Bangladesh Bank
Banker's Selection Committee Secretariat (BSCS)
Assistant Maintenance Engineer/Assistant Engineer (IT)
Exam Date: 08/05/2026

There are 15 (fifteen) questions in this section with total marks 200.
Answer ALL. The symbols have their usual meanings.
Marks of each question are written in brackets.
Assume any reasonable value if required.
Write all your answers within the space provided in the separate answer script.

1.	**A maintenance engineer is setting up a RAID 5 array with five hard drives, each having a capacity of 4 TB. What is the total usable storage capacity, and why is it not 20 TB? (15 Marks)**

In a RAID 5 array, the usable storage capacity is calculated as:

$$\text{Usable Capacity} = (N - 1) \times \text{Size of one drive}$$

Where:

- $N = 5$drives
- Each drive = 4 TB

So,

$$(5 - 1) \times 4 = 4 \times 4 = 16\ \text{TB}$$

Therefore, the total usable storage capacity is:

$$(5 - 1) \times 4\ \text{TB} = 16\ \text{TB}$$

Why is it not 20 TB?

RAID 5 uses **distributed parity** for fault tolerance.
The equivalent capacity of **one drive** is reserved for storing parity information, which allows the system to recover data if one drive fails.

- Total raw capacity = $5 \times 4 = 20$TB
- Parity overhead = 4 TB (capacity of one drive)
- Usable capacity = $20 - 4 = 16$TB

Thus, RAID 5 sacrifices the space of one disk to provide redundancy and data protection.

2.	**Distinguish between Preventive and Corrective maintenance with a real-world example for each in a data center environment. A critical system consists of three components (Power Supply, Motherboard, and Storage) connected in a series configuration. If each component has a reliability of 0.95, what is the total reliability of the system? (15 Marks)**

Difference Between Preventive and Corrective Maintenance

Preventive Maintenance	Corrective Maintenance
Maintenance performed **before a failure occurs** to reduce the chance of breakdown.	Maintenance performed **after a failure occurs** to restore operation.
Planned and scheduled regularly.	Unplanned and done when a problem appears.
هدف is to improve reliability and avoid downtime.	Goal is to repair and recover the failed system.
Usually lower long-term cost.	Can cause higher downtime and emergency repair cost.

Real-World Examples in a Data Center

- **Preventive Maintenance Example:**
  Regularly cleaning server dust filters, checking UPS batteries, updating firmware, and monitoring server temperature to prevent overheating and failures.
- **Corrective Maintenance Example:**
  Replacing a failed hard disk or repairing a damaged power supply after a server suddenly stops working.

Reliability of the System

For components connected in a series configuration, the total system reliability is the product of the reliabilities of all components.

Given:

- Reliability of Power Supply = 0.95
- Reliability of Motherboard = 0.95
- Reliability of Storage = 0.95

So,

$$R_{system} = 0.95 \times 0.95 \times 0.95 = 0.857375$$

Final Answer

$$R_{system} \approx 0.857$$

or about 85.7% reliability.

3.	**At which layer of the OSI model does a standard Router primarily operate, and what is the specific name of the Protocol Data Unit (PDU) at this layer? A router has four contiguous /24 routing table entries: 10.1.0.0/24, 10.1.1.0/24, 10.1.2.0/24, and 10.1.3.0/24. If you summarize (supernet) these into a single route,what will be the new CIDR notation?** **(15 Marks)**

A standard **Router** primarily operates at the **Network Layer (Layer 3)** of the OSI model.

- **OSI Layer:** Network Layer (Layer 3)
- **PDU at this layer: Packet**

Route Summarization (Super netting)
Given routes:

- 10.1.0.0/24
- 10.1.1.0/24
- 10.1.2.0/24
- 10.1.3.0/24

These are four contiguous /24 networks.
Since:

$$4 = 2^2$$

we can combine 4 networks by reducing the subnet mask by 2 bits:

$$/24 - 2 = /22$$

So the summarized route becomes:

$$10.1.0.0/22$$

4.	**What does the Consistency property in ACID guarantee during a bank fund transfer transaction? You are designing the schema for a Savings_Accounts table. Write the specific SQL constraint clause to ensure that the current_balance column can never drop below zero.** **(15 Marks)**

Consistency Property in ACID

The **Consistency** property in ACID ensures that a database transaction always moves the database from one **valid state** to another **valid state** while maintaining all defined rules, constraints, and integrity conditions.
In a Bank Fund Transfer
During a fund transfer:

- The amount deducted from one account must be correctly added to the other account.
- The database must never remain in a partial or invalid state.
- If any step fails, the entire transaction is rolled back.

For example:

- Account A = 10,000
- Account B = 5,000
- Transfer = 2,000

After successful transaction:

- Account A = 8,000
- Account B = 7,000

Total money remains unchanged, preserving database consistency.

