the keys that hash to the same index. When a new key collides with an existing key, it is added to the end of the linked list.

(d)

- Uniformity: The hashing algorithm should be designed to distribute the keys uniformly across the hash table, which minimizes the number of collisions and ensures that the search operation is fast and efficient.
- Deterministic: the same key should always generate the same hash value using the hashing algorithm, allowing the hash table to store and retrieve the key-value pairs consistently.
- Fast computation: The hashing algorithm should be designed to generate hash values quickly, allowing for fast insertion, deletion, and search operations on the hash table.