SQL Constraint Clause
To ensure that the current_balance column never becomes negative, use a CHECK constraint.
CHECK (current_balance >= 0)

Example inside table creation:

```
CREATE TABLE Savings_Accounts (
   account_no INT PRIMARY KEY,
   account_holder VARCHAR(100),
   current_balance DECIMAL(12,2),
   CHECK (current_balance >= 0)
);
```

5.	**Briefly explain "Circular Wait."** **In a Resource Allocation Graph (RAG), if a cycle exists, does it absolutely guarantee that a Deadlock has occurred? Explain briefly.** **(15 Marks)**

Circular Wait

Circular Wait is one of the four necessary conditions for deadlock.
It occurs when a set of processes are waiting for resources in a circular chain.
Example:

- Process (P_1) is waiting for a resource held by (P_2)
- (P_2) is waiting for a resource held by (P_3)
- (P_3) is waiting for a resource held by (P_1)

Thus, each process waits indefinitely for another process in the cycle.

Does a Cycle in RAG Always Mean Deadlock?

Answer:

No, not always.

- If the Resource Allocation Graph (RAG) has **only one instance of each resource type**, then a cycle **guarantees deadlock**.
- If there are **multiple instances of a resource type**, a cycle may exist without deadlock.

Brief Explanation

With multiple resource instances, another free instance may become available and break the waiting chain, allowing processes to continue execution.
Therefore:

- **Cycle + single instance resources → Deadlock guaranteed**
- **Cycle + multiple instance resources → Deadlock may or may not occur**

6.	**A server suddenly starts sending abnormal traffic to external systems.** **Analyze the possible cause and suggest immediate actions.** **(15 Marks)**

A server suddenly sending abnormal traffic to external systems may indicate a **security compromise, malware infection, or misconfiguration**.

Possible Causes

1. **Malware or Botnet Infection**
   - The server may be infected with malware or ransomware and used to send spam, DDoS traffic, or malicious requests.
2. **Unauthorized Access / Hacking**
   - An attacker may have gained access through weak passwords, unpatched vulnerabilities, or stolen credentials.
3. **Compromised Application**
   - A vulnerable web application or service may be exploited to generate abnormal outbound traffic.
4. **Misconfigured Software or Scripts**
   - Faulty automation scripts, backup tools, or applications may accidentally create excessive network traffic.
5. **Data Exfiltration**
   - Sensitive data may be transferred to external attackers.

Immediate Actions

1. Isolate the Server

- Disconnect the server from the network or restrict outbound traffic using firewall rules to prevent further damage.

2. Analyze Network Traffic

- Check logs, running processes, active connections, and unusual ports using tools like:
  - netstat
  - tcpdump
  - Firewall/SIEM logs

3. Scan for Malware

- Run antivirus and anti-malware scans to detect malicious software.

4. Check User Accounts and Access

- Review recent logins, privilege escalations, and suspicious accounts.
- Change passwords and revoke compromised credentials.

5. Identify the Root Cause

- Examine application logs, security logs, and recent configuration changes.

6. Apply Security Fixes

- Patch vulnerabilities, update software, and close unnecessary services or ports.

7. Restore if Necessary

- If heavily compromised, restore the system from a clean backup.

8. Monitor Continuously

- Enable continuous monitoring and intrusion detection to ensure the threat is removed.

Conclusion

Abnormal outbound traffic is often a sign of cyberattack or malware activity. Immediate isolation, investigation, malware scanning, and security remediation are essential to protect the server and network.

7	**Mapreduce,hadoop (repeat previous)** **Assumption of the question are: "A financial institution needs to process massive server logs to detect fraud in real-time and perform predictive maintenance using ML. Identify 5 components of the Hadoop ecosystem suitable for this architecture and justify their roles."**

Hadoop Ecosystem Components and Their Roles

Component	Role in the Architecture	Justification
HDFS (Hadoop Distributed File System)	Distributed storage of massive server logs and transaction data	HDFS can store huge volumes of structured and unstructured data reliably across multiple nodes with fault tolerance and replication.
MapReduce	Parallel processing of large datasets	Processes massive log files efficiently by dividing tasks into Map and Reduce phases, enabling distributed computation for analytics and fraud detection.
Apache Spark	Real-time data processing and Machine Learning	Spark supports in-memory computation, fast stream processing, and ML libraries (MLlib), making it ideal for real-time fraud detection and predictive maintenance.
YARN (Yet Another Resource Negotiator)	Resource management and job scheduling	YARN manages cluster resources efficiently and schedules Hadoop/Spark jobs across nodes for optimized performance.
Apache Hive	SQL-like querying and analysis	Hive allows analysts to run SQL-style queries on large datasets for reporting, log analysis, and fraud investigation without complex programming.

8	**Http status code**

HTTP Status Codes

HTTP status codes are standard response codes returned by a web server to indicate the result of a client's request.

Categories of HTTP Status Codes

Code Range	Meaning
1xx	Informational
2xx	Success

3xx	Redirection
4xx	Client Error
5xx	Server Error

1xx – Informational

Code	Meaning
100	Continue
101	Switching Protocols

2xx – Success

Code	Meaning
200	OK
201	Created
204	No Content

3xx – Redirection

Code	Meaning
301	Moved Permanently
302	Found (Temporary Redirect)
304	Not Modified

4xx – Client Errors

Code	Meaning
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
405	Method Not Allowed

5xx – Server Errors

Code	Meaning
500	Internal Server Error
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout

Example

If a user requests a webpage that does not exist, the server may respond with:
HTTP/1.1 404 Not Found
This means the server could not find the requested resource.

9. Zero-day attack, data in transit & Data at rest technology,

A zero-day attack exploits a previously unknown vulnerability in software or hardware. These vulnerabilities are termed "zero day" because developers have had zero days to fix the flaw before it is exploited by hackers. Unlike more traditional attacks, zero day exploits are unique in that they take advantage of weaknesses that are not yet known to the public or the vendor. This makes them incredibly difficult to defend against, as there are no existing patches or updates that can prevent the attack.

Zero-day attacks can manifest in various forms, including malware, viruses, or even direct network intrusions. They often remain undetected until significant damage has been done. Once an exploit is identified, developers rush to create a patch, but by then, the damage might already be extensive. This urgency and the element of surprise are what make zero day attacks particularly dangerous and challenging to combat.

Data at Rest vs Data in Transit

Data in transit is actively moving across networks—such as emails, file transfers, or API communications—making it vulnerable to interception. In contrast, **data at rest** remains stored on devices, databases, or cloud environments, where unauthorized access is the primary concern.

Data in Transit Security

- **Encryption protocols (TLS, SSL, IPsec):** Cryptographic protocols to secure data transmission by encrypting network traffic to prevent unauthorized access
- **Secure transmission protocols (HTTPS with TLS):** Encrypt data in transit to protect it from interception
- **Authentication mechanisms (MFA, digital certificates):** Verify user identities and ensure that only authorized parties can access transmitted data

Data at Rest Security

- **Encryption of stored files and databases:** Encrypting stored data ensures that even if unauthorized access occurs, the data remains unreadable without the correct decryption key
- **Access control policies (least privilege, role-based permissions):** Implementing strict access controls limits data exposure by ensuring only authorized users can access sensitive information
- **Physical security measures (restricted server access, hardware encryption):** Securing physical data storage locations and using hardware-based encryption protects data from unauthorized access to hardware

Data at Rest

- Hard drives
- Databases
- Cloud storage

Risks:

- Unauthorized access
- Data breaches

Data in Transit

- Emails
- File transfers
- API communications

Risks:

- Interception
- MITM attacks

10.	Preventive and corrective maintenance, series system reliability calculation

Preventive Maintenance
Preventive maintenance is the maintenance performed **before a failure occurs** to reduce the possibility of system breakdown and improve reliability.
Features
- Planned and scheduled
- Helps avoid unexpected failures
- Reduces downtime
- Increases equipment lifespan

Example
Regularly cleaning servers, checking UPS batteries, updating firmware, and monitoring cooling systems in a data center.
Corrective Maintenance
Corrective maintenance is the maintenance performed **after a failure occurs** to restore the system to normal operation.
Features
- Unplanned maintenance
- Done after detecting faults
- May cause temporary downtime
- Focuses on repair and recovery

Example
Replacing a failed hard disk or repairing a damaged server power supply after a system crash.
Difference Between Preventive and Corrective Maintenance

Preventive Maintenance	Corrective Maintenance
Done before failure	Done after failure
Planned and scheduled	Unplanned
Prevents breakdown	Repairs breakdown
Lower downtime	Higher downtime
Improves reliability	Restores operation

Series System Reliability Calculation
In a **series system**, all components must work properly for the system to function. Therefore, total reliability is the product of the reliabilities of all components.
Formula

$$R_{system} = R_1 \times R_2 \times R_3 \times \cdots \times R_n$$

Example Calculation
Suppose:
- Power Supply reliability = 0.95
- Motherboard reliability = 0.95
- Storage reliability = 0.95

Then,

$$R_{system} = 0.95 \times 0.95 \times 0.95 = 0.857375$$

Final Reliability

$$R_{system} \approx 0.857$$

or approximately **85.7%**.

General part:
Math:
Eigen value+matrix related,
Probabilty of defective items(repeat previous)
English:
Two analogy:
Famished:hunger::find ans
Cacophony: sound::find ans
Two translation:
১. হিসাবকারী তার কাজের খুটিনাটি বিষয়ে খুব যত্নশীল ছিলেন।
২.রাজনীতিবিদ সত্যকে অস্পষ্ট করতে চেষ্টা করেছিলেন।

প্রশ্ন কালেকশনঃ
মোস্তাফিজুর রহমান